2019

A mechanism design for Crowdsourcing Multi-Objective Recommendation System

Eiman Aldahari

A mechanism design for Crowdsourcing Multi-Objective Recommendation System

Eiman Aldhahri

February 13, 2019

Contents

1 Introduction **1**

2 Leveraging Crowdsourcing in Cloud Application Development **5**

Motivation . . . 6
SaaS vs Traditional Software . . . 8
Crowdsourcing for SaaS Development . . . 8
Crowdsourcing Facilities . . . 9
Challenges in Software crowdsourcing . . . 12
Challenges as Addressed by Three Crowdsourcing Platforms . . . 15
Benchmarking SaaS-crowdsourcing . . . 15
Crowdsourcing Approaches . . . 16
Conclusion . . . 18

3 Background and Related Work **22**

Recommendation System . . . 23
General Recommendation Papers . . . 24
Crowdsourcing Recommendation Papers . . . 35
Conclusion . . . 45

4 Goals and Contribution **46**

Goals and Contribution . . . 46
Mechanism Design . . . 46
Stakeholders' Goals . . . 47
Contribution . . . 47
Conclusion . . . 48

5 Problem Formulation **49**

Stakeholders Role . . . 49
Worker . . . 49
Employer . . . 49
5.0.1 Service Provider . . . 50
workflow . . . 50
Conclusion . . . 51

6 Solution **52**
The Recommendation Model 52
Worker Objective 52
Employer Objective 58
Service Provider Objective 63
Discount Factor 63
Conclusion 64

7 Cold Start Problem **65**
New Worker 66
Overspecialized 66
New Employer 67
New Task 68
Conclusion 69

8 Testing and Analysis **70**
Baseline Model 70
Data Set 70
Experimental Procedure 71
Illustrated Example 73
Computation Complexity and Scalability 77
Conclusion 77

9 Future Work **78**

10 Publications **80**

11 Appendix **89**
Appendix A: Experimental code 89
Appendix B: Dataset 98

List of Tables

2.1 Comparison of In-House, Global Distributed, Outsourcing, Crowdsourcing Software Development 7
2.2 Crowdsourcing Challenges as Addressed be Three Software Crowdsourcing Platforms 20
2.3 Challenges as Applied to the Three Software Crowdsourcing Approaches 21

6.1 Task List 58

7.1 Workers Rating Matrix 69

8.1 Characteristics of Datasets 71
8.2 Posted Tasks 74
8.3 Processed Data 74
8.4 Applicant List 76
8.5 Worker Success Probability 76
8.6 Applicant Success Average 76

List of Figures

1.1 The four archetypes of crowdsourcing information system [32] 3

2.1 The existing approaches followed in Crowdsurcing Software . 9
2.2 A SaaS development crowdsourcing model that decomposes a project into service 18

3.1 Classification Scheme 23

6.1 Recommendation Model 53
6.2 Task list 60

8.1 Evaluating the Workers' objectives 72
8.2 Evaluating the Employers' objectives 73

Abstract

Crowdsourcing is an approach whereby requesters call for workers with different capabilities to process a task for monetary reward. The emergence of crowdsourcing has drawn increasing attention in recent years as a revolutionary phenomenon. Although crowdsourcing is still considered a developing approach, early signs expectations are promising. The advantage of crowdsourcing resides in its ability to facilitate access to diverse skilled workers to process the outsourced tasks at reduced time and cost. Moreover, crowdsourcing helps reduce unemployment by offering on-demand employment opportunities. With the vast amount of tasks posted every day, satisfying the workers, requesters, and service providers who are the stakeholders of any crowdsourcing system is critical to its success. To achieve this, the system should address three objectives: (1) match the worker with suitable tasks that fit the worker's interests and skills and raise the worker's rewards and rating, (2) give the requester qualified solutions at lower cost and time and raise the employer rating, and (3) raise the task acceptance rate, which will raise the aggregated commissions accordingly. For these objectives, we present a mechanism design capable of achieving holistic satisfaction using a multi-objective recommendation system. The proposed model is designed as an interactive system where every worker and employer could set the parameters that meet their goals. In contrast, all previous crowdsourcing recommendation systems have been designed to address one stakeholder. Moreover, no previous crowdsourcing recommendation systems have considered the other party's behavior to provide more qualified recommendations as we have done. Furthermore, we conducted a survey of one type of macrotask, namely a cloud application development to emphasize the importance of using crowdsourcing for macrotask. We identified its challenges and explored the facilities that support addressing these challenges. We also reviewed two widespread existing approaches for software development crowdsourcing and propose a novel approach. Additionally, we evaluated our proposed approach for its ability to address these challenges and provide future adopters with a list of attributes to assist in choosing the right crowdsourcing service. Finally, we evaluated our model with synthesized datasets. The experimental simulation showed the superiority of the proposed model compared with two other baseline models.

Chapter 1

Introduction

In 2006, Wired Magazine introduced the term crowdsourcing as "the act of a company or institution taking a function once performed by employees and outsourcing it to an undefined (and generally large) network of people in the form of an open call" [42]. Crowdsourcing is thus a process where a crowdsourcer (i.e, requester) outsources tasks to crowdsourcees (i.e. a large network of crowd workers) for monetary reward. The advantage of crowdsourcing resides in the ability of the employers to access a large pool of highly skilled workers who are able to process the outsourced tasks in a reduced amount of time and cost compared to in-house workers [42, 11, 86]. Moreover, crowdsourcing facilitates access to a more diverse talent pool than what might be available locally. Recently, there has been a significant trend towards crowdsourcing systems, and several major crowdsourcing platforms have emerged, such as ClickWorker [14], CloudCrowd [16], UpWork [84], and the well-known Amazon Mechanical Turk [4].

Crowdsourcing systems have three stakeholders: the worker, the employer, and the service provider. The employer posts tasks to the crowd with a deadline and monetary reward. Workers apply to tasks that could increase their reward and rating. The service provider's role is to offer a recommendation list that matches workers with tasks accurately in order to maximize the commission from accepted tasks.

Due to the large number of tasks and workers available on crowdsourcing systems, finding an appropriate task (or set of appropriate tasks) and a worker (or set of workers) is a strenuous and time-consuming process [92, 31]. What constitutes an appropriate tasks depends mainly on two factors: interest and skills [31]. Worker interest is measured based on multidimensional factors that are weighted differently for each worker, including the monetary reward and the worker's rating score. Moreover, selecting the most qualified worker is also a challenge even if we consider the worker's rating score. This score could reflect the worker's overall proficiency rather than the specialized rating for each skill possessed. The aforementioned task-worker matching is a major factor to eliminate low-quality solutions, which is an important problem in crowdsourcing data management [55, 26].

Another problem that could affect the stakeholders goals is if a worker gets a list of recommended tasks and works on a large number of tasks at the same time, which could decrease solution efficiency for some tasks. As an alternative, part of these tasks could be assigned to less experienced workers who have more time, which could increase the solution efficiency. In another scenario, if we recommend tasks to the most efficient worker, the employer's goal will be satisfied. However, the worker may get busy processing low monetary tasks and miss tasks offering a higher monetary reward. Therefore, a well-structured recommendation system, which satisfies all stakeholders and addresses the aforementioned difficulties, should be constructed. Such a system would entail workers finding their preferred task, employers getting a higher quality solution, and service providers increasing the task acceptance rate to increase their platform's income and popularity.

Crowdsourcing systems can be grouped into four archetypes based on the platform's main function: crowd processing, crowd rating, crowd solving, and crowd creation [Figure 1] [31]. Crowd processing systems seek microtasks that do not require specific skills from workers. These systems rely on an accumulated solution from independent workers and validate the solution based on their identifications. Amazon Mechanical Turk [4] is an example of such a processing system. Crowd rating systems seek nonspecific skilled workers' perspectives on a given topic and then aggregate their opinions to deduce an overall rating, which is what TripAdvisor does [82]. Crowd solving systems seek a task that requires certain skilled workers, where solutions are acquired independently, as with InnoCentive [45]. Crowd creation systems seek some defined tasks from workers who have different specific skills, where the submitted solutions are aggregated to include the overall task solution, as with Wikipedia [31].

Crowdsourcing systems can also be classified based on their behavior, which can be competitive or hiring [3]. In a competitive behavior system, any worker may contribute and process the task without permission to start from the employer. In contrast, in hiring behavior system, employers need to grant their permission to the worker before he/she can start processing the task. In competitive crowdsourcing, the prize goes to one or more workers who provide the best solution. In hiring crowdsourcing, a worker receives the rewards for the solution based on its correctness and conformity to the employer requirements. Moreover, tasks in crowdsourcing systems can be classified as microtasks (e.g., labeling an image), which take several seconds, and macrotasks (e.g., creation of an analytical paper, web design), which take more time [55].

Various areas contribute to building recommendation systems, including cognitive science, approximation theory, information retrieval, forecasting theories, management science, and marketing [72].

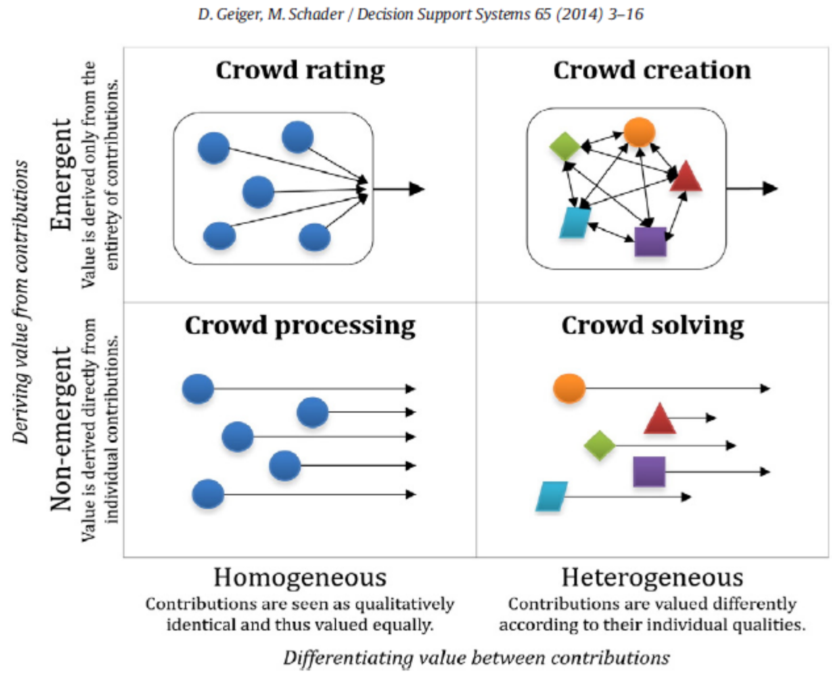


Figure 1.1: The four archetypes of crowdsourcing information system [32]

The first step in any recommendation system is building the user profile, which contain user preferable features either explicitly or implicitly. Implicit profiles are based on users' previous behavior. User behavior can be interpreted as preferable features. For instance, if a user has purchased a book in programming, we could infer that a user preferable feature is programming. Explicit profiles are based on asking users to complete a preferable features form. Several techniques could use this information, including a collaborative filtering approach, a content-based approach, and a hybrid approach that combines the other two approaches.

Collaborative filtering finds a group of users with similar behavior and gives a recommendation for a user depending on the group's previous choices. In collaborative filtering, the item's features and the user's preferable attributes would not be considered to make a decision on what to recommend. It also suffers from a cold start problem, which is a new user or a new item. On the other hand, content-based approaches focus on the user's preferable attributes and the item's features and use this information to construct a recommendation despite other users' behavior. Content-based approaches suffer from overspecialized, limited content analysis because the system will not differentiate between two different items with the same features and could choose the less efficient one. The new user problem is another issue.

The hybrid approach has been proposed to overcome the limitations of the other two approaches and combine them into single approach in order to provide a more accurate recommendation.

As the majority of existing crowdsourcing research has focused on microtasks, we chose to focus on macrotasks, which are an important research topic [55]. In spite of the fact that microtask crowdsourcing is more widespread than macrotask crowdsourcing, macrotask crowdsourcing platforms could be very effective if well designed and managed. Cloud application development or SaaS development is an example of harnessing crowdsourcing to find solutions for macrotasks. The nature of cloud application infrastructure, which operated by a third party in a remote data center facilitate the application development to be be outsourced [75]. This kind of crowdsourcing platform consists of a set of internal and external developers, a set of software development tools, and collaboration tools that enable resource sharing and automated collaboration to implement and maintain software. This is known as the crowdsourcing ecosystem. Through such a system, developers can collaboratively build software on top of a single platform [12, 10]. Chapter 2 discuss this matter in detail.

We assume that every stakeholder acts selfishly in order to maximize profit. Because of this assumption, we present a mechanism design based on a multi-objective recommendation system to achieve holistic satisfaction through the following: matching the worker with a suitable task that fits the worker's skills, increasing the worker's rewards and rating, giving employers better solutions at lower cost without affecting their rating, and raising the task acceptance rate, which in turn will increase the aggregated commissions.

In Chapter 2, we survey SaaS crowdsourcing to identify its challenges and explore the crowdsourcing facilities that support addressing these challenges. Furthermore, we review two widespread existing approaches for software development crowdsourcing and propose a novel approach. Additionally, we evaluate our proposed approach for its ability to address these challenges. Finally, we provide future adopters with a list of attributes to assist them in choosing the proper crowdsourcing service. Chapter 3 describes the related literature, chapter 4 describes the study's goals and contributions, chapter 5 contains the problem formulation, chapter 6 describes the proposed recommendation model, chapter 7 describes the cold start problem, chapter 8 describes the experiment, chapter 9 proposes future research, and chapter 10 lists the publications of the researchers.

Chapter 2

Leveraging Crowdsourcing in Cloud Application Development

Crowdsourcing is still considered a new, developing approach, especially for macrotasks; however, the signs and expectations are promising. As the majority of existing crowdsourcing research has focused on microtasks, we chose to focus on macrotasks, which are considered an important research topic [48]. In spite of the fact that microtask crowdsourcing is more widespread, macrotask crowdsourcing platforms could be very effective if used properly. This study sheds light on cloud application development or SaaS development to emphasize the importance of utilizing crowdsourcing for macrotasks.

Recruiting a workforce with sufficient experience to develop software has consistently been a great challenge. In particular, there is an unexpectedly high demand for workers to move organizations' data and computation to the cloud. However, experts in SaaS development are hard to find [54]. This problem has led to the need to outsource tasks in software development to experts off-site. Software crowdsourcing is an emerging approach to hire workers from outside the organization to complete certain software development tasks. Crowdsourcing is thus a process where a crowdsourcer (i.e, requester) outsources tasks to crowdsourcees (i.e. a large network of crowd workers) for monetary reward.

Ecosystems support software development crowdsourcing by providing a unified platform that enables experts to contribute to various project tasks (e.g, requirements, design, implementation, testing, etc.). It provides an interface for interaction between crowdsourcers and crowdsourcees [10]. There are a number of commercial ecosystem platforms, such as CloudBees [15], which have a wide range of tools to simplify the development process. The CloudBees ecosystems for instance, provides tools for configuration management, building, continuous integration, code analysis, documentation, planning, tasks, backups, and artifact management.

The usefulness of crowdsourcing goes beyond solving the problem of finding the right experts for a given task. Companies are increasingly calling on outsiders to hear different voices, eliminate expert bias, and maximize access to a variety of solutions. In 2011, Nokia started the "Ideasproject," an online community created by Nokia to allow users and developers from all around the world to brainstorm [87]. Nokia employed the idea of crowdsourcing because they believed that user-centered innovation offers great advantages over manufacturer-centric innovation. The Ideasproject was founded on the philosophy of democratized innovation to enable users to create new products and services for themselves.

Despite the many useful crowdsourcing features currently available (e.g. infinite pool of skilled workers, on-demand hiring, obtaining alternative solutions), adopting crowdsourcing for SaaS development remains uncommon. SaaS development still faces challenges in different levels of development lifecycle, such as requirements, engineering, and testing [30, 48, 40, 47, 51], which have yet to be completely addressed. Our objective is to highlight the capability of crowdsourcing in developing effective SaaS. Hence, we pose the following questions:

- Can crowdsourcing be utilized in SaaS development to cover the shortage of experts and other obstacles?
- What are the challenges involved in SaaS crowdsourcing?

Motivation

This chapter is motivated by well-known approaches in software engineering (SE) such as (1) "Global distributed software development" where software is developed in multiple locations [38], (2) "Outsourced software development" where work is performed by known individuals/organizations and maybe "shipped" off-shore [70], and (3) "open-sourced software development" where software source code is publicly available to be used, changed, and/or shared [27]. Conducting software projects in multiple global locations or hiring others is likely to result in benefits such as cost reduction and reduced time-to-market [54, 72]. Therefore, these development approaches have been increasingly utilized by the industry [36].

Table 1 illustrates a comparison between software development crowdsourcing and the three aforementioned software development approaches in terms of publicity of participants, the anonymity of participants, sacrificing intellectual property, location diversity, and multiplicity. These aspects and their impacts were extensively researched and effective solutions were proposed in the previous literature. However, Table 1 demonstrates that crowdsourced software development shares all four aspects with the other three SE approaches. Thus, crowdsourcing software development tends to be subject to all the challenges–and perhaps some additional challenges–that may emerge due to the incorporation of two or more aspects.

Table 2.1: Comparison of In-House, Global Distributed, Outsourcing, Crowdsourcing Software Development

Aspect / Approach	*In-house*	*Global Dis.*	*Open sourcing*	*Outsourcing*	*Crowd sourcing*
Public Participants	*No*	*No*	*Yes*	*No*	*Yes*
Unknown Participants	*No*	*No*	*Yes*	*No*	*Yes*
Transfer Intellectual Property	*No*	*No*	*No*	*Yes*	*Yes*
Multiple Locations	*No*	*Yes*	*Yes*	*Yes*	*Yes*

The literature suggests there has been a substantial work where software manufacturers delegated to a third party to develop software fully or partially. However, providing SaaS is becoming a trend in the software industry nowadays. Software engineers use service-oriented architecture (SoA) to patternize applications, guide the development process, and thus develop effective SaaS. SoA is defined as a set of architectural tenets for building autonomous yet interoperable systems [46]. SoA defines eight principles that guide its development, maintenance, and service usage: abstraction, autonomy, composability, discoverability, formal contract, loose coupling, reusability, and statelessness [24]. A glance at the SoA principles reveals that SoA principles and software crowdsourcing share some commonalities. Software crowdsourcing can contribute to any software development phase(s) or can be incorporated into various conventional or agile development processes. As a result, it is favorable for the crowdsourced tasks to be abstract, loosely coupled, and independent from the underlying infrastructure and application logic in order to create SaaS with maximal features (e.g. reusability and composability).

To crowdsource, a phase of the software development process is not something new. Several research papers addressed crowdsourcing requirement engineering [41], stakeholders' analysis [58], testing, support, and maintenance [78]. Cloud-based software crowdsourcing is fairly researched too [83]. In this paper, we want to provide future adopters (e.g. crowdsourcers and crowdsourcees) and potential researchers with a thorough, systematic review of the existing status of software crowdsourcing and specifically SaaS development [52].

SaaS vs Traditional Software

SaaS aims to increase software adoption, accelerate upgrades/updates, and provide less strenuous scalability and supportability. Although all SaaS development methodologies in literature are considered adaptations of the traditional software development lifecycles with additional phases (e.g. evaluation, Subscribing, etc.), these phases are critical for SaaS success [79]. SaaS applications are developed, deployed and delivered to customers using various cloud computing related architectures (e.g. multitenancy, multi-instance, etc.) and design principles (reusability, composability, etc.). In addition to the application's functional and non-functional requirements, SaaS development possesses cloud-related requirements that can be compositional (coordination, conformance, monitoring, and QoS) and managerial (certification, rating, SLA, and support) [63]. These architectures, requirements and principles are relatively new to traditional software development; as a result, developing software with these advantageous features adds new dimensions of considerations to the traditional software development process.

It is apparent that a lack of experts is the key challenge that makes SaaS development more problematic than traditional software development. Therefore, we are seeking solutions that provide a sufficient quantity and quality of experts who can deal with SaaS complexities. Additionally, when developing SaaS for customers, developers are playing in someone else's garden and they need to play by their rules. For example, when developing SaaS for a customer in different country, challenges like compliance and understanding currency and taxation are preferably addressed by experts from the same country.

Crowdsourcing for SaaS Development

SaaS development can be crowdsourced using two different approaches. The first approach (Figure 1.a) demonstrates a software project that is defined, analyzed, and then decomposed into smaller tasks to be outsourced to the public. These tasks can be requirement elicitation, design, requirement implementation, or testing. Accomplished tasks are then gathered and integrated into a working SaaS. This approach is being adopted by many crowdsourcing services that are available online like Topcoder [80]. The second approach (Figure 1.b) involves crowdsourcing the whole SaaS development project as a single unit where public workers perform the requirement elicitation, design, development, and post-deployment phases as one task. This particular approach fits very small or tiny projects like the ones crowdsourced in Upwork [84].

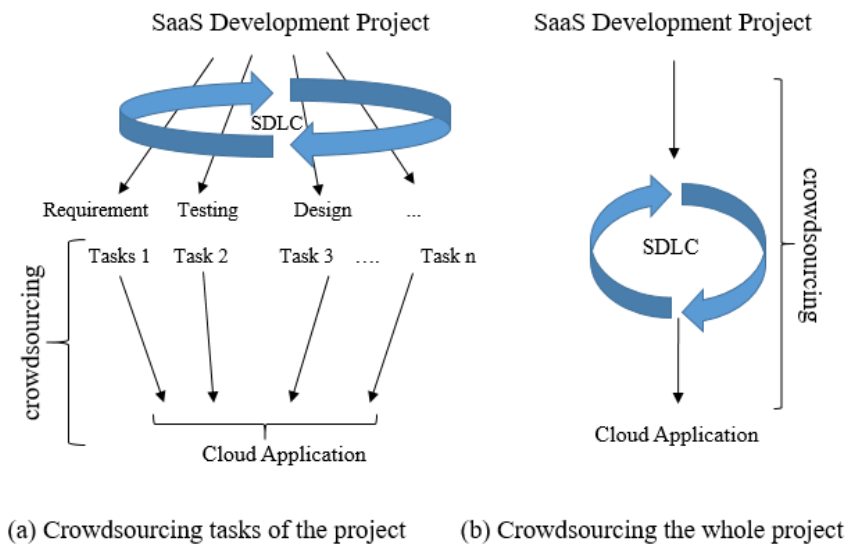


Figure 2.1: The existing approaches followed in Crowdsurcing Software

Crowdsourcing Facilities

Developing an effective SaaS can be a challenging process through centralized organizations [34, 1, 17]. In centralized organizations, all the project members are located in one location [89]. Challenges like finding proper experts, reducing project cost, and reducing project duration, etc. are common in software development projects. Researchers are still looking for contemporary solutions that could facilitate the development process and address the aforementioned challenges [89]. Harnessing crowdsourcing in SaaS development can contribute effectively toward this purpose due to the multiple efficacious facilities that crowdsourcing possesses. These facilities are illustrated as follows:

(A) Access to Pools of Global Distributed Skilled Workers

Crowdsourcing platforms are open for public participation and thus encourage creativity. Contrary to centralized organizations, crowdsourcing customers are not limited to specific workers. The Netflix Prize for example is a remarkable crowdsourcing experiment that demonstrates the usefulness of having access to minds from all over the globe [53].

(B) Access to Lower-Cost Workers

Crowdsourcers benefit from lower wages and the purchasing power of currencies in developing countries. For instance, some Asian countries offer skilled workers (software engineers in this context) with a significantly lower wage compared to North American workers who have similar capabilities [1].

(C) Temporary Hiring and On-demand Hiring

Unlike centralized organizations that are obligated to pay monthly salaries to workers who may be underutilized over a period of time, crowdsourcers benefit from the pay per task facility and they can request the service ondemand. This dynamic availability results in reducing either cost or duration of a task; thus, project managers have the advantage of sacrificing time for budget or vice verse in struggling projects.

(D) The Ability to Obtain Alternative Solutions

The crowdsourcing platform has two behavioral natures (i.e. competitive and hiring). In competitive crowdsourcing platforms, crowdsourcers only pay for the winner. In other words, there are no extra charges when a crowdsourcer receives alternative solutions. Therefore, crowdsourcing encourages creativity through engendering the spirit of competition.

(E) Direct Access to the Voice of the Customer

Crowdsourcing acts like an avenue where customers' opinions and feedback about a service can be heard. It motivates customers to share their thoughts that can contribute effectively to a software project's success. IdeaStorm, for instance, is a project launched by Dell to give their customers an avenue to share their thoughts and cooperate with each other and Dell. Customers' thoughts have generated 432 inventive implementations [44]. This facility can be utilized throughout the life cycle of the software by crowdsourcing software design, testing, and evaluation.

(F) Eliminate Experts' Bias

To eliminate expert bias, crowdsourcees are assessed based on their successful contribution to the crowdsourcing platform instead of their positions, titles, or background. Experts from the public are motivated to participate regardless of their backgrounds. Students can compete with professors to win a crowdsourcing competition. The Longitude Prize in 1714 is a good example that confirms the fact that better solutions could be produced from less experienced people [74]. The competition was to determine the longitude of ships at sea. Experts in the field worked on the problem without success; however, a clockmaker solved the problem in the end.

The following list demonstrates how the aforementioned crowdsourcing facilities can be used to cater SaaS application development.

- Adding more creativity to the development processes: due to crowdsourcing, a pool of skilled workers as well as innovative and superior solutions is becoming reachable. Software manufacturers are able to form an innovative team of workers who share a variety of best practices perspectives because of their individual backgrounds [1].

- Recruiting more skilled software engineers with a lower cost: through crowdsourcing, customers could access engineers from developing countries who cost much less than engineers with similar capabilities from developed countries [1].

- Avoiding redundant skills: because hiring workers is ondemand, customers can scale the number of workers up and down depending on their needs.

- Achieving the best possible efficiency: better efficiency can be achieved due to the ability to find lower cost, skilled software engineers.

- Having the right skills on demand: there is no obligation to provide a monthly salary for workers who are underutilized for a certain amount of time. Alternatively, workers are paid based on what they have achieved.

- Utilizing cross-sourcing: software manufacturers can crowdsource some or all independent tasks concurrently to different workers. This in turn reduces the software development time. It is true that cross-sourcing is not exclusive to crowdsourcing and it can be emphasized wherever task execution can occur independently and concurrently. However, task parallelism is limited to the number of the workers who can contribute to a task. Because crowdsourcing gives access to skilled workers globally, cross-sourcing tasks can be simpler.

- Utilizing time-zone advantages: the fact that workers in crowdsourcing can be located in different time zones could be very useful in speeding up the software development process. When a sequence of (start-to finish) dependent tasks is distributed effectively, dependent tasks can start immediately after successive tasks finish if located properly in consecutive time zones, resulting in reduced wait times that may add up to a significant duration [1].

- Meeting stakeholders' needs: client-site application development provides developers with more insight into the system requirement [19, 1]. Likewise, in SaaS development, stakeholders are located all over the world, so it would be more effective to elicit SaaS requirements from stakeholders who understand other cultures' needs.

- Improving task modularization: decomposing tasks to modules represents a substantial role in managing coordination between workers [65]. The fact that the SaaS development can be decomposed into several tasks in crowdsourcing enables crowdsourcers to effectively create independent tasks. This is achieved by dividing tasks into services that can be integrated seamlessly into different applications [35]. Afterwards, each service is assigned to a worker. A detailed explanation of this model is illustrated in the discussion.

- Reducing coordination cost: assigning tasks to workers who are globally distributed contributes positively in reducing the coordination cost [25].

- Tracking communication logs: the nature of crowdsourcing implies the absence of simultaneous communication. Because workers are distributed globally in different time zones, they can rely on electronic communication through emails or instant messengers [25]. Thus, communication history will be preserved which yields traceability and accountability [9].

- Improving documentation: distributing tasks among workers requires documentation. Documenting tasks and their statuses in every phase aims to make tasks well supported and communication between task performers unambiguous and more transparent [21, 33]. The absence of face-to-face meetings necessitates that workers document solutions in a more detailed and descriptive way.

- Defining processes clearly: in centralized application development, processes are not predominantly formalized [1]. Distributing tasks among different workers from different backgrounds raises the necessity of formalized and standardized processes to ensure consistency.

- Enhancing testability: it is more feasible to test SaaS by the crowd who are the potential users. It provides more insight into the end-user needs. Moreover, it is more cost effective than hiring a permanent local worker to test the application.

Challenges in Software crowdsourcing

Delegating software development tasks to workers in different geographical areas has become widespread and accepted as a business necessity to overcome some of the traditional software development drawbacks such as high cost [37]. Although modern communication tools are available to support task distribution, software crowdsourcing is still considered a serious challenge [33] and most crowdsourcing commercial platforms tend to highlight the success stories in software crowdsourcing with little or no attention to the challenges. In the following list, we investigate the potential challenges that could encounter in general crowdsourcing application development but with more emphasis on SaaS development.

(A) Communication issues

Application development requires substantial communication among workers and customers, especially at early stages. This communication usually occurs in two different ways: (1) formal communication, which includes decomposing tasks, responsibility assignment, and updating the application status [39]; and (2) informal communication, which occurs frequently between workers when tasks are interdependent. Because crowdsourcing workers are distributed globally, this could hinder simultaneous communication among workers due to time zone differences. Consequently, productivity also decreases [73].

(B) Language barriers

Language differences can cause project failure when application development processes are crowdsourced. Workers might misunderstand task requirements or other co-workers' communications, which may lead to failure to meet task requirements.

(C) Cultural issues

Human resources are an essential element in the software development industry. Cultural differences among human resources are also a concern. The cultural gaps can be represented in time commitment, communication manner, and behavior towards team members [74]. These differences can be critical, especially for interdependent tasks.

(D) Tasks and Workers Coordination Issues

Planning for project decomposition and distribution among workers and then assigning task dependencies can also become a dilemma. For example, task parallelism is useful when tasks are independent. However, interdependent tasks that different crowdsources have developed can lead to another type of integration challenge known as tasks incompatibility. For this, inadequate task coordination can be fatal in the application development process.

(E) Worker Collaboration Issues

Because crowdsourcing workers are distributed globally, sharing experiences, practices, tools, and decisions may not be an easy task. Managing such a collaboration technique among crowdsourcing workers is another considerable challenge.

(F) Planning and Scheduling

In crowdsourcing, tasks are usually assigned to unknown workers. This may cause a loss of control over many aspects (e.g. intellectual property). In addition, schedule uncertainty due to a lack of knowledge of the level of workers' experience is a challenge that is yet to be completely resolved.

(G) Quality Assurance

Crowdsourcing providers claim that crowdsourcing's solutions provide high quality work [71, 39]. However, in software development, it is hard to guarantee the quality because it is a relative measure [76]. Although crowdsourcing is meant to be a solution to the shortage of experts, there is no guarantee that the assigned worker can finish a task on time. In addition, crowdsourcing, like other forms of temporary employment, can be superficial work. Not only are crowdsourcers subject to an ongoing hassle from task doers (e.g. late or low quality task), but task doers (crowdsourcees) also face difficulties to guarantee being paid on time and keeping a steady stream of work. All above are enough reasons to produce low quality and quick work from crowdsourcee side.

(H) Hidden Costs

Software development projects are usually decomposed into several tasks that may be handled in a parallel or in a sequential manner. However, assigning a task to a worker does not guarantee the task's completeness and correctness. Consequently, this may result in reassigning incomplete or incorrect (unaccepted) tasks to other workers. It is worth mentioning that for tasks that are not accepted, there is no payment obligation, but development slippage does affect the application's development cost.

(I) Copyrighting and Intellectual Property

Protecting copyrights and intellectual property for a project's ideas and solutions can be a challenge. Crowdsourcing platforms are open to the public from anywhere on the globe to sign up, explore tasks, possibly participate in a task, etc. The task payment guarantee is an issue because it depends on the crowdsourcer's decision after that entity is granted the solution [29]. For that, the need for reputation systems has emerged.

Challenges as Addressed by Three Crowdsourcing Platforms

After introducing the crowdsourcing facilities' influence on SaaS development and challenges, Table 2 explains how potential software crowdsourcing services can be compared in terms of their likelihood to address these challenges. We consider three prominent commercial software crowdsourcing services, namely, TopCoder [80], UpWork [84] and Freedomsponsors [29]. We compare these crowdsourcing platforms in a tabular form, as shown in Table 2. The first column enumerates the challenges of adopting crowdsourcing for software development; the remaining columns explain how the three crowdsourcing platforms address these challenges. Some commercial crowdsourcing services may not publish enough details about their platform and the way they address challenges

Benchmarking SaaS-crowdsourcing

Like any commercial service, there has been a significant amount of work on trying to rank crowdsourcing platforms (e.g. ranker.com) [8]. Contrary to any other crowdsourcing ranking intermediaries that rank crowdsourcing platforms based on popularity, this work is to enable crowdsourcers to discover the crowdsourcing platform that best fits their SaaS development projects (if any) by identifying the platforms' capabilities and readiness to address the challenges of software crowdsourcing. The following are attributes that enable both crowdsourcers and crowdsourcees to make well-informed decisions.

(A) Behavioral Nature

A crowdsourcee needs to know the behavioral nature (e.g. competitive or hiring) of the crowdsourcing platform. In the competitive behavior, any crowdsourcee may contribute and process the task without permission from the crowdsourcer to start. By contrast, in the hiring behavior, crowdsourcers need to grant their permission to the crowdsourcee before he/she can start processing the task. In competitive crowdsourcing, a prize goes to one or more crowdsourcees who provide the best solution. However, in hiring crowdsourcing, a crowdsourcee receives a prize for solution correctness and conformity to crowdsourcer requirements. Knowing the behavioral nature of the crowdsourcing platform would help the crowdsourcee to predict the probability of winning the prize.

(B) Reputation systems

Crowdsourcing platforms need to be equipped with a means to assess workers (crowdsourcees) based on their expertise and behavior. Such information can be collected from the worker's history. In order to achieve this, a crowdsourcing environment requires a reputation system that assesses crowdsources and crowdsourcers and predicts the trustworthiness of user contributions [88].

(C) Supporting Tools

Crowdsourcing platforms vary in providing software development facilities such as ecosystems and communication/ collaboration tools. These facilities aim to support, facilitate the development process and eventually improve the software quality.

(D) Copyright Protection

A major challenge in software crowdsourcing is to protect copyrights from infringement. Ideas rights for the posted tasks should be reserved to the crowdsourcer. Likewise, unless a crowdsourcee agrees to abandon his/her rights, solution rights should be also reserved to the crowdsourcee. The main concern is to ensure copyrights are created correctly and that they preserve the right of both sides. Thus, any crowdsourcing platform needs to develop a set of solutions that guarantees copyright protection and complies with the laws and regulations governing both parties.

(E) Recommendation system

Quite a large number of tasks are posted every day in crowdsourcing platforms. For workers, finding the appropriate task that fits their skills, time, and expertise is time consuming. Similarly, crowdsourcers need to be able to learn about the task doer's ability to solve similar problems. Therefore, a recommendation system aims to shorten the distance between both parties by providing suggestions that can effectively support their decision making. This need is addressed by a matching process that is crucial to the crowdsourcing platform's success.

(F) Popularity

All the aforementioned attributes can attract more contributors (i.e, crowdsourcees or crowdsourcers). Once a crowdsourcing platform becomes known and popular, other contributors may be more inclined to utilize it because people tend to obtain/use popular services that are recommended or used by others.

Crowdsourcing Approaches

Crowdsourcing SaaS development processes can be quite empowering. However, we discuss some challenges that can be critical if not addressed. Figure 1 (a and b) shows two approaches that are widely adopted for software crowdsourcing. Table 2 shows the two approaches in Columns 2 and 3 and their preparedness to address these challenges.

As noted, when the entire project is crowdsourced, it is subjected to all challenges in Column 1 except the communication and collaboration challenges. In the second approach (i.e, project decomposition into tasks), the project is subject to all challenges. As an alternative, we propose a third approach that better suits the nature of SaaS development. This approach is a service-based decomposed SaaS. In a service-based decomposition approach, the SaaS development project is decomposed into services. Each service is then outsourced to the crowd. Implemented services are collected afterward and composed into a larger server of services. This approach is not utilized in the existing crowdsourcing platforms and to the best of our knowledge, this paper is the first time such an approach has been proposed. A service-based decomposition of a crowdsourced SaaS project is motivated by two key points: (1) SoA that SaaS applications promote and (2) crowdsourcing works better for specific software development tasks that are less complex and stand-alone without interdependencies [50]. The rationale behind this proposed approach is that every service will function separately without any interdependencies with the other services' functionality and will be independent from the underlying infrastructure and/or programming language. In other words, the output from one service can be used as input for the other services.

Figure 2 shows a crowdsourcing model that decomposes the SaaS development project into sub-services. Each service is then crowdsourced to the public throughout a crowdsourcing platforms "ecosystem". The ecosystem has facilities that guide and control the development processes, such as software development tools, project management tools, communication tools, and collaboration tools. All software development phases (i.e, requirements, design, implementation, testing, and post deployment) are performed on every sub-service. The software development phases take place after the project is decomposed into services and each service is clearly defined in terms of its input, output, and process. This model works better for both small and large-scale SaaS projects. The proposed approach should include a quality control process to ensure that tasks (services in this context) are correct and complete. The proposed model benefits from such qualities as replacing project managers with a system that can assign workers to tasks and predict tasks and project duration based on pre-collected data. In the future, we plan to build a web-based crowdsourcing platform equipped with an ecosystem and all aforementioned support tools. In addition, we plan to test and propose more SaaS development crowdsourcing models to enable crowdsourcers to easily conceptualize and plan for a software development project, elicit its requirements, and decompose it into tasks to be developed, tested, and submitted.

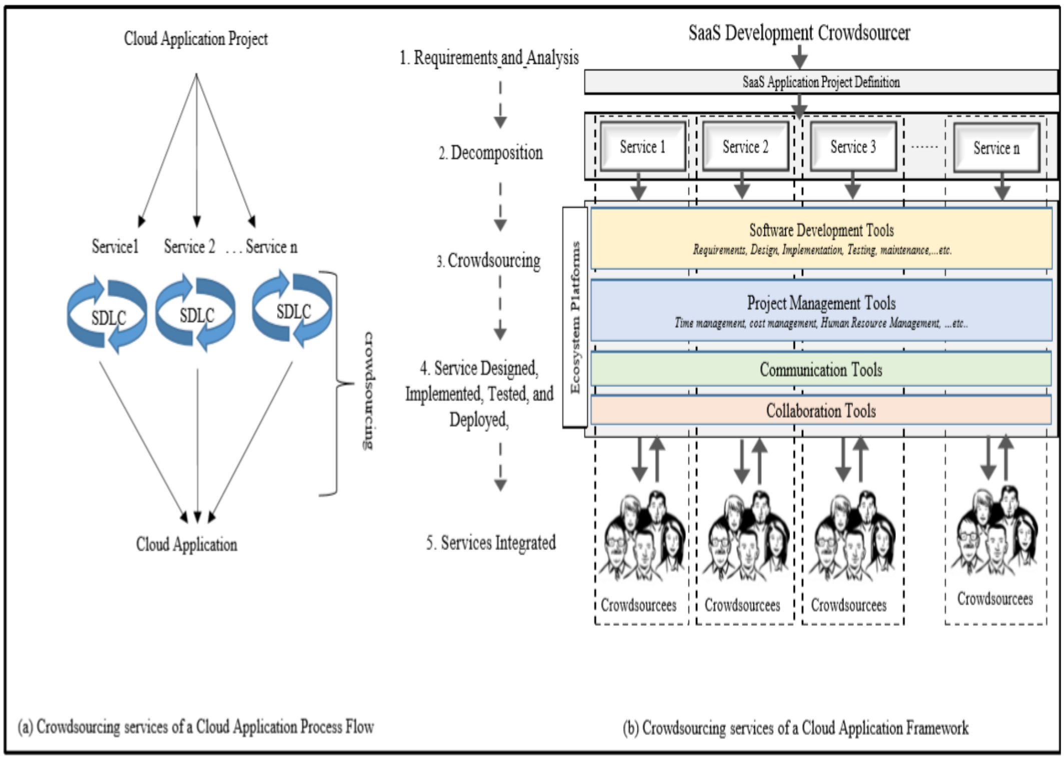


Figure 2.2: A SaaS development crowdsourcing model that decomposes a project into service

Although crowdsourcing may present solutions to overcome some SaaS development challenges, at this time, crowdsourcing software development has not yet fully matured, and we are aware of some limitations to the previously mentioned approach. For example, a crowdsourcer may need to contact the developer to perform some changes to the service after the project is delivered. This limitation, among others, leads to potential issues that need to be further investigated.

Conclusion

Crowdsourcing can be very effective for SaaS development. The facilities of crowdsourcing support building cohesive and loosely coupled SaaS if used soundly. In this chapter, we conducted a survey of one type of macrotask crowdsourcing for SaaS development.

We described the facilities of crowdsourcing and their influence on SaaS development. We also identified and described the approaches currently used in SaaS crowdsourcing and the challenges in utilizing crowdsourcing to develop SaaS. We used this list of challenges to identify the ability of existing software crowdsourcing platforms to address these challenges. Moreover, we proposed a service-based crowdsourcing approach for SaaS development and demonstrated its ability to deal with some of these challenges for large-scale projects. We drew attention to SaaS development to show the importance of macrotasks in crowdsourcing. To the best of our knowledge, this is the first study to recommend a model for macrotask crowdsourcing.

Table 2.2: Crowdsourcing Challenges as Addressed be Three Software Crowdsourcing Platforms

Challenges	*TopCoder*	*Upwork*	*Freedomsponsors*
Communication	*Avilable*	*Not avilable*	*Aviable*
Language	*Remains a challenge*	*Remains a challenge*	*Remains a challenge*
Culture	*Remains a challenge*	*Remains a challenge*	*Remains a challenge*
Coordination	*Remains a challenge*	*not a challenge bacuase the project is developed by a single crowdsourcee and not decompsed into tasks*	*Remains a challenge*
Collaboration	*Provides a communication tool to support sharing experiences and practices among crowdsoursees.*	*Remains a challenge*	*Provides a simple forum as a communication tool. It enables developers to collaborate.*
Planning and Scheduling	*Tasks are posted as a competition for all crowdsorsees. There is no risk in assigning the task for non suitable crowdsourcee.*	*Remains a challenge*	*Remains a challenge*
Quality Assurance	*Remains a challenge*	*Remains a challenge*	*Remains a challenge*
Hidden Costs	*Remains a challenge*	*Remains a challenge*	*Remains a challenge*

Table 2.3: Challenges as Applied to the Three Software Crowdsourcing Approaches

Challenges	*Entire Project*	*Project decomposed into tasks*	*Project is decomposed into service*
Communication	*Yes*	*No*	*Yes*
Language	*No*	*No*	*Yes*
Culture	*No*	*No*	*No*
Coordination	*No*	*No*	*Yes*
Collaboration	*Yes*	*No*	*Yes*
Planning and Scheduling	*No*	*No*	*No*
Quality Assurance	*No*	*No*	*No*
Hidden Costs	*No*	*No*	*No*

Chapter 3

Background and Related Work

In this chapter, we have conducted an inspection and critical study of a state-of-the-art recommendation system that is ubiquitous among crowdsourcing and other online systems.

The main objective in this chapter is to investigate various online recommendation systems by analyzing their input parameters, effectiveness, and limitations in order to assess their usage in crowdsourcing systems. In other words, how can we derive the best practices from various recommendation systems, which share some common features with crowdsourcing systems, and harness these practices to model an effective recommendation system? We concentrate on seven factors to distinguish between each of the efforts needed to achieve the objectives:

1. What parameters of the system are used in formulating the problem?
2. What is the "computational problem" formulated to make recommendations?
3. How does the formulation of the problem handle the private imperfect/ information of the stakeholders?
4. How is the problem solved?
5. How is the solution implemented to construct the recommendation system?
6. How scalable is the solution?
7. What are the limitations of the work?

We present a classification of recent works' attempts to design effective recommendation systems. Some research focuses on developing new technologies while others emphasize new methodologies. Fig.3.1 shows the topical classification tree. In the following section, we analyze the works and classify them based on their domain and their main contribution.

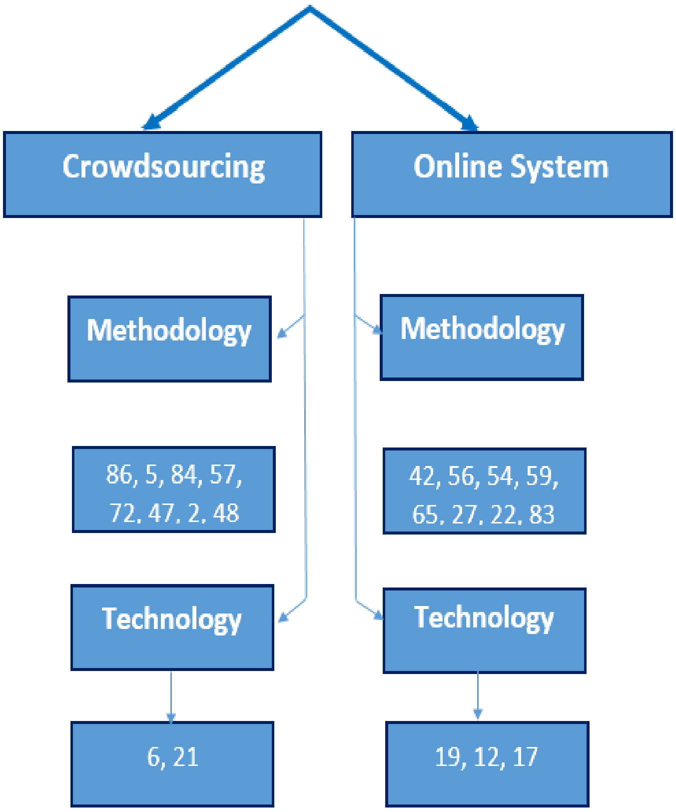


Figure 3.1: Classification Scheme

Recommendation System

In this section, we present a critical review of the works as shown in Fig. 3.1. We classify them based on their main contribution in the methodologies and technologies associated with the recommendation system. In the reviews, we present a general summary of each, identify the main contribution, and evaluate how they address the seven questions we formulated in the previous section.

General Recommendation Papers

In this section, we explore some of the online systems' recommendation systems. We tried to include diverse online systems. When we found multiple papers on a system we chose only the latest, unless others differed in strategies for efficiencies.

Methodology papers

Hwang et al. [43] built a personalized recommendation system in e-commerce applications to recommend products to customers. The system uses a hybrid recommendation system approach based on the Genetic Algorithm (GA) and a user-based collaborating filtering. Considering that each feature of the product has different levels of user preference, they applied different weights for each feature. Moreover, by using user-based collaborative filtering, they find the similarity between users to derive the recommendation decision. Actually, using the GA to provide a dynamic recommendation decision could be considered a time-consuming process. Moreover, the proposed recommendation system would overcome the new-item problem. However, they did not address the new user problem. The main contribution of this paper is employing the GA to learn personal preferences of customers.

1. Product features, customer transaction history, and customer general information.

2. Use product profiles, customer transactions, and customer general information to build user preferable profiles. Select the most similar neighbors based on the user preferable profile. Recommend products for the customer based on neighbors' selections.

3. They only take in to account the user as the main and only stakeholder. They did not take care of increases in the product sale rate, which leads to the seller benefit or increase the general sale rating which is beneficial for the service provider.

4. A product profile is presented by several features that indicate if the product has that feature. Customer product preference profile is built from the customer transaction data and the product profile. The GA is applied to find the feature weighting for a customer. Products are recommended to users by using a collaborative filtering approach.

5. There is no real experiment in this study. However, the evaluation of the system is measured by a 5-fold cross-validation approach and it uses the precision metric, recall metric, and F1-measure metric. They use a real dataset from a Telemarketing Company collected over two years. It contains 15,376 transaction records from 753 users for 239 products.

6. For a large dataset, the system would not be able to scale properly.

7. We are not sure about how the GA is effective in a real time system. What is the cost for applying such an algorithm? There is no experimentation with real life system and no solution for the new-user problem.

Lin et al. [59] built a personalized news recommendation system utilizing social experts for online news readers. The system uses a user-based collaborative filtering approach to identify experts who have influence in a particular group of readers. Incorporating that with a content-based approach is a major improvement in news recommendation systems as they proved in their experiment. Using an expert influence opinion has solved the cold start problem in the news scope. However, using an expert influence opinion is limited to the news environment because purchasing a programing book would not be beneficial to seek expert users for example. Instead, taking similar people's opinions would be more beneficial in this case. The main contribution here is recruiting the experts' opinions to solve the cold start problem.

1. Reader history, which is constructed from the story entity that is read (e.g. when or where it happened), and reader implicit rating, which reflects if the reader opens the story or not, are used.

2. From the reader history, find the reader preference profile, which indicates how much each given factor is preferable. Find the set of experts who influence the reader. Incorporate reader preferable factors with the expert opinion to recommend news stories to the reader.

3. The goal is to satisfy one stakeholder, who is the news reader. Increasing the reading behavior for specific news stories could be another factor to satisfy what was neglected here.

4. Because the news story has a short lifetime that cannot measure the user's long-time interest, user interest is represented by preferable entities, which are a set of news story episodes. Each entity is described by several factors such as who was involved, what it was about, etc. From the factors, they create a user-entity matrix. They use user news story access history to create a user-story matrix, which is a binary matrix that indicates 1 if the user read the story or 0 otherwise. User-story matrixes are sparse due to the large amount of online news stories. To solve this problem, they predict the missing story rating for each user by using a user-entity matrix. By extracting entities from each story and from a user-entity matrix, the system can predict the missing rating for the missing values in the user-story matrix. Moreover, they find sets of experts based on the reading time order for users. Finally,

they recommend news stories based on the user preferable factors and experts' opinions.

5. There is no experimentation with a real life system. However, they performed a comprehensive study on a real dataset that has been collected from several popular news web sites from July to December 2010. There were 2,015,479 visiting records for 1,192,435 distinguished accounts to 55,418 different stories. The evaluation of the system was measured by comparing the proposed algorithm with three recommendation algorithms. can scale properly because it uses model-based collaborating filtering.

6. There is no experimentation with a real life system.

Li et al. [57]'s goal is to build a personalized recommendation system in an e-commercial website to recommend products to customers. They use several social network theories to build a multi-theoretical e-commerce recommendation system. This recommendation system has an applied collaborative filtering approach. Its assessment of the similarity between users depends on applying different social network theories on the users' behaviors in the social network. In this approach, they did not take in to account users' preferable attributes, which could lead to more accurate recommendations that have been proven in many papers. Using a recommendation system that is based only on the similarities between users would not be able to solve the new item or new-user problem. Their main contribution is to design a kernel-based machine learning approach that assesses individuals' similarities.

1. All the user information is in social networks including personal information like gender, location, friendships, and behavior.

2. By deploying different social network theories, find similar users based on the users' behaviors in the social network. Recommend products to users based on a collaborative filtering approach.

3. This study satisfies one stakeholder who is the user. They did not discuss how to increase the product sales or to increase the product-purchasing rate.

4. A kernel-based machine-learning problem is based on social theory to assess individuals' similarities and then recommend products to users based on similar users' choices.

5. There is no experimentation with a real life system. However, they collect a real data set from a movie review website. The website allows users to rate and comment on movies on a scale of 1 to 10. The tested

records include 452 movies and 100 reviews by 6,155 reviewers with at least one friend. The evaluation of the system is measured by comparing the proposed algorithm with a trust-based and collaborative filtering approach by adopting the root mean square error measurement.

6. It can scale properly because it uses model-based collaborating filtering.

7. There is no experimentation with a real life system, they neglect the content-based approach in order to recommend more accurately. Moreover, there is no solution for the cold start problem.

Meehan et al. [62] proposed a personalized tourism recommendation system to recommend attractions to user. A hybrid approach has been used by combining context information to extract information about location, weather, and time as well as personalized information with collaborative filtering by extracting negative and positive tweets for each attraction, which will change the attraction priority based on the Social Media Sentiment. The main contribution in this study is adding the Social Media Sentiment factor. The cold start problem can be addressed by recommending attractions based in location, weather, time, and Social Media Sentiment.

1. User location, current weather, current time, Social Media Sentiment, which is negative and positive tweets, as well as personalized information, such as age, gender, and marital status are used.

2. Use user location, current time, and current weather to decide which attractions are more favorable in a current state. Use tweets about certain attractions from a tweeter to sorts the priority for each attraction. Combine the aforementioned factors to produce the appropriate recommendations.

3. The proposed recommendation system has addressed one stakeholder, who is the user, and did not raise the visiting rate of an attraction or increase the prosperity of tourism, which could be a service provider benefit.

4. Extract user-personalized information from social media if users have a social media account; otherwise, they explicitly ask the user to fill out personal information. They use GPA, GSM, and Wi-Fi to get the accurate location and WorldWeatherOnline API to get the weather information. They use Alchemy API to analyze tweets if it is a positive, negative, or neutral tweet. Then, an artificial neural network could denote each factor as a node and decide a proper weight for each factor. Finally, they list the recommended attractions.

5. The study is still in progress without a real description about how to solve the problem in detail.

6. From the proposed factors, the system can scale in the real implementation, but whether it can still associate tweets in this system or how tweets will be integrated to the system is not clear.

7. This study is still in progress with no detailed explanation or experimentation with a real life system.

Rong et al. [69] proposed a recommendation system for an e-commerce website that focuses mainly on solving the cold start problem. The user-based collaborative filtering approach is used to recommend products to users. The main problem to solve is to predict the user rating for different items. This problem could be addressed by a collaborative filtering approach if the user has sufficient history. However, if the user is new, the prediction accuracy is decreased. Some studies have used social information to obtain information that could contribute in solving the cold start problem. Nevertheless, social information is not always available, which means they have come up with a new approach to solve the cold start problem using only user history rating information.

1. Rating history for each user.

2. From users' rating history, find the similarity between a target user and other users to predict the missing elements' rating for the target user.

3. The proposed recommendation system has addressed one stakeholder who is the user and did not raise the selling rate of an item or increase the general selling activity that could help a service provider.

4. The system uses a rating matrix to find the similarity between a target user and the other users. In case the user has a very limited rating history, the system uses an undirected, weighted bipartite graph to find similar users. Undirected, weighted bipartite graphs have two sets of vertices: users and items. They define a random walk to find the similarity between users using the Monte Carlo algorithm. There are two types of walk: from a user to a previously rated item and from an item to another user who has the most similar rating for that item. Essentially, from user , they find the most similar user , who has the closest rating to the target user, and they fill in the gaps of the missing rating in the matrix of . Then from user B, they find the most similar user , they and use his rating to fill in the remaining gaps in matrix A and so on. Finding the similarities between users is a pre-computed process.

5. The test of the proposed recommendation system was on 5 real datasets from MovieLens, Epinions, Bookcrossing, Amazon2, and Yahoo! Music using 4-fold cross validation to reduce the influence of sampling. They use 10% of the previously rated items by a user and predict the remaining 90%. They then use Mean Absolute Error to assess the prediction quality.

6. It could scale properly because it is model-based collaborating filtering.

7. There is no experimentation with a real life system. They did not use a content-based approach as a factor to recommend more accurately and overcome the new-item problem.

Feng et al. [28] proposed a recommendation system that could work in many areas such as e-commerce websites and Movie recommendations. This recommendation system recommends products or movies to users by enhancing an item-based collaborative filtering approach by using K-means clustering. The main approach in this study is to overcome the collaborative filtering drawbacks. In a pure item-based collaborative approach, the performance is reduced as datasets increase. K-means clustering itself has a high performance rate even in a large dataset. However, K-means clustering would not be beneficial in the cold start problem. Cooperating K-means clustering with an item-based collaborative approach could address the previously mentioned shortages, which is the main contribution in this study.

1. The rating history for each user is used.

2. From users rating history, find the similarity between items. Recommend items that meet the user's preferable attributes, which could be inferred from the rating history. Overcome the scalability barrier by using collaborative filtering approach with K-means clustering.

3. The proposed recommendation system has addressed one stakeholder who is the user and did not raise the selling rate of an item or increase the general selling activity, which could help a service provider.

4. Firstly, they generate an item-based collaborative filtering result. Secondly, they generate a K-means clustering result from the collaborative filtering result. Finally, they combine results by using a Simulated Annealing algorithm.

5. They use a real dataset from MovieLens that contains 100,000 movie ratings rated by 943 people for 1,682 movies. The evaluation assessments used the average payoff value, the standard deviation, the precision rate, the recall rate, and the running time of the algorithm.

6. It could scale properly

7. There is no experimentation with a real life system.

Dror et al. [23] proposed a multi-channel recommender mechanism for the Yahoo! Answers system to recommend questions to users. A hybrid approach has been constructed using content based from all available signals that are derived from the questions, user preference profiles, and collaborative filtering using similar users' actions. This approach can be useful for Community Question Answering sites in general.

1. Question attributes, which include three classifications: textual, categories and user IDs who interact with the question, were used. Additionally, user properties that include the three aforementioned classifications for each user from the questions answered before plus users' explicit preferences attributes were used.

2. By using question user attributes, predict which users will answer a question. Handle it as a classification problem. Train a Gradient Boosted Decision Trees classifier using logistic loss.

3. In this work, the proposed recommendation system addressed the three main stakeholders: the answerer, the asker, and the service provider. The three stakeholders' satisfaction is met by default when recommending more related questions to the answerer.

4. They proposed a Multi-Channel Recommender system model that first maps questions and users to their attributes. The question attribute is described by a matrix. The columns of the matrix are textual token "words" that have been extracted from the question, category, and user. The rows describe the attributes. For example, (title, football) has the value 1 if the term football appears once in the title. For another example, (best answer, id) is the id for the user who has the best answer for that question. User attributes are driven from question attributes for the questions that the user has interacted with as several channels. Each channel describes the nature of interaction such as asker, best answer, voted to answer, etc. Each channel is represented by a matrix that describes the question attributes and the preferable attributes. Pairing each question attribute with each user attribute creates multiple features, which are used by a classifier with actions of other similar users to evaluate the match between the user and the question.

5. There is no experimentation with a real life system. They create a large-scale dataset consisting of 1,256,262 examples with an equal number of positive and negative examples. The set had 169,392 unique users. However, as they did not mention the source of the data set, we assume that data set was not real data. The evaluation of the

system is measured by calculating the accuracy and the Area Under ROC Curve (AUC) on test examples, where accuracy is a standard metric for classifiers' performance and the AUC metric measures the probability that a positive example is scored higher than a negative example.

6. It can scale properly.

7. There is no experimentation with a real life system.

Yan et al. [90] proposed a personalized recommendation system for a community question-answering framework to recommend questions to an answerer. The recommendation system is designed by using a content-based approach. The system extracts topics' distributions from the questions, which is similar to tags. Then they rank the different relations between asker, topic, and answerer in the offline phase. They recommend the new question to the highest corresponding ranked answerer who has answered questions on that topic by that asker in the online phase.

1. Transaction history is used.

2. By using the set of questions, extract a set of topics. For each question, find topic distributions. From the transaction history, find the relationship between asker, topic, and answerer, and rank these relationships. For each new question with its distributed topic, choose the most related topic and find an answerer who has answered this kind of topic.

3. In this work, the proposed recommendation system addressed the three main stakeholders: the answerer, asker, and service provider. The three stakeholders' satisfaction is met by default when recommending more related questions to the answerer.

4. Firstly, they train the system to construct question topics by using a Latent Dirichlet Allocation (LDA) model, which is well known in IR. Secondly, they extract the relationship between asker, question, and answerer from the transaction history. Thirdly, they alter the previous relation to be among asker, topic, and answerer by using the Tensor Factorization model. Fourthly, they rank the previous relation by the strength of the correlation, which means the answerer a, who answered topic t by asker u, is ranked higher than the other answerer is. Finally, it recommends the highest ranked answerer for the new question.

5. There is no experimentation with a real life system. However, the designed system has been tested by using a real dataset from Yahoo! Answer and Tencent Wenwen. It uses four metrics to evaluate the

performance: Mean Reciprocal Rank (MRR), Mean Average Precision (MAP), Precision at rank N (P@N), and Recall at rank N (R@N).

6. It could scale well under a limited number of topics. However, as the number of questions, askers, and answerers increases in the training set, the performance will increase as well, and the training time will increase correspondingly.

7. The proposed system did not use hybrid approach, which has proven to outperform the content-based approach and could give options that are more diverse. There is no experimentation with a real life system.

Technology Papers

Davidson et al. [20] proposed a personalized recommendation system in YouTube to recommend videos to users. They used a content-based approach by extracting input details from the users watched videos list to recommend a new list based on the extracted information. The nature of user-video interactions is different from another user-movie or user–product interaction because it is more diverse, which makes the interaction noisy. Another challenging factor is that YouTube videos sometimes have a poor metadata description. To address the aforementioned challenges this study proposed a recommendation system that balances the user's specifications and the diversity, which is its main contribution.

1. Watched videos' attributes, such as video streams and video metadata are used. User activity data is divided into explicit, such as rating, and implicit, such as assessing users' watching activity. For example, the user has started watching the video or the user has watched a large portion of a video.

2. From users' histories, construct a seed set for each user that contains a set of the watched videos. Using the seed set, build a candidate set of videos that fit the user's preferable attributes and meet the diverse condition to recommend it to the user.

3. The proposed recommendation system has addressed one stakeholder, who is the user, but did not raise the watching rate for the video or increase the watching general activity, which could lead to the service provider benefit.

4. To have a candidate set from the seed set videos, they need to map each video in the seed set to related videos using co-visitation counts. Considering a group of sessions in a period, they count how many times each pair of videos has been watched within sessions. Then, they can get the related videos to each video in the set ranked by their

co-visitation count. They then expand the candidate set by taking a limited transitive closure over the related videos to make the candidate set more diverse. By setting a limited number to find related videos for each seed video to insure diversity and remove too similar videos, they rank the candidate videos using three signals: video quality (e.g. popularity), meeting user specification from the user watched history, and diversity. Finally, instead of recommending the most relevant videos, they chose a diverse video from the candidate set.

5. The system works as a feature on YouTube's home page using a pre-computation approach. The data set is updated several times per day. The recommendation system works as 1) Data collection, 2) Data processing and then save it in a table. 3) Recommendation generation through a series of Map Reduces computations, and 4) Recommendation serving. Comparing the recommendation system with other algorithms, such as most watched videos and top ranked video, the recommended videos were more watched than other video lists.

6. Scalable.

7. Because the recommendation generating is pre-computed, there will be a delay between generating the recommendation and producing it to the user. The delay will not be severe due to several updates during the day. There is no solution for the new-user problem. Still the recommended videos will be narrower than if collaborative filtering were used.

Chen et al. [13] designed a personalized recommendation system to recommend products to customers in small retail websites. The aim of this recommendation system is to fit the limited resources in small retail websites using a content-based approach and association rules mining. Different prediction criteria were extracted from each product and weighted depending on the user criteria assessment. The main contribution in this study is to build a recommendation system that fits the modest resources with good performance. The cold start problem could be addressed by considering association rules mining criteria to recommend products that users may like.

1. Product data, user information, and current product selection were used.

2. Use products' data to extract the products' categories. Use the previous user's purchasing to extract user preference categories. From each product purchase, extract the custom country to assess the popularity of a certain category in different countries. Associate each

product category with purchasing month in order to assess the category's popularity in a certain month. Review all the previous factors to recommend the most likely product the user is willing to purchase next.

3. The proposed recommendation system has addressed two stakeholders who are the customer and the service provider by increasing the purchasing rate. However, it did not raise the purchasing rate of a product that benefits the seller.

4. By applying four algorithms, each constructs a certain relationship, produces a list of recommended categories, and saves it in separate tables. The first algorithm is used to assess the relationship between the categories from the transactions information by applying an Apriori algorithm. The second algorithm is used to assess the relationship between the category and the country by counting the frequency. The third algorithm associates the customer with the preferable categories by counting the frequency from the purchasing history. The fourth algorithm associates the category with the month by counting the frequency for each category in different months. From each table, the top five categories are selected. How each category is weighted depends on its importance. The category with largest value is selected.

5. There is a real implementation for an online retailer website. To find the relationship between the categories from the transactions, they use the 'arules' library in R to remove duplicate category ids within the same orders. The 'plyr' library in R was used to do a frequency count of category ids by month, country, and customer id. Then, a PHP interface was used to queries on the tables. PHP functions are used to handle the calculation in the model layer of any ecommerce site, which uses an MVC framework. The dataset used in the training process contains 4,000 records.

6. The proposed system works just for a small dataset.

7. The recommendation output is a product category instead of a specific product, which could be very broad.

Cosley et al. [18] presents SuggestBot to recommend editing tasks in Wikipedia to members. SuggestBot uses a hybrid approach using similarity of text, connection links, and connection through co-editing activity. They earmark the existing recommendation techniques to suit the nature of the Wikipedia system. The main contribution is the experiment with a real life system of SuggestBot and the valuable result from applying the recommendation system.

1. Users' editing history was used.
2. Using the user editing history, find similar articles based on the text similarity that most likely will met user interests. Using the explicit connections through links in the edited articles to find related articles, find similarities between users' histories to recommend articles for similar users. Based on the three aforementioned factors, identify the articles that would be most interesting to the users.
3. In this work, the proposed recommendation system addressed the three main stakeholders: the answerer, asker, and service provider. the three stakeholders' satisfaction is met by default when recommending more related questions to the answerer.
4. A Jaccard metric for set similarity between profiles, SQL quires to measure text similarity, and explicit connections through links are used.
5. It is implemented by building a recommendation tool for real communities with real users in Wikipedia's website using MySQL 4.1's built-in tool.
6. It scales well and task routing will grow more valuable as the number of tasks increases.
7. There is no solution for the new-user problem.

Crowdsourcing Recommendation Papers

In this section, we explore crowdsourcing recommendation papers. We searched Google Scholar for recommender systems in crowdsourcing, recommendation system crowdsourcing, and Task matching. Moreover, in each paper, we looked over the references to find related studies.

Methodology papers

Yuen et al. [93] proposed a recommendation system to recommend tasks to workers in Amazon Mechanical Turk (Mturk). The proposed system uses only the matrix factorization approach by extracting the user's preference tasks from both a worker's performance history and a worker's task searching history. In a worker's performance history, they retrieve information about the worker's ability of performing different kinds of tasks such as the number of browsed tasks, the number of selected tasks, and the number of completed tasks. In a worker's task searching history, they gather information about the relationship between a worker and a task by analyzing the worker's task interaction. For instance, if a worker has browsed the task information, it indicates the worker may be interested in similar tasks. If a worker has completed the task, the worker may have the ability to complete similar tasks and so on. They assessed the workers' task preferences on a 5-point scale: a worker 1) did not browse the task, 2) browsed the task; 3) worked on the task 4) completed the task; and 5) accepted the task. Then, they used the matrix factorization to estimate the missing values for the worker-task preference scale. The main contribution is adding a worker task searching history.

1. Task features such as title, time allotted, reward, and expiration date were used. A worker's performance history features such as number of browsed tasks, number of selected tasks, number of completed tasks, number of accepted tasks, and percentage of accepted tasks were used. Five worker task searching history features were used: 1) a did not browse the task; 2) browsed the task; 3) worked on the task; 4) completed the task; 5) and accepted the task.

2. By using a worker's performance history and a worker's task searching history, predict the missing values in the worker task matrix.

3. The proposed recommendation system has addressed one stakeholder, who is the worker, and neglects the requester and service provider benefits.

4. It is solved by using Probabilistic Matrix Factorization to recover the worker-task matrix.

5. No experimentation with a real life system on the proposed recommendation system was used. However, to prove that the searching history is a major factor for workers to determine the next task, they post a survey as a task for 100 workers and ask the workers about how much they would prefer to work on a similar task. The result of the survey shows that the workers would prefer to work on a task similar to what was accepted before.

6. Scalable.

7. There is no experimentation with a real life system. The proposed recommendation system cannot solve the cold start problem.

Ambati et al. [5] proposed a recommendation system to recommend microtasks to workers in a crowdsourcing platform. The recommendation system is designed based on a content-based approach. The main objective in this study is to overcome the drawbacks in the existing crowdsourcing platform, which can be summarized in five points according to the paper: 1) workers cannot find the appropriate task on time before it is allotted to other less suitable workers; 2) noisy output from less skilled workers; 3) reducing a task's reposting for further judgments; 4) the platform's reputation, which could be affected by the number of rejected tasks; and 5) lack of trust between worker and requester. They claim that the proposed recommendation system would address the aforementioned drawbacks. However, the real contribution in this paper is designing a recommendation system based on worker performance and interests to recommend tasks that will mostly be accepted by the requester.

1. User profiles, worker explicit feedback about the task, implicit worker feedback which is user-task interaction were used. Task details, such as task descriptions, reward associations, number of associated hits, and timestamps, were also used. Moreover, they used the requester feedback whether the task was accepted or rejected.

2. Use worker profile information, worker explicit feedback, worker implicit feedback, task details, and requester feedback to learn user task preference.

3. The proposed recommendation system has addressed one stakeholder who is the worker by recommending tasks that meet the worker's interests and skills. However, increasing the output quality, which leads to the requester's benefit, was not addressed. The output quality could be decreased by assigning tasks to busy workers even if they have the appropriate skills.

4. Two different approaches have been proposed to learn user task preferences: a Bag-of-Words approach and a classification based approach. In a Bag-of-Words approach, they use worker history information to extract the previous task features and the the accocited scale of preference.. In the classification-based approach, the system uses binary classification 1,-1, which indicates if the user has completed that kind of task or not. Each task is represented by a set of weighted features based on the user history. Then they use one of the learning preference approaches to rank the available task and to recommend the top few tasks to the worker.

5. There is no real implementation for the proposed system. However, they collect a real dataset from Amazon MTurk to evaluate the proposed system. From the dataset, they find 24 workers who have worked on tasks from 10 different kinds of interest. For each user, they use half of the worker's interest to train the system. Then, they rank the other half of the interest. If the recommended task has been completed by a worker, that confirms the system effectiveness.

6. Scalable.

7. There is no experimentation with a real life system. The recommendation system is designed mainly based on a content-based approach, which has several limitations as we mention previously. Moreover, the proposed recommendation system cannot solve the cold start problem.

Yuen et al. [91] proposed a task recommendation system to recommend tasks to workers in a crowdsourcing platform. The system uses a content-based approach to match workers with suitable tasks. The main contribution in this study is to assist workers to find their suitable tasks and to improve the quality of the output. The system recommends tasks based on the user's interest and performance. Moreover, they suggest that each task has to be classified under one category before posting it in the platform.

1. The worker's performance record which contains the previous task selection preference is used. Task selection preference includes task category, reward and time allotted. In addition, task acceptance rate is used to infer the worker performance in each category.

2. Use worker's performance records and worker's task acceptance rates to rank all the available tasks based on the best matching for worker's interest and performance.

3. The proposed recommendation system has addressed one stakeholder who is the worker by recommending tasks that matched the worker's interest and his previous performance. However, the other stakeholders who are requester and service provider have not been addressed as we have mentioned in the analysis of the previous study

4. Each worker has performance record, which has four values: 1) tasks acceptance rate in each category; 2) task category preference score; 3) reward preference score; 4) time allotted preference score. All these above information extract from the worker selection history and task accepted rate. Then based on the worker performance record, rate all the available tasks. Then lists all the available tasks based on the best match. After each completed task, update the worker performance record.

5. There is no experimentation with real life system. However, the evaluation for the proposed recommendation system was by applying case study involved 12 participants and 4 task's categories. Each category has 10 tasks. Then, they asked the participants to work on 10 tasks out of 40. To evaluate the proposed system, they used Mean Absolute Error to compare the task's user rate from the experiment and the task' user rate that generated by the proposed algorithm.

6. The system could face a major delay to rank all the available task for each worker.

7. There is no experimentation with real life system. They used only content based approach which has a major limitation that has discussed earlier in this paper. There is no solution for cold start problem.

Lin et al. [60] proposed task recommendation system in crowdsourcing to recommend tasks to workers. The system uses matrix factorization collaborative filtering approach. The system uses implicit signals to predict the positive and negative task's rating for each user. The goal for this approach is to recommend tasks that the user did not work on them before but these tasks will belong to the user's interest by modeling the workers' preferences based on their behaviors. The main contribution in this paper is efficiently model a predictive system by incorporating negative implicit feedback.

1. User history, which contains user-task interaction is used.

2. Use user history to infer the task's positive or negative rate for the available tasks. Then, recommend the tasks with the highest rating value.

3. The proposed system address one stakeholder who is the worker by recommending tasks that meet the worker's interests and neglect the requester and service provider benefits.

4. The positive feedback has been assessed based on how many times worker has been perform that kind of task. The negative feedback based on the task availability. On other words, if a task has a high availability and the user did not choose to work on that task, this task will have negative rating. However, if the task availability rate is low, the negative rating will be minimize. Then based on the positive and negative rating, recommend tasks by multiply each predicted rate with task availability in the training set to produce predicted throughputs, and sorts tasks accordingly.

5. There is no experimentation with real life system. However, the effectiveness of the proposed system has been evaluated using real dataset that collected from Microsoft's internal Universal Human Relevance System (UHRS). The dataset includes 17,000 users and above 2,100 tasks. Then, they compare the proposed system with other two approaches which are neighbor based and task popularity based.

6. Scalable.

7. There is no experimentation with real life system. Even though collaborating filtering has been proved in several studies to outperform content based, Hybrid approach could outperform the two approaches. There is no solution for cold start problem.

Yuen et al. [78] proposed a recommendation system in crowdsourcing to recommend tasks to worker. The system use matrix factorization collaborative filtering approach. The objective of this study is to solve the cold start problem. The user rate implicitly inferred from user-task interaction history. The main contribution in this study is considering the dynamic scenarios for the new worker and the new task in the recommendation system.

1. Worker history and tasks' categories are used.
2. Use worker's history to infer the worker's preferable tasks. From the task, extract the task category. Use worker's preferred tasks to extracts worker's preferable categories.
3. The system satisfies one stakeholder who is the worker by recommending tasks that matches his interest and skills. However, the system neglect the requester and service provider benefits.
4. The values in worker-task preferring matrix factorization indicate the user rating for that task. The rating inferred from the worker-task interaction history which is range from 0-5 as described in study [6]. Then, they apply matrix factorization to generate two other matrixes which are worker-categories, and tasks-categories. Then, they recommend that tasks with the categories that matches the user's categories interest.
5. There is no experimentation with real life system. However, they evaluate the work by using the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) for matrix comparison. They compare their approach with the state of the art approach.
6. Scalable.
7. There is no experimentation with real life system. Their main goal is to address cold start problem. However, new user problem is still being an issue in this system. Moreover, new item that belong to new category is also still an issue.

Kang et al. [49] design a task recommendation model for crwodsourcing platforms. The task recommendation is based on workers preferences and reliabilities. They designed learning framework based on multi-armed-bandit to learn the worker reliabilities and preferences in each task category. basically, the worker preferences and reliabilities is obtained from speical kind of task called gold task that is not associated with rewards and the answer is known in advance. The preferences is measured by worker acceptance for that kind of task, and the reliabilities is measured by the solution correctness. Then, they recommends tasks based on greedy algorithm. However, the assumption is not realistic because they assume that the worker preferences and reliabilities are independent. The main contribution is they formulated the task recommendation problem in crowdsourcing as a MAB problem.

1. tasks categories, workers preferences, and workers reliabilities.
2. using gold tasks from different categories and recommend it to the worker to acquire the user preferences and reliabilities.
3. They only take in to account the user as the main and only stakeholder. They did not take care of increases the solution quality.
4. based on the worker reaction toward the recommended gold task. if the worker accept to work on the gold task from a certain category, this category will be added to his preferences. the reliabilities is measured based on the solution correctness and then recommend task following greedy algorithm.
5. they perform a simulations using different settings with synthesized dataset.
6. It could scale properly.
7. There is no experimentation with a real life system, the model designed based on non realistic assumption.

Alamer et al.[2] propose location privacy aware task recommendation that protect workers location in spatial crowdsourcing during task recommendation using encryption techniques. The main contributions is this study is designed privacy-preserving location matching mechanism by encrypting the user region information.

1. worker region, worker solution, tasks.
2. recommend tasks to the workers who are in the required geographical area while protecting the workers location.
3. The goal is to satisfy the worker in term of privacy matter only and satisfy the requester by increase the solution quality.
4. The tasks recommended to the workers located in the a specific geographic regions. Protect worker location against malicious attackers using by encryption.
5. The experiment was mainly designed to count the computational overhead and communication overhead which increases linearly.
6. Not scalable.

7. The only contribution here is the confidentiality. There is no new contrition in the recommendation techniques.

Karim et al. [50] design software development crowdsourcing recommendation system that combine learning strategy and earning strategy based on analyzing the worker preferences and worker history. The two strategies of the search are combined with a preference choice for the worker towards one or the other strategy.

1. Worker preference value for learning, worker history.
2. Using the worker current active status, worker preferences, and worker preferable search strategy, recommend task based on the worker winning chance prediction on each task.
3. They only take in to account the worker as the main and only stakeholder.
4. Extract all the completed task and the associated registered workers, winner workers.Find out active workers, new, and on going tasks. Then, for each active worker, the system predict the winning chance for each task based on combining the calculated learn and earn score. Then rank the tasks for each worker and recommend the top 10 tasks.
5. The solution implement as a prototype in TopCoder platform and evaluated by a survey for worker satisfaction for the recommended task.
6. Scalable.
7. The recommendation had address well the balance between learning and earning. However, it would be better if each user can choose his priority of searching techniques.

Sun et al. [77] has design a spatial crowdsourcing recommendation model for online delivery route recommendation. The goal is to maximize a single worker income under online scenarios considering three influences factors: (1) the distance between the worker location and the start point for each task, (2) the distance between the start point and the destination for each task, (3) the possible future demand that start and each task destination.

1. The worker location, the tasks starting location and ending location, each task ending time.
2. Use the worker location, visible tasks start and ending point, to recommend the best set of tasks that maximize the worker income.
3. The proposed recommendation system has addressed one stakeholder, who is the worker.

4. The model recommend the tasks that distention point has more potential starting route requests with the route length consideration to maximize the worker income.

5. experiment on synthetic and real-world datasets.

6. The system is scalable.

7. The recommendation model is efficient for scheduled routing request. However, most of route request are dynamic and if we recommend a set of task to be done in order, the waiting time for the requester could be increased.

Technology papers

Basak et al. [6] has built a framework, which can be used as a tool to test tasks recommendation algorithms for crowdsourcing. The system has task creation and management interface. Moreover, it consist of three module: workermodeling module, task-modeling module, and task recommendation module. The recommendation module supports different recommendation techniques such as item-based, and user-based nearest-neighbor. The system offers task recommendation features based on set of worker's and task's properties. In addition to the known task properties, they add additional description categories which are: 1) task media type; 2) task operation type; 3) task topic. Workers properties could be formed by using three techniques: 1) explicitly from external sources; 2) implicitly from the user history; 3) requester feedback. The framework enable researchers to customize the framework using plugin extensions. The main contribution in this paper is build a framework to test different recommendation algorithms for crowdsourcing.

1. Worker-modeling module, a task-modeling module, and a task recommendation module are used.

2. This paper present a framework to facilitate recommendation algorithms test for crowdsourcing.

3. TThe paper didn't add a new methodology for recommendation system. Their contribution was in building framework to test different recommendation methodology.

4. The framework has task creation, and management interface to enable researcher to control the framework. There is three main modules: 1) worker-modeling module; 2) task-modeling module; 3) task recommendation module. Workers and tasks can be modeled using different properties.

5. To test the framework they implement an experiment for three algorithms. 1) feature-independent. By implementing user-item matrix. 2) feature-based. By implementing user- feature matrix. 3) Composite. Which combine user-item matrix and user-feature matrix. They created a database with 70 tasks from 10 categories. 24 subjects have been participated in this experiment.

6. The framework support a wide variety of recommendation algorithms.

7. The framework has a real potential influences in the recommendation system algorithm testing.

Difallah et al. [22] proposed different task recommendation approach for crowdsourcing based on push methodology instead of the currently used pull methodology. The proposed system use content based approach. The idea of this work is to use social media website Facebook to acquire users' skills and interest. This information gathered from the pages that the user liked and the tasks that he previously completed. In current task recommendation system all tasks get posted in the platform and wait for users to work on them. Contrary to the previous approach, tasks in the proposed system posted on the related worker's page. By doing so, unrelated workers will not be able to give unqualified answers. The main contribution in this paper is the automated pushing mechanism to increase answer quality.

1. User history, which contains liked Facebook pages and completed tasks, is used. Hits description is also used.

2. User history is used to build each worker's profile, which contains the worker's skills and interests. For each submitted task, they decompose several microtasks and assign them as a hit with a specified monetary reward. Each liked page is linked to an entity in the Linked Open Data (LOD) cloud to categorize them. Then they match users with hits based on the similarities between them.

3. The proposed system addresses one stakeholder (i.e., the requester) by increasing output quality. The system partially satisfies the workers by assigning related tasks to them. However, the workers may need to work on more tasks to gain more rewards. The service provider benefits could be decreased by the reduction in the number of tasks that could be performed.

4. Each hit has a textual description, a data field, a set of candidate answers, and a list of Facebook categories. Each worker profile is represented by two factors: the liked pages categories and the completed task categories. Then for each hit, all workers are ranked based on the matching between the hit and the user. Finally, the hit is posted to the top matching users.

5. Facebook App called OpenTurk is used to push hits to the Facebook users as described above. A total of 170 workers contributed to this experiment to work on two hit categories. The selected workers had 12,000 liked Facebook pages. The system uses cloud-based storage and back-end processing to enable scalability when the number of users and requesters increases.

6. The system is scalable.

7. They only used a content-based approach with no solution for the cold start problem.

Conclusion

Our survey has explored major state-of-the-art online systems and crowdsourcing system. Because crowdsourcing systems are an emerging field, the related recommendation papers are limited [**86**, **5**, **84**, **57**, **72**, **2**, 29, 8]. Therefore, we expanded our search to papers that proposed recommendation systems for different kinds of online systems to highlight the potential of the best approaches, which could be applied in a crowdsourcing system.

However, crowdsourcing systems have some important features, such as monetary rewards and task deadlines, which do not belong to some online systems, such as Wikipedia [27] and Yahoo! Answers [41]. Therefore, we have covered other recommendation systems that address online systems featuring other common factors, such as the monetary outcome in e-commerce systems [**51**, 33]. Our research shows that most recommendation systems address one stakeholder, such as [**51**, **56**, **62**, **81**, 33, 63, 80, 36]. However, in some systems, satisfying one stakeholder leads to holistic satisfaction, as in [**79**, 41, 27]. All crowdsourcing recommendation papers addressed only one stakeholder, either the worker or the requester.

To the best of our knowledge, no previous papers have considered the satisfaction of all stakeholders. Designing a recommendation system that achieves stakeholders' gratification would be a great opportunity for effective crowdsourcing. Moreover, we did not find any paper that used a neighbor-based collaborative filtering approach or a hybrid approach, which could outperform existing methodologies.

Chapter 4

Goals and Contribution

Goal

In this dissertation, we assume that every stakeholder (worker, employer, service provider) acts selfishly in order to maximize profit. Because of this assumption, we have presented a mechanism design based on a multi-objective recommendation system to achieve holistic satisfaction of all stakeholders by matching the worker with a suitable task that fits the worker's skills, increasing the worker's rewards and rating, giving employers more qualified solutions at lower cost without affecting their rating, and raising the task acceptance rate, which will increase aggregated commissions accordingly.

Mechanism Design

A mechanism design is a set of rules for economic activities. It starts with a desired outcome in mind and tries to create a mechanism that allows users to reach this outcome. When we design the mechanism, we cannot control the players or users' behavior or decide what they should care about. This information that we cannot control is called the settings. The settings are described in the tuple (N, O, θ, P, U), where:

- N is a finite set of n agents or workers.
- O is a set of outcomes or tasks.
- $\theta = \theta_1 * \theta_2 * .....\theta_n$ is a set of possible joint type vectors, which consist of the players or the workers' private information, such as their rating or previous performance.
- P is a (common prior) probability distribution on θ.
- $U = (u_1, ..., u_n)$ where $u_i : O * \theta \to R$ is the utility function for each player.

Given the outcome (tasks) and the type of workers available , the mechanism is designed to optimize the workers' utility function. The mechanism is a pair of actions and map (A, M), where

$A : A_1 * ....A_n$, where A_i is a set of actions available to the player or the worker $i \in N$.

$M : A \to O$ maps each action profile to the outcomes.

Thus, we can specify:

1. The action set for the workers.

2. The mapping to the outcomes, over which workers have utility (based on the recommendation algorithm).

The goal here is for workers and employers to behave in a certain way, reach certain outcomes that maximize revenue, and reach those outcomes while not having control over the settings. The trick is to set up the rules of the mechanism to cause workers and employers to behave the way we want even though we cannot directly control their actions. Therefore, the rules of the mechanism are set so that stakeholders acting on their own will achieve the desired outcomes.

Stakeholders' Goals

The crowdsourcing recommendation model has three stakeholders, workers, employers, and the service provider. Each stakeholder has a multi-objective goal that is weighted differently based on each user. The main goal consists of satisfying three smaller goals:

The workers' goal, which is to get tasks that raise their monetary rewards and the overall rating, as demonstrated in Section 6.2.1.

The employers' goal, which is to get more qualified solutions at lower cost without affecting their rating, as demonstrated in Section 6.2.2.

The service provider's goal, which is to raise the task acceptance rate, which in turn will increase aggregated commissions, as demonstrated in Section 6.2.3.

Contribution

The main contributions of this study are as follows:

1. A model for quantitatively formulating the strategic interaction of the following stakeholders: employers, workers and the crowdsourcing service provider.

The proposed crowdsourcing model involves two decision makers (workers and employers), where each faces a set of behavioral choices. Each worker or employer strives to maximize utility (to achieve the most payoff). The payoff obtained by a given worker depends not only on the choices that the worker makes but also on the choices made by employers and other workers.

All workers and employers have opposing interests, causing their behavior to be proactive and strategic.

2. Algorithms to compute the recommendation for employers and workers and a recommendation model composed of:

   - Task recommendation for workers.
   - Worker recommendation for employers.

   In this study, we analyzed all the stakeholders' behavior in detail from their history and profiles. Consequently, a hybrid approach between content-based and collaborative filtering approaches was used. Then, based on each worker or employer's characteristics, we recommend the best choices for workers and employers to optimize certain qualities.

3. A numerical simulation to evaluate the effectiveness of the recommendation.

   We evaluated our model with synthesized datasets. To make the datasets realistic and unbiased, we generated them from two distributions, binomial and uniform, with different scales.

Conclusion

In this Chapter we presented the goal of the recommendation model which is achieving holistic satisfaction of all stakeholders, workers, employers, service provider. Moreover, we describes our main contribution which is the crowdsourcing recommendation model.

Chapter 5

Problem Formulation

The actual system that solves this problem should consider k employers post n tasks to m workers to maximize the commission.

This is not mainly about the money but rather a complex and definitive matrix of overall platform success. In other words, maximizing the commissions means maximizing the accepted tasks rate, which is a consequence of satisfying the employers by giving them a qualified solution and satisfying workers by giving them the associated rewards. This could be achievable by providing the correct recommendation for both employers and workers.

Stakeholders Role

In the following, we will identify the exact role for each stakeholder in the crowdsource platform.

Worker

- Lists skills on the profile
- $\langle$*Decision*$\rangle$ apply for n_{11} out of n_1 tasks that fits the profile.
- $\langle$*Decision*$\rangle$ completes $n_1{}^{‘}{}_1$ out of n_{11}
- $n_1{}^{‘}{}_1 < n_{11} < n_1 < n$

Employer

- Posts the task.
- $\langle$*Decision*$\rangle$ allots the task to the best m_{11} workers out of m_1 who applied for the task.
- $\langle$*Decision*$\rangle$ pay the $m_1{}^{‘}{}_1{}^{‘}$ workers out of $m_1{}^{‘}{}_1$ who submitted the solution to the task.

- $m_{1`_{1`}} < m_{1`_1} < m_{11} < m_1 < m$

5.0.1 Service Provider

- Orders the recommended tasks for the workers.
- Sorts n_1 tasks in the order that leads to maximize $\sum c_1$
  The ordering of n_1 is a list that looks like
  $j_{1_0}, j_{1_1}, j_{1_2}, j_{1_3}, \ldots, j_{1_{n_1}}$
  The worker accepts tasks with probability
  $P(j_{1_0}), P(j_{1_1}), P(j_{1_2}), \ldots, P(j_{1_{n_1}})$
  Where $P(j_{1_y}) \geq P(j_{1_{(y+z)}})$
  Where $z > 0, Z \in I$

workflow

In this section, we will provide an overview of the work flow as an interaction scenario for the stakeholders of the crowdsourcing platform.

- Employers $e_1, e_2, e_3, ...$ register as members.
- Workers $w_1, w_2, w_3, ...$ register as members.
- e_h posts tasks $a_j, a_{j+1}, .. : h, j = 0, 1, 2, ..$ at time $t_1, t_2, t_3, ...$
- For each task, employers may specify:
  1. The required skills,
  2. Monetary rewards, and
  3. Time deadline.
- Workers w_i, w_{i+1}, w_{i+2} are qualified for the task a_j.
- At time t_1, t_2, t_3, t_4, workers w_i, w_{i+2} apply for the task a_j as long as $t_{j+n} > t_l : l = 1, 2, 3, 4$. t_{j+n} is the threshold time when the e_h must respond to the workers who accepted the task with a decision of hired / not hired.
- Employer e_h removes the task a_j from the available tasks after two conditions are met: 1) the task was allotted to the number of required workers, and 2) the employer accepted the task from one or more workers.
- Workers w_i, w_{i+2} completed and submitted their work for the task a_j
- Employer e_h accepted the work for the task a_j from one or more workers, who submitted their work, and paid the associated rewards.

- In t_l second the service provider S made C_l dollars as a commission for the task a_j.[1]
- Consider another case for task a_{j_1} a_j, which finally gives C_2 in t_2 seconds[2].
- In a given duration of time T, maximize $\sum C_1 + .. + C_n$
- The probability that the task is completed and the employer accepted the task is P_1.
- The probability that S will get the commission C_1 is P_1.

The recommendation system should order the task's recommendation list to the workers such that the expected cumulative commission is maximized, which means
$\sum {P_1}^1 C_1 + {P_1}^2 C_2 + ... + {P_1}^n C_n$ is maximized.

Conclusion

In this Chapter we identified each stakeholder role in the crowdsource platform. Moreover, we described the stakeholder interaction through comprehensive scenario.

[1] C_1 is the commission that the service provider grant when the task a_j is accepted.
[2] C_2 is the commission that the service provider grant when the task a_{j_1} is accepted.

Chapter 6

Solution

This Chapter describes a mechanism design based on a multi-objective recommendation model to achieve holistic satisfaction of all stakeholders.

The Recommendation Model

This section describes the proposed model (Fig. 6.1). It is a multi-objective problem for both workers and employers. The worker's goal is to work on the tasks that maximize the reward and rating during a specified time. The employer's goal is to have more qualified solutions, pay less, and decrease the negative rating. In other words, if the employer hires a large number of workers for a task, the probability of getting more qualified solutions will increase. However, the employer in this case will have two choices. First, the employer could pay for all the workers who submitted qualified solutions, which will increase the cost. Second, the employer could pay for a subset of the workers who submitted qualified solutions, which will decrease the employer's rating from unsatisfied worker reviews.

The proposed model recommends the optimal choices for each worker and employer. Accordingly, the accepted tasks rate will be maximized and the service provider's goal will be achieved.

There are two cases in the proposed model: Case-0, where workers can work only on one task at a time, and case-1, where workers can work on multiple tasks at a time.

Worker Objective

In this section we will demonstrate the worker's goal in detail.

For each worker w_i,

Step one: Find the set of tasks that fits his interest

For each task a_j there are required skills $Sk[a_j]$=$\{sk_1, sk_2, ..\}$, and each worker has a set of skills $Sk[w_i] = \{sk_1, sk_2, ..\}$. If $Sk[a_j] \subset Sk[w_i]$, then

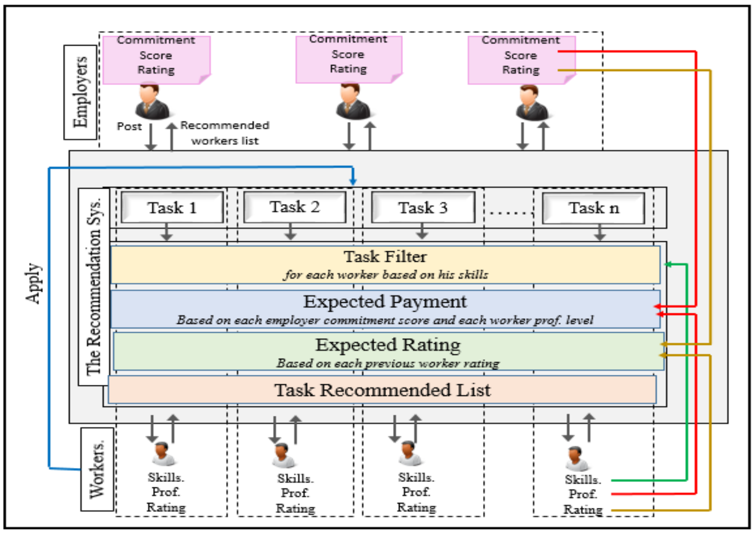


Figure 6.1: Recommendation Model

$a_j \in Tasks[w_i]$ where $Tasks[w_i]$ is the set that contains all the tasks that fit the worker's interest.
Step two: Calculate the expected monetary rewards for each task in the list $Tasks[W_i]$ using Algorithm 1. Considering the history, with weighted consideration of the future expectations, we apply the Discount Factor equation Eq.(16).
Step three: Calculate the expected rating for each task in the list $Tasks[w_i]$ using Algorithm 2.
Step four: Calculate each task's type weight using Algorithm 3.
Step five: Recommend tasks to the worker that will maximize the rewards and rating using Algorithm 4.

Expected Payment (ExP)

Each task a_j has a specified monetary reward, deadline, and required skills. The payment is not guaranteed unless the employer approves the work. Usually if the submitted work meets all the required specifications, the employer will approve the worker payment. However, there is no obligation for payment if the employer refuses to pay. Therefore, the employer's rating is an important factor to reflect the employer's trustworthiness.

From the worker's history, we can get an expectation of how likely the worker will be paid for each type of task in the set $Tasks[w_i]$. Moreover, based on the employer history, we can estimate how likely each employer will pay the worker.

Calculating the expected payment consists of two steps:
First: From the worker history, calculate the proficiency level of the worker

in each skill or type of task using the following equations:

Q_j: is the probability that worker w_i will complete tasks from type j

$$Q_j = \frac{\sum S[tasks_j]}{\sum H[tasks_j]} \tag{6.1}$$

Where for each worker, $S[tasks_j]$ is the submitted or the completed tasks from type j, $H[tasks_j]$ is the tasks that the worker was hired to process from type j

Q_{j1}: is the probability that worker w_i will be paid for tasks from type j

$$Q_{j1} = \frac{\sum Paid[tasks_j]}{\sum S[tasks_j]} \tag{6.2}$$

Where $Paid[tasks_j]$ is the accepted tasks from type j
The worker proficiency level in type j tasks is

$$Prof^j = Q_j * Q_{j1} \tag{6.3}$$

Second: Calculate the degree of employer trustworthiness, considering the worker's rating as a substantial factor to get more accurate results. For instance, a review from a five-star worker has more impact than a review from a two-star worker because more highly rated worker is more trustworthy.

From the employer history:
Q_h: is the probability that the employer e_h will pay the workers who submitted the solutions considering the worker's ratings $R[w_i]$ as a weight factor

$$Q_h = \frac{\sum Paid[w_i] * R[w_i]}{\sum S[w_i] * R[w_i]} \tag{6.4}$$

where for each task $Paid[w_i]$ is the workers who got paid. $S[w_i]$ is the workers who submitted the task.
Then from equations 6.3 and 6.4, we can calculate the expected payment for each task in the worker's task list by the following equation:

$$ExP[a_j] = Prof^j * Q_h * Reward[a_j] \tag{6.5}$$

Where $Reward[a_j]$ is the monetary reward for task a_j, $Prof^i$ is the proficiency level of the worker in type j tasks.

Maximize $ExP(W_i) = \sum_{i=1}^{y} ExP[a_j]$
Where y is the number of tasks in the $Tasks[w_i]$ set.

Algorithm 1 Expected Payment

INPUT: Tasks set $Tasks[w_i] = \{a_1, a_2, ..., a_n\}$
INPUT: Employers who posted this tasks $E = \{e_1, e_2, ..., e_h\}$
Each task a_j is a tuple of three values $\{rewards, deadline, skills\}$
Task types in $Tasks[w_i] : T_a = \{a_1, a_2, ..., a_m\}$
OUTPUT: The expected payment for each task a_j in $Tasks[w_i]$
First: Calculate the proficiency level for each type of task
for all a_t in T_a **do**
  Calculate $Q_j = \frac{\sum S[tasks_j]}{\sum H[tasks_j]}$
  Calculate $Q_{j_1} = \frac{\sum Paid[tasks_j]}{\sum S[tasks_j]}$
  Calculate $Prof^j = Q_j * Q_{j_1}$
end for
Second: Calculate the employer commitment
for all e_i in E **do**
  $Q_h = \frac{\sum Paid[w_i]*R[w_i]}{\sum S[w_i]*R[w_i]}$
end for
Finally: Calculate the ExP for each task a_j in $Tasks[w_i]$
$ExP[a_j] = Prof^j * Q_h * Reward[a_j]$

Expected Rating (ExR)

The rating system in the crowdsourcing system allows employers and workers to rate each other. Rating is a substantial factor, so it is important to optimize the rating score. From the employer's perspective, workers' ratings could help decide which worker should be hired. From the worker's perspective, rating could help decide which tasks to apply for. As we have described in the employer rating equation 6.4, the evaluator rating is considered to aggregate the overall rating score. To justify the evaluator rating factor needs, consider the following example. Because the rating system is mutual as we explained, a dishonest employer could give workers a bad rating to decrease their overall rating. This lowered rating would result in workers' evaluations not having much effect on the employer's rating in equation 6.4. However, if we consider the employer's rating in evaluating the workers' ratings, the rating score could be more trustworthy.

$$ExR[j] = \frac{\sum_{x=1}^{n} R[a_x] * R[e_h]}{\sum_{x=1}^{n} R[e_h]} \tag{6.6}$$

Where $ExR[j]$ is the expected rating for type j tasks, n is the total number of type j tasks that the worker has submitted before. $R[a_x]$ is the rating score for task x. $R[e_h]$ is the employer e_h rating.

Algorithm 2 Expected Rating

```
INPUT: Task set Tasks[w_i] = {a_1, a_2, a_3, ..., a_n}
INPUT: Employers who posted these tasks E = {e_1, e_2, ..., e_h}
Each employer has a rating score value R[e_h]
OUTPUT: The expected rating for each task a_j in Tasks[w_i]
for all a_j in Tasks[w_i] do
    Calculate ExR[j] = (sum_{x=1}^{n} R[a_x]*R[e_h]) / (sum_{x=1}^{n} R[e_h])
end for
```

Skill Based Workload

From the workers' history, we can calculate how many tasks each worker can handle successfully at the same time. In other words, what is the worker's appropriate workload based on the task's type. For example, worker w_i could work successfully on average of three tasks simultaneously when he is working on programing task, average of two task when he is working on design task, and so on for each type of task.

For each worker, calculate the appropriate workload for each task's type j.

For each worker, if there were k_1 tasks were being done together during a given instance s_i, among which the present task's type was one, find the average of the total number of tasks as follow:

$$L[j] = \frac{\sum_{i=1}^{S} Paid[a_j]}{S} \tag{6.7}$$

Where $L[j]$ is the worker w_i workload for task's type j, S is the total number of the instances, and $Paid[a_j]$ is the number of the accepted tasks during an instance s_i, considering only the instances which the present task's type was one.

By applying Eq.(7) each worker w_i will have a different workload for each task's type. Each workload will be converted to a weight score by the following Equation:

$$Tw[j] = \frac{1}{L[j]} \tag{6.8}$$

Where $Tw[j]$ is the worker's w_i weight score for task's type j.

For example, if the worker w_i workload for programming task is 3, that means he could work efficiently on two more task beside one programming task and the weight score for programming task will be equal to 0.33.

Algorithm 3 Task's Type Weight (Tw)

INPUT: worker w_i task's type set $T_a = \{a_1, a_2, ..., a_m\}$
INPUT: Previous instances set $S = \{s_i, ..., s_n\}$
Each instance s_i is a tuple of instance's id and the list of the worker's accepted task during this instance.
OUTPUT: Tw set which contain the weight score for each task's type.
Int ins-counter = 0, task-counter = 0;
for all a_i in T_a **do**
    ins-counter = 0, task-count = 0;
    for all s_i in S **do**
        if s_i contains task from type a_i **then**
            ins-counter ++ ;
            task-counter + = the total number of tasks in s_i;
        end if
    end for
    $L[a_i]$ = task-counter / ins-counter ;
    $Tw[a_i] = 1/L[a_i]$
end for

Worker Recommendation Task

The worker utility function is $Maximize[reward, rating]$
It is a multi-objective optimization problem (MOP) with two objectives: reward and rating. In the literature, researchers have studied MOP from a different point of view, and so different solution philosophies and goals exist. There are three main classes for preference MOP, where preference information is needed to solve the problem. These classes are a priori, a posteriori, and interactive where a preference information is involved from the decision maker (DM) in different ways. In the a priori method, the DM will first determine the preference information, and then the solution will be found. In the a posteriori method, the solutions will be found first, and then the DM will choose from them. In the interactive method, the DM's preference information will be specified during the computation.

To optimize the worker objectives, the a priori method is used in this paper. Workers will specify their preference rating score first, which will be used as a constraint value and solve for maximizing the reward's value.

New workers could be more interested in building a robust history and set the rating constraint value to four or five stars in order to increase their future chances to compete with senior workers, who have a high rating score, to get hired. However, each worker could set the rating constraint based on interest.

$$Maximize f(x) = \sum_{j=1}^{n} Reward[a_j] \quad (6.9)$$

Subject to $R[a_j] \geq R$
Where R is a rating constraint value set by the worker.

There are two cases in the proposed model:
Case-0: Only one task at a time. It can be solved by sorting the tasks based on the expected payment considering the expected rating as a constraint value.
Case-1: multiple tasks at the same time. If the worker wants to work on a set of tasks to maximize his/her objectives during a specified time, dynamic programming is used to solve the problem.

It becomes a knapsack problem where we try to maximize the value within the time limit considering the weight score for each task from Algorithm 4 where the total weight score should be equal or less than one. The following is an illustrated example:

If the worker has the matched tasks set as in Table 1, by applying the Tw Algorithm 4, the tasks demonstration during the time period is shown in Figure 6.2.

Table 6.1: Task List

Task	*Type*	*Deadline*	Tw_i	*ExP*
1	*Programming*	*30*	*0.33*	*300*
2	*Programming*	*90*	*0.33*	*600*
3	*Web design*	*120*	*0.5*	*500*
4	*Web design*	*60*	*0.5*	*400*
5	*Programming*	*30*	*0.33*	*250*
6	*Web design*	*90*	*0.5*	*300*

The Worker Recommendation Task algorithm uses dynamic programing for the knapsack problem [81], where $KnapSack(ExP, Tw, n, T)$ = recommended task set, ExP is the item value, Tw is the weight value, n is the number of items, and T is the total weight.

Employer Objective

In this section, we will demonstrate in detail the employer's goal. The employer objective is to obtain more qualified solutions for the payment expended and rating awarded. Hiring more workers could increase the chance of receiving a better solution. Consequently, the two main objectives, which are cost and rating, will have a negative effect as we described in section 1.

Algorithm 4 Worker Recommendation Task

```
INPUT: Task set Tasks[w_i] = {a_1, a_2, ..., a_k}
INPUT: ExP = {ExP[a_1], ExP[a_2], ...}
INPUT: ExR = {ExR[a_1], ExR[a_2], ...}
INPUT: Tw = {Tw[a_1], Tw[a_2], ...}
INPUT: minimum rating constraint R
INPUT: Time to optimize TL.
OUTPUT: The recommended tasks
W_i = ∅
for all a_j in Tasks[w_i] do
    if a_j deadline > TL then
        exclude a_j
    end if
    if ExR[a_j] > R then
        add a_j to W_i
    end if
end for
if Worker works on one task at a time then
    Sort W_i based on ExP[a_j] in decreasing order
else
    TaskSet = KnapSack (ExP, Tw, n, T)
end if
Print TaskSet
```

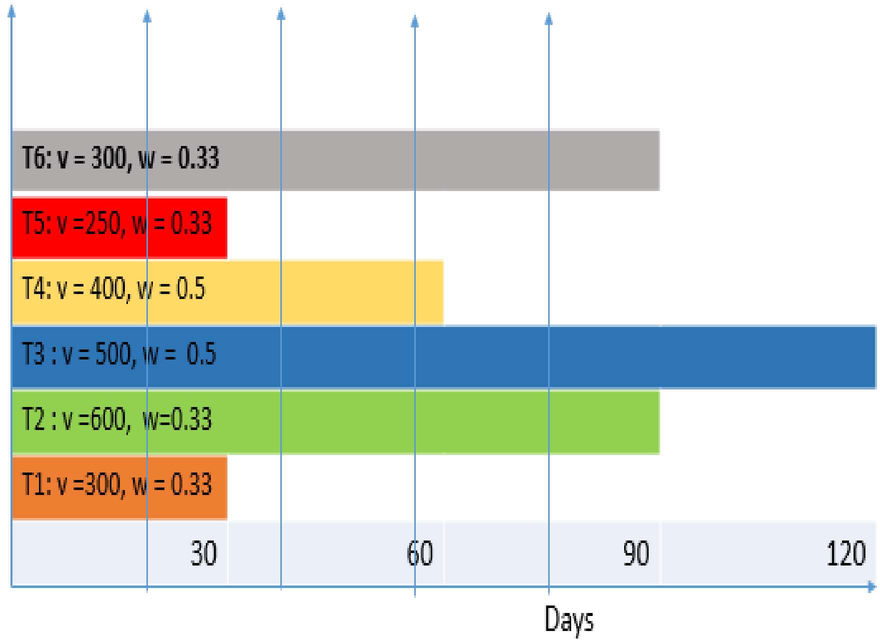


Figure 6.2: Task list

The employer objectives could be optimized by choosing fewer qualified workers. By doing so, we will decrease the number of unsatisfied (due to not being compensated as per their expectation) workers, which could cause negative ratings and decrease the rewards.

Another important factor is the worker's current workload. Some workers may apply for a large number of tasks and then choose a subset of these tasks to process. Some other workers may choose to work on a large number of tasks, which could decrease the quality of the task solution.

To address the employer objectives, we will tackle the aforementioned factors as follows:

First, for each task a_j posted by employer e_h, k workers will apply. Each worker w_i will have a proficiency level for this kind of task from equation 6.3. Moreover, each worker has a rating score based on employers' evaluations. Then, for each worker, we calculate the potential success (PS) by

$PS = Prof^i * R[w_i]$

Let x be the number of workers for task a_j.The expected expenditure for task a_j is

$$F(x) = Minimize_x[Maximize \sum_{i=1}^{k} Prof^i * R[w_i](Reward[a_j])] \quad (6.10)$$

The employer objective function will be
$Maximize\ [workdone - payment - negativerating]$

Second, from the workers' current processing tasks, we can get the current worker's workload based on each task's weight score in section 5.1.3.

If the worker w_i current workload (CW) is 0.75, that indicates the worker still could work efficiently on more tasks. However, if the CW of another worker w_j is 0.0, it indicates worker w_j could process the task better because the worker has more time compared with w_i considering an equivalent or comparable PS score for both workers.

Finally, to optimize the employer objectives, the service provider needs to recommend workers with a higher PS score and a lower CW. To solve this MOP, the interactive method is used as follows:

1. Employer e_h sets the number of the required workers.

2. The service provider will find all the non-dominated solutions as described in the Worker Recommendation List Algorithm.

3. Based on this list, the employer e_h will reset the number of required workers.

The service provider will recommend the employer to choose at least two qualified workers and some new workers who are willing to build a history. Hiring new workers could increase the chance of getting better solutions in terms of increasing the number of workers, but it does not have much negative effect on the employer rating because they do not have a sufficient rating score.

Employers will set their own parameters to optimize their goal based on the applicants' PS and CW. If there are two 0.9 applicant workers, the employer could set the number of required workers to two plus some new workers to help them in building a history. However, if the applied worker has a lower PS and higher CW, the employer could increase the number of required workers. The mutual rating system could contribute to minimizing employing unnecessarily large numbers of workers and wasting their time processing a task with a low chance of acceptance.

Algorithm 5 Worker Recommendation List

INPUT: Workers set $W = \{w_1, w_2, w_3, ..., a_k\}$
Each worker w_i has a proficiency score $Prof_i$ and rating score $R[w_i]$
Each worker w_i has a CW
ND list $= \varnothing$
OUTPUT: List of all the non dominated workers
for all w_i in W **do**
  Calculate the PS: $PS[w_i]$ by $Prof_i * R[w_i]$
end for
for all w_i in W **do**
  if $PS[w_i] > PS[w_{i-1}]$ **then**
    if $CW[w_i] < CW[w_{i-1}]$ **then**
      Add W_i to ND
    end if
  end if
end for
for all w_i in W **do**
  if $CW[w_i] < CW[w_{i-1}]$ **then**
    if $PS[w_i] > PS[w_{i-1}]$ **then**
      Add W_i to ND
    end if
  end if
end for

Service Provider Objective

The service provider objective is to maximize the aggregated commission by listing the recommended tasks to the workers and the recommended workers to the employers.

S_h : Probability of worker w_i will apply for task a_j

S_{h_1}: Probability of worker w_i will get the task a_j

$S^{`}{}_{h_1}$: Probability of worker w_i will complete the task a_j

$S^{``}_{h_1}$: Probability of worker w_i will get paid for task a_j

$MaximizeTask(a_j) = \sum_{i=1}^{y} S_h S_{h_1} S^{`}{}_{h_1} S^{``}_{h_1}(C)$

Where $C =$ the commission that aggregated for task a_j.

The service provider utility function is

Maximize [commission - negative employ rating - negative worker rating]

Discount Factor

We get the value of 'n'– the depth of history we are going to consider – from the demography in the system. We consider the history of similar tasks until the effect of that state becomes less than ϵ in terms of probability. In other words, the effect of that state is no more than random on the present state. If most workers completed three tasks one after another, we would get $n = 3$, which means we are going to consider three history records.

Once we have the n, we can calculate the discount factor β. The discount factor is needed because we are considering that what happened in the recent past is more influential on the worker's future attitudes.

$$\beta + \frac{\beta}{2} + \frac{\beta}{3} + ... + \frac{\beta}{n} \tag{6.11}$$

From equation 6.11, we can get the value of β

$$\beta = \frac{1}{1 + \frac{1}{2} + \frac{1}{3} + ..\frac{1}{n}} \tag{6.12}$$

We are looking for the probability of worker w_i getting payed for job j_i now.

$$P(\epsilon(j_{i-1}) = \beta_0 * P(\epsilon(j_i) + \beta_1 * P(\epsilon(j_{i_1}) + ..\beta_n * P(\epsilon(j_{i-n}) \tag{6.13}$$

$Where \sum_{k=1}^{n} \beta_k = 1$

$$\sum_{k=1}^{n} \beta_k P(\epsilon(j_{i-K+1}) \tag{6.14}$$

$$Where \sum_{k=1}^{n} \beta_k = 1$$

Conclusion

In this chapter, we presented a mechanism design capable of achieving holistic satisfaction using a multi-objective recommendation model that recommends the optimal choices for each worker and employer. The model is designed as an interactive system, where each worker and employer can set the parameters that meet their goals. We used a hybrid approach that combines content recommendation and collaborative filtering.

The model has addressed the goals of all three stakeholders (the worker, the employer, and the service provider) through a comprehensive history analysis for each worker and employer. This information is then used in a dynamic algorithm that seeks to maximize the benefit to all stakeholders with some preference constraints specified by each stakeholder.

Chapter 7

Cold Start Problem

In a situation where there is a new worker, new employer, or new type of task, traditional recommendation systems cannot provide a proper recommendation due to a lack of required information. This is a well-known problem in recommendation systems called the cold start problem.

There are two main categories of recommendation systems [31]: The content-based approach and the collaborative filtering approach. Collaborative filtering outperforms the content-based approach in providing more diverse recommendations, while the content approach provides overspecialized recommendations. However, the major drawback of collaborative filtering is the cold start problem.

The proposed crowdsourcing recommendation model is a hybrid system that combines these two approaches to overcome their limitations (as mentioned in Chapter 1) and to provide more accurate recommendations. However, the model still has some cases that face the cold start problem.

There are several studies in the literature dedicated to solving this problem but most of them suffer from scalability issues. In addition, the cold start problem in the crowdsourcing paradigm has not been well studied [96].

The most common solution is based on matrix factorization. In this solution, the recommendation model extracts features from each stakeholder. Each item is represented by a set of features denoted as a vector, and each user is represented by a set of features with a rating score for each feature. The rating score could be produced from worker history, demographic information, a social network, or explicit questions. Next , all this information is combined linearly in a unified rating vector [64] or by assembling independent rating vectors [66]. then, the recommendation system will recommend the items with features favorable to each user [64, 96, 97, 56, 67, 61].

This kind of solution cannot address the cold start problem efficiently in the case of a new worker, new employer, or new task because of the lack of features needed to produce a rating score.

This study proposes a different approach to solve this problem by adding a simple technique without any negative effect on scalability.

New Worker

In our proposed recommendation model, workers get task recommendation based on explicit factors in the form of listed skills and implicit factors from their proficiency and rating scores, as well as the employer's trustworthiness score. The worker proficiency and rating scores are deduced from worker history.

New workers will get task recommendations based on the skills listed in their profiles. However, the probability for new workers to get hired is low compared to workers who have a longer history.

Employers will get a worker recommendation list based on the workers' potential for success (see Section 5.2.2).

To help new workers build their work history, we suggest two kinds of hiring requests:

1. Paid hiring request, which is the regular hiring request.

2. Evaluation hiring request. New workers can apply for evaluation hiring request to build their working history to increase the probability of getting paid hiring approval in the future.

In an evaluation hiring request, employers rate workers and choose whether to accept or not accept. That means if the workers was being paid hired, would the employer paid the associated reward or not. This information is needed to build the workers' proficiency score and rating score to decide his potential success score (see Section 5.2.2). Consequently, the workers' probability of getting paid hiring requests will increase.

Overspecialized

Workers get recommendations based on their past performance for each listed skill in their profile. However, if workers start in processing type j_1 task, their proficiency score will change in this specific type based on their performance regardless of the other types. In this case, workers will get stuck in one type of task recommendation, which is called the overspecialized recommendation problem.

To overcome the overspecialized recommendation problem and broaden the task recommendations, workers can apply for evaluation hiring in other types of tasks to build their work history in their other skills.

$$[w_i] = Max[\sum_{j=0}^{J} Prof^j * R[j]] \tag{7.1}$$

Where J is worker's w_i number of skills.

New Employer

The essential factor for employers is their trustworthiness, which is measured based on employer history (see Section 5.2.1). As mentioned in Chapter 1 and section 5.2.1, in crowdsourcing, there is no payment obligation. Employers can hire any number of workers to process their tasks and then pay for a subset of workers who submitted a good solution.

Employers e_h pay for $W_y \subseteq W_x \subseteq W_j \subseteq W_i$.

Where:
W_y is the set of paid workers.
W_x is the set of workers who submitted a good solution.
W_j is the set of workers who submitted a solution.
W_i is the set of hired workers.

The goal is $Min[W_i - W_y]$ which means minimizing the number of unpaid workers by recommending the right workers for each task. Workers prefer to work with trusted employers to increase the probability of getting paid.

To overcome the new employer problem, we need a new kind of task that only hires experienced workers. Experienced workers are workers with a high proficiency score in a particular skill. The new kind of task is called an evaluation task.

To promote tasks posted by a new employer, the new employer needs to post an evaluation task request and hire an experienced worker to evaluate the workers' solutions as a trust factor. The evaluator is granted access to reward control in order to issue the payment request. Then, the new employer posts the task along with the evaluator. After solving the trustworthiness problem, workers will be more willing to apply for this task.

If the new employer accepts the task from one or more workers, the evaluator will be paid for his/her contribution to increase the employers trustworthiness. If the new employer refuses to pay any worker who has submitted a solution, the task solution will be transferred to the evaluator. If the evaluator decides to reject the task, no further action is required. If the evaluator decides to accept, he/she can issue the payment order to the worker who submitted the best solution and rate the employer.

New Task

Each type of task is associated with one or more skills. A new kind of task here means a task that requires a new skill. If the skill is listed in some workers' profiles, the task with this particular skill will be recommended to them.

If a task type is new, it means no task of this type has been posted before. Consequently, many workers may not list this skill in their profile even if they have this skill and thus miss the opportunity to work on this type of task.

To solve this problem, we used a user-based collaborative filtering approach to endorse the new skill to similar workers.

The collaborative filtering approach is built on three main assumptions [95]:

1. People have similar tastes and interests.
2. Their tastes and interests are stable.
3. We can infer their interests from their previous behavior.

Collaborative filtering is designed based on a comparison between users' behavior in order to find similar users, called neighbors, and according to the user's neighbours, we can predict user preferences.

The first step in the collaborative filtering algorithm is to have the users' history profile and then represent it as a rating matrix where each row represents a worker and each column represents a skill. The value in the intersection contains the proficiency score for worker w_i in skill sk_j. The missing value of a rating score at an intersection indicates that worker w_i does not have this skill or did not list it in his/her profile.

An example would be if we had 10 workers with 4 skills sk_1, sk_2, sk_3, sk_4 or task types and a new skill sk_5 that we wanted to promote (see Table 1).

First, new task is recommended to the workers who already list the new required skill in their profiles. Then, if they agree to work on this task, the value of the intersection in the workers' rating matrix will be set to one, and if they reject it, it will be set to zero.

Second, for the users who did not list the skill in their profile, their nearest neighbors will be found by calculating the similarity between users. There are many ways to measure this similarity. We chose to use the Cosine similarity measure method [95], which is calculated by the following equation:

$$sim(x,y] = \frac{\sum_{s \in S_{xy}} r_{x,s}, r_{y,s}}{\sqrt{\sum_{s \in S_{xy}} r_{x,s}^2}\sqrt{\sum_{s \in S_{xy}} r_{y,s}^2}} \tag{7.2}$$

Table 7.1: Workers Rating Matrix

Workers	sk_1	sk_2	sk_3	sk_4	sk_5
w_1	*0.9*	*0.8*	-	*0.5*	-
w_2	*0.7*	*0.6*	*0.4*	-	-
w_3	*0.8*	*0.7*	-	-	*1*
w_4	*0.9*	*0.7*	*0.6*	-	*1*
w_5	*0.7*	*0.8*	*0.5*	-	*0*
w_6	*0.4*	*0.8*	*0.7*	*0.5*	*0*
w_7	*0.8*	*0.9*	*0.8*	-	-
w_8	*0.7*	*0.6*	*0.8*	*0.5*	-
w_9	*0.8*	*0.8*	*0.5*	-	*1*
w_10	*0.8*	*0.6*	-	*0.9*	-

Where r_x is proficiency score of user x on skill s, and r_y is proficiency score of user y on skill s. S_{xy} indicates the skill that is common between users x and y.

The third step is to calculate the potential that workers will accept the new skill by creating a weighted average of the neighbors' reaction to the new type of task via the following equation:

$$r_{x,s} = \bar{r}_x + \frac{\sum_{y \in S_{xy}} (r_{y,s} - \bar{r}_x) sim(x,y)}{\sum_{y \in S_{xy}} sim(x,y)} \tag{7.3}$$

Where $\bar{r}_x$ is the average reaction of user x. If the average reaction ≥ 0.5, the majority of the user's neighbors agree to work on the new task and the skill will endorse to user x. In other words, the skill will be suggested to user x so that they can accept or reject it as a skill they have.

Finally, if the new skill is not listed in any of the workers' profiles, the skill will endorse to all workers to either confirm or reject this skill. In other words, all workers will be given the option to confirm or reject this skill.

Conclusion

This chapter addressed the overspecialization and cold start problems, which prevent the recommendation system from providing sufficient recommendations for new workers, employers, and task types by presenting a new type of hiring request and using the user-based collaborative filtering approach. Moreover, we adapted the LinkedIn [85] technique of recommending new skills to neighbors.

Chapter 8

Testing and Analysis

In this chapter, we describe an experiment that simulates the crowdsourcing paradigm. Our experiment is designed to address three questions:

1. How does the proposed method compare with baseline and state-of-the-art approaches?
2. What is the computational complexity of the proposed recommendation model?
3. How scalable is the proposed model?

To demonstrate the superiority of our proposed model, we chose two models as a baseline for the comparison: the traditional model and the most recently published model.

Baseline Model

The traditional model use greedy algorithm to recommend the highest reward tasks that match workers' skills. The most recent recommendation system in crowdsourcing relies on matrix factorization based on worker performance history and worker task searching history [94].

Data Set

The data needed to evaluate our proposed model requires the complete worker history and employer history. To the best of our knowledge, such data is only accessible by the crowdsourcing administrators and is not publicly available.

We evaluated our model with synthesized datasets. To make the datasets realistic and unbiased, we generated them from two distributions, binomial and uniform, with different scales. Table 1 shows the characteristics of the synthesized datasets. Binomial distributions were chosen because each submitted task has only two possibilities, accept or reject. The rating value was generated using discrete uniform distribution, yielding integers only. The datasets generated are implemented using numpy.random sampling module in Python [**numpy**]. With this module, the generated data can be customized randomly from any distribution with specified parameters. Experiments were conducted on a standard desktop PC (Quadcore Intel i7 CPU@3.5 GHz).

Table 8.1: Characteristics of Datasets

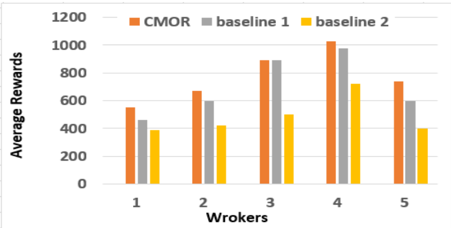
Dataset	*Dist.*	*Task*	*Category*	*Worker*	*Employer*
D1	*binomial*	*1000*	*5*	*50*	*50*
D2	*binomial*	*5000*	*10*	*100*	*100*
D3	*uniform*	*1000*	*5*	*50*	*50*
D4	*uniform*	*5000*	*10*	*100*	*100*

Experimental Procedure

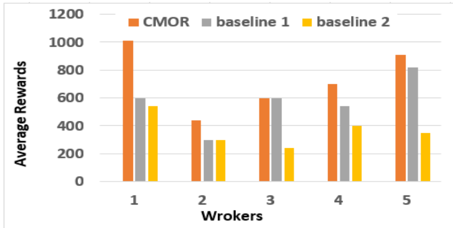
First, we evaluated the workers' objectives. For the comparison goals, we compared the reward average of five randomly selected workers in each model. Each worker had different proficiency and rating scores associated with each skill.

In our model, we calculate the expected rewards and rating. However, in the crowdsourcing paradigm, payment is not guaranteed as described earlier. To simulate the crowdsoursing paradigm, we designed a stochastic program that runs 10, 20, and 50 times, each time with a possibility of acceptance or rejection based on the employer's commitment score and the worker's proficiency score. In each run, a random number will be generated. If the number is between zero and the potential acceptance score, the task will be considered accepted; otherwise, it will be rejected. Then, the number of accepted times will be multiplied by the actual rewards. Finally, we calculate the reward's average. The potential acceptance is calculated by multiplying the employer's commitment score by the worker's proficiency score as described earlier in the algorithms.

We ran the simulation 10, 20, and 50 times on each dataset, and compared the average rewards for the selected workers in the proposed model with the average rewards of the same workers in the two baseline models. In Baseline 1, a greedy approach was used to choose the set of tasks that would maximize the worker's objectives. In Baseline 2, the worker's performance was considered and tasks would be recommended based on the worker's previous performance. To evaluate the potential acceptance in the baseline models, we considered additional information consisting of the em-



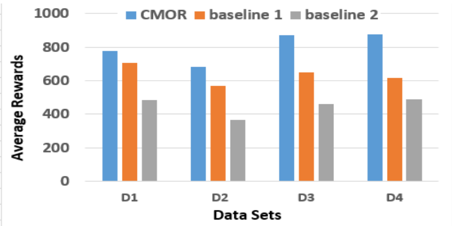
(a) Workers' Reward Average in D1



(b) Workers' Reward Average in D2

(c) Workers' Reward Average in D3

(d) Workers' Reward Average in D4



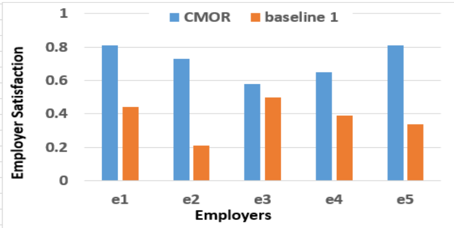
(e) Overall Rewards Average in each Dataset

Figure 8.1: Evaluating the Workers' objectives

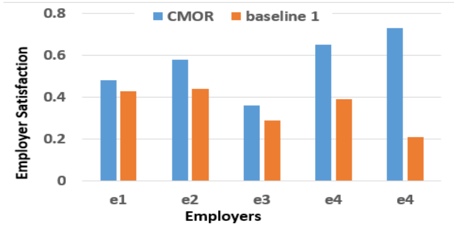
ployer's commitment score and the worker's proficiency score. The following fugues shows the average rewards for the selected workers in dataset D1 Fig. 3(a), dataset D2 Fig. 3(b), dataset D3 Fig .3(c), dataset D4 Fig .3(D). The average rewards for all selected workers in each dataset is shown in Fig. 3(E).

Second, we evaluated the employers' objectives. For the comparison goals, we randomly selected five employers and for each employer we randomly selected one task. To evaluate the employers' objectives, we compared their potential satisfaction with hiring each worker for the selected task.

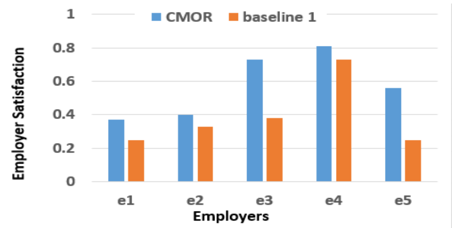
To simulate the employer's role, we ran the simulation 10, 20, and 50 times on each dataset, the simulation calculating the probability of the potential satisfaction for each selected employer. Each time had a possibility of worker success or failure based on the worker's PS for task a_j. In each run, a random number was generated. If the number was in the interval of the worker's success range based on the $PS[a_j]$ score, the task was considered accepted and the worker succeeded. Then, the program calculated the



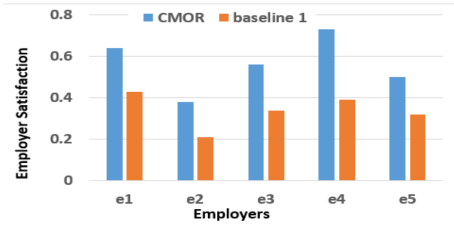
(a) The Probability of Employers' PS in D1



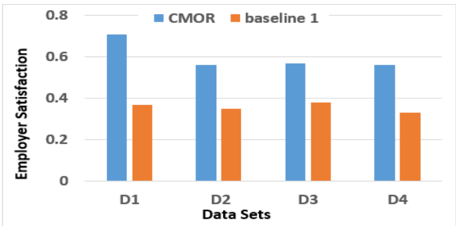
(b) The Probability of Employers' PS in D2



(c) The Probability of Employers' PS in D3



(d) The Probability of Employers' PS in D4



(e) The Probability of Overall Employers' PS in each Dataset

Figure 8.2: Evaluating the Employers' objectives

average of the worker's successful outcomes. After running the program 10, 20, and 50 times, the success average for each worker was calculated and compared with the baseline for the employer recommendation system, which recommends workers with the highest rating scores. The following fugues shows the probability of the employers satisfaction for the recommended workers in dataset D1 Fig. 4(a), dataset D2 Fig. 4(b), dataset D3 Fig .4(c), dataset D4 Fig .4(D). The average rewards for all selected workers in each dataset is shown in Fig. 4(E).

Illustrated Example

Data set consists of ten tasks from five categories posted by three employers for four workers. Employers have different rating scores {5, 3, 4} and different commitment scores {0.90, 0.70, 0.80}, respectively. Workers have different proficiency scores and different rating scores for each skill. The evaluation consists of two parts.

First, we evaluate the worker objectives.
For the comparison goals, we will compare the reward average of one worker in each model. This worker has three skills in his profile {Programming, Web Design, Graphics} each one associated with a proficiency level score {0.8, 0.7, 0.85} and a rating score {4, 3, 5}, respectively.

Table 8.2: Posted Tasks

Tasks	*Category*	*Employers*	*Reward*	*Deadline*
1	*Programming*	e_1	*550*	*30*
2	*Web Design*	e_2	*400*	*90*
3	*Algorithm*	e_3	*300*	*60*
4	*Database*	e_2	*600*	*70*
5	*Algorithm*	e_1	*400*	*30*
6	*Web Design*	e_3	*600*	*120*
7	*Web Design*	e_1	*500*	*60*
8	*Programming*	e_2	*350*	*30*
9	*Programming*	e_2	*560*	*90*
10	*Programming*	e_1	*559*	*110*

First, the tasks will be filtered to match the workers' skills. Then by applying the expected payment algorithm 1, expected rating algorithm 2, and task type weight algorithm 3. The task's list is represented in Table 3.

Table 8.3: Processed Data

Tasks	*e.Comm.*	*W. Prof.*	*ExP*	*ExR*	*Tw*
1	*0.9*	*0.8*	*396*	*4*	*0.33*
2	*0.7*	*0.7*	*196*	*3*	*0.5*
6	*0.8*	*0.7*	*336*	*3*	*0.5*
7	*0.9*	*0.7*	*315*	*3*	*0.5*
8	*0.7*	*0.8*	*196*	*4*	*0.33*
9	*0.7*	*0.8*	*313*	*4*	*0.33*
10	*0.9*	*0.8*	*402*	*4*	*0.33*

Case-0: Only one task at a time.

If the worker chooses the recommended task (task 10), which has the highest expected payment, the potential acceptance will be 0.72 . To simulate this fact, in each run, a random number will be generated. If the number is between zero and the potential acceptance score, the task will be considered accepted; otherwise, it will be rejected. Then, the number of accepted times will be multiplied by the actual rewards. Finally, we calculate the reward's average.

We ran the simulation 20, 50, and 100 times, and the average reward was between \$397 and \$446. We compared this average with the average of the traditional recommendation system, which chooses the highest rewards. As Table 1 shows, the recommended task will be task 6. Considering the additional information, which are the employer's commitment score and the worker's proficiency score, the potential acceptance is 0.8 multiplied by 0.7 and is equal to 0.56. The acceptance range will be in the interval between zero and 0.56. Then, by applying the same steps for running the proposed model, the average reward was between \$240 and \$326.

The same procedure was used to test the recent work, which considered the worker performance history. The recommended task will be task 9 and the average reward was between \$279 and \$360.

Case-1: Multiple tasks at the same time.
Assuming that the worker chooses to maximize his/her objectives with a three-stars rating constraint, we will compare the three approaches as follows:

In the traditional recommendation system, a greedy approach is used to choose the set of tasks that will maximize the worker's objectives. Only task 6 will be recommended, which has the highest rewards. However, the potential acceptance rate for this task is 0.56 based on the worker's and employer's history information. Based on this information, the average reward is \$336.

In the most recent approach where the worker's performance is considered, task 9 and task 1 will be recommended. The potential acceptance for the two tasks are 0.56 and 0.72, respectively. Considering the four possibilities, which are (accept task 9 and task 1), (accept task 9 only), (accept task 1 only), and (reject the two tasks), the average reward is \$354.

In our proposed approach, task 10, task 1, and task 8 will be recommended with potential acceptance rates of 0.72, 0.72, and 0.56, respectively. Considering the eight possibilities, which are (accept task 10, task 1 and task 8), (accept task 10 and task 1), (accept task 10 and task 8), (accept task 1 and task 8), (accept task 10), (accept task 1), (accept task 8), and (reject the three tasks), the average reward is \$497.

Second, to evaluate the employers' objectives, we compared the employer's potential satisfaction for hiring each worker. For a specific task from type programming, for example, four workers have applied with different proficiency scores {0.9, 0.9, 0.7, 0.8} and different ratings {5, 5, 4, 3.5} respectively for programming skills. Moreover, each worker has an LP score {1, 0.5, 0.5, 0} as shown in Table 4.

Table 8.4: Applicant List

Worker	*Prof.*	*Rating*	*Rating in percentage*	*LP*
1	*0.9*	*5*	*1*	*1.0*
2	*0.9*	*5*	*1*	*0.5*
3	*0.7*	*4*	*0.8*	*0.5*
4	*0.8*	*3.5*	*0.7*	*0.0*

The proposed recommendation model will calculate the PS_{a_j}, which is the potential success for the task a_j for each worker who applied for the task a_j as shown in Table 5.

Table 8.5: Worker Success Probability

Worker	*PS*	*LP*	$Free_{time}(1 - LP)$	PS_{a_j}
1	*0.9*	*1.0*	*0.0*	*0.0*
2	*0.9*	*0.5*	*0.5*	*0.45*
3	*0.56*	*0.5*	*0.5*	*0.28*
4	*0.56*	*0.0*	*1.0*	*0.56*

The proposed recommendation system will recommend workers w_2 and w_4 in the first place. To simulate the employer's role, the designed program will calculate the probability of the potential employer satisfaction. The program runs several times, each time with a possibility of worker success or failure based on the worker's PS for task a_j. In each run, a random number will be generated. If the number is in the interval of the worker's success range based on the PS_{a_j} score, the task will be considered acceptable and the worker succeeds. Then, the program calculates the average of the worker's successful outcomes. After running the program 20, 50, and 100 times, the success average for each worker is presented in Table 6.

Table 8.6: Applicant Success Average

Worker	*Success Average*
4	*0.65*
2	*0.58*
3	*0.30*
1	*0.0*

However, that does not mean the failure of worker 1 is guaranteed even though his/her success average is 0.0. The formula gave an estimate based on the worker history.

The baseline for employer recommendation systems is to recommend workers with the highest rating scores. Based on this, the recommended worker list will be {1 ,2} in the first place then 3 then 4. In this case, there will be no preference to hire worker 2 over 1, which means the employer will have a 45% less chance to get a completed task if he/she chose to hire worker 2 over worker 1. The probability of the employer choosing worker 2 is 50%.

Computation Complexity and Scalability

The computational complexity for the proposed algorithms is $O(nW)$. All the history building components, such as employer commitment and worker proficiency, can be calculated in a constant time after each submitting process for the workers and employers involved. The computation complexity of the expected payment and expected rating algorithms is $O(n)$ because we have one loop that runs n times, which contains some equations that run constantly. The time complexity for the dynamic solution of the knapsack problem is $O(nW)$, where n is the number of tasks that match the worker's skills, and W is the total time during which the worker chose to maximize his/her objectives.

Conclusion

The experimental simulate the crowdsurce paradigm to evaluate the stakeholders' objectives on a synthesized dataset. The experimental simulation showed the superiority of the proposed model compared with two other baseline models.

Chapter 9

Future Work

We have proposed a multi-objective recommendation model for the crowdsourcing paradigm. The model has addressed the satisfaction of all major stakeholders, including workers, employers, and the service provider. The model met stakeholders' objectives by (1) recommending tasks to workers that will maximize their monetary rewards and rating; (2) recommending the best workers to employers, who will minimize overall cost and increase the employer's rating; and (3) raising the task acceptance rate, which will increase the aggregated commissions. The model is designed as an interactive system where every worker and employer can set the parameters that meet their goals. All previous crowdsourcing recommendation systems are designed to address only one stakeholder , either the worker or the employer. Moreover, no previous crowdsourcing recommendation systems have considered the other party's behavior to provide more qualified recommendations as we have done.

The experimental simulation showed the superiority of the proposed model compared with two other baseline models.

Our model uses a one-shot game where the decision is made simultaneously [7]. However, in the future, we plan to use a sequential game [68] where one player makes a decision and then based on that decision, the other player will make a decision. In this model, a recommendation decision will be provided in each stage. In other words, a list of recommended tasks will be given to the worker. Then, after the worker has made a choice, he/she will apply to the recommended set of tasks or part of this set. After that, the employer will decide which workers to hire. The workers who applied to a large number of tasks will have less potential to work properly on each task. Thus, hiring workers who applied to fewer tasks will result in a higher potential for success.

Moreover, adding priority decisions could also be considered. For example, if a worker prefers working on the same type of tasks each time, that could make those tasks easier to perform.

Crowdsourcing is still considered a new and developing approach, especially for macrotasks as mentioned in Chapter 2. This study has provided an intensive theoretical examination of the proposed crowdsourcing recommendation system and evaluated the model using a numerical simulation.

In the future, we plan to implement a framework for macrotask crowdsourcing that applies our recommendation model with strategic rules to improve solution quality (which is the major drawback in the current crowdsourcing platforms).

Chapter 10

Publications

1. Eman Aldhahri, Vivik Shandilya, and Sajjan Shiva. "Towards an Effective Crowdsourcing Recommendation System: A Survey of the State-of-the-Art". In:Service-Oriented System Engineering (SOSE) (2015)

2. Eman Aldhahri, Abdullah Abuhussein, and Sajjan Shiva. "Leveraging Crowdsourcing in Cloud Application Development". In:SoftwareEngineering and Applications(2015).

3. Eman Aldhahri, Vivik Shandilya, and Sajjan Shiva. "Crowdsourcing Multi-Objective Recommendation System" In: The Web Conference(2018)

4. Eman Aldhahri, and Sajjan Shiva. "Crowdsourcing Multi-Objective Recommendation System Cold Start Problem". under preparation.

5. Abdulaziz Alnowiser, Eman Aldhahri, Abdulrahman Alahmadi, and Michelle Zhu. "Enhanced weighted round robin (ewrr) with dvfs technology in cloud energy-aware". In:omputational Science and Computational Intelligence (CSCI)(2014).

6. Eman Aldhahri."Dynamic Voltage and Frequency Scaling Enhanced Task Scheduling Technology Toward Green Cloud Computing". In:In 4th Intl Conf on Applied Computing and Information Technology (2016).

Bibliography

[1] Pär J Ågerfalk et al. "Benefits of global software development: the known and unknown". In: *International Conference on Software Process*. Springer. 2008, pp. 1–9.

[2] Abdulrahman Alamer et al. "Location privacy-aware task recommendation for spatial crowdsourcing". In: *Wireless Communications and Signal Processing (WCSP), 2017 9th International Conference on*. IEEE. 2017, pp. 1–6.

[3] Eman Aldhahri, Abdullah Abuhussein, and Sajjan Shiva. "Leveraging Crowdsourcing in Cloud Application Development". In: *Software Engineering and Applications* (2015).

[4] *Amazon Mechanical Turk (MTurk)*. `https://www.mturk.com/`.

[5] Vamshi Ambati, Stephan Vogel, and Jaime G Carbonell. "Towards task recommendation in micro-task markets." In: *Human computation*. 2011, pp. 1–4.

[6] Debarshi Basak, Babak Loni, and Alessandro Bozzon. "A Platform for Task Recommendation in Human Computation". In: *RecSys 2014 CrowdRec Workshop. ACM*. 2014.

[7] Michael R Baye and John Morgan. "A folk theorem for one-shot Bertrand games". In: *Economics Letters* 65.1 (1999), pp. 59–65.

[8] *"Best Crowdsourcing Websites Pay Per Task Sites"*. `https://ranker.com/`.

[9] David Boland and Brian Fitzgerald. "Transitioning from a co-located to a globally-distributed software development team: A case study at Analog Devices Inc". In: *3rd Workshop on Global Software Development*. IET. 2004.

[10] Jan Bosch and Petra Bosch-Sijtsema. "From integration to composition: On the impact of software product lines, global development and ecosystems". In: *Journal of Systems and Software* 83.1 (2010), pp. 67–76.

[11] Daren C Brabham. “Crowdsourcing as a model for problem solving an introduction and cases”. In: *Convergence: the international journal of research into new media technologies* 14.1 (2008), pp. 75–90.

[12] Bernd Bruegge et al. “Unicase–an ecosystem for unified software engineering research tools”. In: *Third IEEE International Conference on Global Software Engineering, ICGSE*. Vol. 2008. Citeseer. 2008.

[13] Junnan Chen, Courtney Miller, and Gaby G Dagher. “Product recommendation system for small online retailers using association rules mining”. In: *Innovative Design and Manufacturing (ICIDM), Proceedings of the 2014 International Conference on*. IEEE. 2014, pp. 71–77.

[14] *ClickWorker*. `https://www.clickworker.com/`.

[15] *CloudBees enterprise Jenkins and DevOps*. `http://www.cloudbees.com`.

[16] *CloudCrowd | Crunchbase*. `https://www.crunchbase.com/organization/cloudcrowd`.

[17] Eoin Ó Conchúir et al. “Global software development: where are the benefits?” In: *Communications of the ACM* 52.8 (2009), pp. 127–131.

[18] Dan Cosley et al. “SuggestBot: using intelligent task routing to help people find work in wikipedia”. In: *Proceedings of the 12th international conference on Intelligent user interfaces*. ACM. 2007, pp. 32–41.

[19] Daniela Damian and Deependra Moitra. “Guest editors’ introduction: Global software development: How far have we come?” In: *IEEE software* 23.5 (2006), pp. 17–19.

[20] James Davidson et al. “The YouTube video recommendation system”. In: *Proceedings of the fourth ACM conference on Recommender systems*. ACM. 2010, pp. 293–296.

[21] William DeLone et al. “Bridging global boundaries for IS project success”. In: *System Sciences, 2005. HICSS’05. Proceedings of the 38th Annual Hawaii International Conference on*. IEEE. 2005, 48b–48b.

[22] Djellel Eddine Difallah, Gianluca Demartini, and Philippe Cudré-Mauroux. “Pick-a-crowd: tell me what you like, and i’ll tell you what to do”. In: *Proceedings of the 22nd international conference on World Wide Web*. ACM. 2013, pp. 367–374.

[23] Gideon Dror et al. “I want to answer; who has a question?: Yahoo! answers recommender system”. In: *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2011, pp. 1109–1117.

[24] Thomas Erl. *SOA design patterns*. Pearson Education, 2008.

[25] J Alberto Espinosa and Erran Carmel. “The effect of time separation on coordination costs in global software teams: A dyad model”. In: *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*. IEEE. 2004, 10–pp.

[26] Ju Fan et al. “icrowd: An adaptive crowdsourcing framework”. In: *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*. ACM. 2015, pp. 1015–1030.

[27] Joseph Feller, Brian Fitzgerald, et al. *Understanding open source software development*. Addison-Wesley London, 2002.

[28] Zhi Ming Feng and Yi Dan Su. “Application of Using Simulated Annealing to Combine Clustering with Collaborative Filtering for Item Recommendation”. In: *Applied Mechanics and Materials*. Vol. 347. Trans Tech Publ. 2013, pp. 2747–2751.

[29] *freedomsponsors*. `https://freedomsponsors.org/`.

[30] Jerry Gao, Xiaoying Bai, and Wei-Tek Tsai. “Cloud testing-issues, challenges, needs and practice”. In: *Software Engineering: An International Journal* 1.1 (2011), pp. 9–23.

[31] David Geiger and Martin Schader. “Personalized task recommendation in crowdsourcing information systems—Current state of the art”. In: *Decision Support Systems* 65 (2014), pp. 3–16.

[32] David Geiger et al. “Crowdsourcing information systems-definition, typology, and design”. In: (2012).

[33] Dorina C Gumm. “Distribution dimensions in software development projects: A taxonomy”. In: *IEEE software* 23.5 (2006), pp. 45–51.

[34] Sami ul Haq et al. “Issues in global software development: A critical review”. In: *Journal of Software Engineering and Applications* 4.10 (2011), p. 590.

[35] Sajid Ibrahim Hashmi et al. “Using the cloud to facilitate global software development challenges”. In: *Global Software Engineering Workshop (ICGSEW), 2011 Sixth IEEE International Conference on*. IEEE. 2011, pp. 70–77.

[36] Matthew J Hawthorne and Dewayne E Perry. “Software engineering education in the era of outsourcing, distributed development, and open source software: challenges and opportunities”. In: *Proceedings of the 27th international conference on Software engineering*. ACM. 2005, pp. 643–644.

[37] James D Herbsleb and Rebecca E Grinter. “Splitting the organization and integrating the code: Conway’s law revisited”. In: *Proceedings of the 21st international conference on Software engineering*. ACM. 1999, pp. 85–95.

[38] James D Herbsleb and Deependra Moitra. “Global software development”. In: *IEEE software* 18.2 (2001), pp. 16–20.

[39] James D Herbsleb and Deependra Moitra. “Global software development”. In: *IEEE software* 18.2 (2001), pp. 16–20.

[40] Alex Homer et al. *Cloud design patterns: Prescriptive architecture guidance for cloud applications*. Microsoft patterns & practices, 2014.

[41] Mahmoud Hosseini et al. “Towards crowdsourcing for requirements engineering”. In: (2014).

[42] Jeff Howe. “The rise of crowdsourcing”. In: *Wired magazine* 14.6 (2006), pp. 1–4.

[43] Chein-Shung Hwang, Yi-Ching Su, and Kuo-Cheng Tseng. “Using genetic algorithms for personalized recommendation”. In: *International Conference on Computational Collective Intelligence*. Springer. 2010, pp. 104–112.

[44] *IdeaStorm*. `http://www.ideastorm.com/`.

[45] *InnoCentive is an open innovation and crowdsourcing company.* `https://www.innocentive.com/`.

[46] François Jammes, Antoine Mensch, and Harm Smit. “Service-oriented device communications using the devices profile for web services”. In: *Proceedings of the 3rd international workshop on Middleware for pervasive and ad-hoc computing*. ACM. 2005, pp. 1–8.

[47] Debasish Jana and Debasis Bandyopadhyay. “Management of security and privacy issues of application development in mobile cloud environment: A survey”. In: *Recent Advances and Innovations in Engineering (ICRAIE), 2014*. IEEE. 2014, pp. 1–6.

[48] BJD Kalyani. “Challenges in the Cloud Application Development”. In: *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)* 2.1 (2013), pp–310.

[49] Qiyu Kang and Wee Peng Tay. “Task Recommendation in Crowdsourcing Based on Learning Preferences and Reliabilities”. In: *arXiv preprint arXiv:1807.10444* (2018).

[50] Muhammad Rezaul Karim et al. “Learn or Earn?-Intelligent Task Recommendation for Competitive Crowdsourced Software Development”. In: (2018).

[51] Hanieh Kashfi. “Software Engineering Challenges in Cloud Environment: Software Development Lifecycle Perspective”. In: (2017).

[52] Staffs Keele et al. “Guidelines for performing systematic literature reviews in software engineering”. In: *Technical report, Ver. 2.3 EBSE Technical Report. EBSE*. sn, 2007.

[53] Yehuda Koren. “The bellkor solution to the netflix grand prize”. In: *Netflix prize documentation* 81 (2009), pp. 1–10.

[54] Sang-Gun Lee, Seung Hoon Chae, and Kyung Min Cho. “Drivers and inhibitors of SaaS adoption in Korea”. In: *International Journal of Information Management* 33.3 (2013), pp. 429–440.

[55] Guoliang Li et al. “Crowdsourced data management: A survey”. In: *IEEE Transactions on Knowledge and Data Engineering* 28.9 (2016), pp. 2296–2319.

[56] Wu-Jun Li and Dit Yan Yeung. “Relation regularized matrix factorization”. In: *21ST INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE (IJCAI-09), PROCEEDINGS*. 2009, p. 1126.

[57] Xin Li, Mengyue Wang, and T-P Liang. “A multi-theoretical kernel-based approach to social network-based recommendation”. In: *Decision Support Systems* 65 (2014), pp. 95–104.

[58] Soo Ling Lim, Daniele Quercia, and Anthony Finkelstein. “StakeSource: harnessing the power of crowdsourcing and social networks in stakeholder analysis”. In: *Proceedings of the 32Nd ACM/IEEE International Conference on Software Engineering-Volume 2*. ACM. 2010, pp. 239–242.

[59] Chen Lin et al. “Premise: Personalized news recommendation via implicit social experts”. In: *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM. 2012, pp. 1607–1611.

[60] Christopher H Lin, Ece Kamar, and Eric Horvitz. “Signals in the Silence: Models of Implicit Feedback in a Recommendation System for Crowdsourcing.” In: *AAAI*. 2014, pp. 908–915.

[61] Jovian Lin et al. “Addressing cold-start in app recommendation: latent user models constructed from twitter followers”. In: *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. ACM. 2013, pp. 283–292.

[62] Kevin Meehan et al. “Context-aware intelligent recommendation system for tourism”. In: *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on*. IEEE. 2013, pp. 328–331.

[63] Michael P Papazoglou et al. “Service-oriented computing: State of the art and research challenges”. In: *Computer* 40.11 (2007).

[64] Seung-Taek Park and Wei Chu. “Pairwise preference regression for cold-start recommendation”. In: *Proceedings of the third ACM conference on Recommender systems*. ACM. 2009, pp. 21–28.

[65] David Lorge Parnas. "On the criteria to be used in decomposing systems into modules". In: *Communications of the ACM* 15.12 (1972), pp. 1053–1058.

[66] Michael J Pazzani. "A framework for collaborative, content-based and demographic filtering". In: *Artificial intelligence review* 13.5-6 (1999), pp. 393–408.

[67] Sanjay Purushotham, Yan Liu, and C-C Jay Kuo. "Collaborative topic regression with social matrix factorization for recommendation systems". In: *arXiv preprint arXiv:1206.4684* (2012).

[68] Matthew Rabin. "Incorporating fairness into game theory and economics". In: *The American economic review* (1993), pp. 1281–1302.

[69] Yu Rong, Xiao Wen, and Hong Cheng. "A Monte Carlo algorithm for cold start recommendation". In: *Proceedings of the 23rd international conference on World wide web*. ACM. 2014, pp. 327–336.

[70] Sundeep Sahay, Brian Nicholson, and Srinivas Krishna. *Global IT outsourcing: software development across borders*. Cambridge University Press, 2003.

[71] Eric Schenk and Claude Guittard. "Crowdsourcing: What can be Outsourced to the Crowd, and Why". In: *Workshop on Open Source Innovation, Strasbourg, France*. Vol. 72. 2009.

[72] Thimo Schulze, Simone Krug, and Martin Schader. "Workers' task choice in crowdsourcing and human computation markets". In: (2012).

[73] Suprika Vasudeva Shrivastava et al. "Distributed agile software development: A review". In: *arXiv preprint arXiv:1006.1955* (2010).

[74] Dava Sobel. *Longitude: The true story of a lone genius who solved the greatest scientific problem of his time*. Bloomsbury Publishing USA, 2007.

[75] Borja Sotomayor et al. "Virtual infrastructure management in private and hybrid clouds". In: *IEEE Internet computing* 13.5 (2009).

[76] Klaas-Jan Stol and Brian Fitzgerald. "Two's company, three's a crowd: a case study of crowdsourcing software development". In: *Proceedings of the 36th International Conference on Software Engineering*. ACM. 2014, pp. 187–198.

[77] Dezhi Sun et al. "Online delivery route recommendation in spatial crowdsourcing". In: *World Wide Web* (2018), pp. 1–22.

[78] *The Professional Network for Testers*. `http://www.utest.com/`.

[79] *The SaaS Development Lifecycle.InfoQ*. `http://www.infoq.com/articles/SaaS-Lifecycle`.

[80] *TopCoder. 2010. 10 Burning Questions on Crowdsourcing: Your starting guide to open innovation and crowdsourcing success.* http://www.topcoder.com/blog/10burningquestions-on-crowdsourcing-and-open-innovation.

[81] Paolo Toth. "Dynamic programming algorithms for the zero-one knapsack problem". In: *Computing* 25.1 (1980), pp. 29–45.

[82] *TripAdvisor: Read Reviews, Compare Prices Book.* http://www.tripadvisor.com.

[83] Wei-Tek Tsai, Wenjun Wu, and Michael N Huhns. "Cloud-based software crowdsourcing". In: *IEEE Internet Computing* 18.3 (2014), pp. 78–83.

[84] *Upwork.* :https://www.upwork.com/.

[85] *Vusiness and employment-oriented service that operates via websites and mobile apps.* https://www.linkedin.com/.

[86] Paul Whitla. "Crowdsourcing and its application in marketing activities". In: *Contemporary Management Research* 5.1 (2009).

[87] *Why use crowdsourcing? Three fundamental arguments.* http://www.hypios.com/news/blog/210711/why-usecrowdsourcing-three-fundamental-arguments.

[88] Chunchun Wu et al. "An endorsement-based reputation system for trustworthy crowdsourcing". In: *Computer Communications Workshops (INFOCOM WKSHPS), 2015 IEEE Conference on.* IEEE. 2015, pp. 89–90.

[89] Wenjun Wu, Wei-Tek Tsai, and Wei Li. "Creative software crowdsourcing: from components and algorithm development to project concept formations". In: *International Journal of Creative Computing* 1.1 (2013), pp. 57–91.

[90] Zhenlei Yan and Jie Zhou. "A new approach to answerer recommendation in community question answering services". In: *European Conference on Information Retrieval.* Springer. 2012, pp. 121–132.

[91] Man-Ching Yuen, Irwin King, and Kwong-Sak Leung. "Task matching in crowdsourcing". In: *Internet of Things (iThings/CPSCom), 2011 International Conference on and 4th International Conference on Cyber, Physical and Social Computing.* IEEE. 2011, pp. 409–412.

[92] Man-Ching Yuen, Irwin King, and Kwong-Sak Leung. "Task recommendation in crowdsourcing systems". In: *Proceedings of the First International Workshop on Crowdsourcing and Data Mining.* ACM. 2012, pp. 22–26.

[93] Man-Ching Yuen, Irwin King, and Kwong-Sak Leung. “Task recommendation in crowdsourcing systems”. In: *Proceedings of the First International Workshop on Crowdsourcing and Data Mining*. ACM. 2012, pp. 22–26.

[94] Man-Ching Yuen, Irwin King, and Kwong-Sak Leung. “Taskrec: A task recommendation framework in crowdsourcing systems”. In: *Neural Processing Letters* 41.2 (2015), pp. 223–238.

[95] Zhi-Dan Zhao and Ming-Sheng Shang. “User-based collaborative-filtering recommendation algorithms on hadoop”. In: *Knowledge Discovery and Data Mining, 2010. WKDD’10. Third International Conference on*. IEEE. 2010, pp. 478–481.

[96] Zhou Zhao et al. “Socialtransfer: transferring social knowledge for cold-start cowdsourcing”. In: *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*. ACM. 2014, pp. 779–788.

[97] Ke Zhou, Shuang-Hong Yang, and Hongyuan Zha. “Functional matrix factorizations for cold-start recommendation”. In: *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM. 2011, pp. 315–324.

Chapter 11

Appendix

Appendix A: Experimental code

Java code for case 0 and case1 experiment.

Class Worker is for Case 0 :

worker.java

```
package Recommendation;

import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.util.Random;

import org.apache.poi.xssf.usermodel.XSSFSheet;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;
import java.util.Arrays;


public class worker {


    // Case 0: Workers works on one task at a time
    public static void main(String[] args) throws Exception {
        // TODO Auto-generated method stub

        double prof[] =  new double[5];
        int rating[] = new int[5];
        int TW[] =  new int[5];


        double EmployerComm[] =  new double[50];
        int EmployerRating[] = new int[50];


        int TaskType[] = new int[1000];
        int TaskEmployer[] = new int[1000];
        int TaskReward[] = new int[1000];



File src = new File("C:\\Users\\Eiman\\Documents\\workspace\\Book1.xlsx");

        FileInputStream fis = new FileInputStream(src);

        XSSFWorkbook wb = new XSSFWorkbook(fis);

        XSSFSheet sheet1 = wb.getSheetAt(0);

        XSSFSheet sheet2 = wb.getSheetAt(1);

        XSSFSheet sheet3 = wb.getSheetAt(2);

        XSSFSheet sheet4 = wb.getSheetAt(3);

        XSSFSheet sheet5 = wb.getSheetAt(4);

         Random rr = new Random();
         int worker1= 1+ rr.nextInt(50);

        System.out.println(worker1);

         for(int i = 1 ; i<=5; i++)
            {
```

```java
                double data0 = sheet1.getRow(worker1).getCell(i).getNumericCellValue();
                data0 = round(data0, 2);
                prof[i-1]= data0;
                //System.out.println("The data is" + prof[i-1]);
            }


        for(int i = 1 ; i<=5; i++)
            {
                double data1 = sheet2.getRow(worker1).getCell(i).getNumericCellValue();
                //data1 = round(data1, 2);
                rating[i-1]= (int)data1;
                //System.out.println("The data is" + rating[i-1]);
            }

        for(int i = 1 ; i<=5; i++)
            {
                double data2 = sheet3.getRow(worker1).getCell(i).getNumericCellValue();
                //data2 = round(data2, 2);
                TW[i-1]= (int)data2;
                //System.out.println("The data is" + TW[i-1]);
            }

        for(int i = 1 ; i<=50; i++)
        {
             double data3 = sheet4.getRow(i).getCell(1).getNumericCellValue();
                data3 = round(data3, 2);
                EmployerComm[i-1]= data3;
                //System.out.println("The data is" + EmployerComm[i-1]);

        }


        for(int i = 1 ; i<=50; i++)
        {
             double data4 = sheet4.getRow(i).getCell(2).getNumericCellValue();
                //data3 = round(data3, 2);
             EmployerRating[i-1]= (int)data4;
                //System.out.println("The data is" + EmployerRating[i-1]);

        }


        for(int i = 1 ; i<=1000; i++)
        {
             double data5 = sheet5.getRow(i).getCell(1).getNumericCellValue();
                //data3 = round(data3, 2);
             TaskType[i-1]= (int)data5;
                //System.out.println("The data is" + TaskType[i-1]);

        }

        for(int i = 1 ; i<=1000; i++)
        {
             double data6 = sheet5.getRow(i).getCell(2).getNumericCellValue();
                //data3 = round(data3, 2);
             TaskEmployer[i-1]= (int)data6;
```

```
            //  System.out.println("The data is" + TaskEmployer[i-1]);

        }

        for(int i = 1 ; i<=1000; i++)
        {
            double data6 = sheet5.getRow(i).getCell(3).getNumericCellValue();
               //data3 = round(data3, 2);
            TaskReward[i-1]= (int)data6;
               //System.out.println("The data is" + TaskReward[i-1]);

        }


        int a,b,c,e,g,f;
        double d,h;
        double EXP[] = new double[1000];
        int EXR[] = new int[1000];
        int WTW[] = new int[1000];



        for(int i = 0 ; i<1000; i++)
        {

            a  = TaskType[i];
            b = TaskEmployer[i];
            c=  TaskReward[i];
            d = prof[a-1];

           // System.out.println("test d" + d);

            e = rating[a-1];
            f = TW[a-1];
            h = EmployerComm[b-1];
            g= EmployerRating[b-1];

            EXP[i] = d*h*c;

           // System.out.println("test exp" + EXP[i]);
            EXR[i] = (e+g)/2;
            WTW[i] = f;

           // System.out.println("test WTW" +  WTW[i]);

        }

       ///////////////////////////////////////////////////////
        int taskNumber[] = new int[1000];

        for(int i = 0;i<1000;i++)
        {
            taskNumber[i]= i+1;
        }

        double temp1;
        int temp2,temp4,temp3;
```

```
        for (int i = 1; i < 1000; i++) {
         for (int j = i; j > 0; j--) {
          if (EXP[j] < EXP [j - 1]) {

           temp1 = EXP[j];
           temp2 = EXR[j];
           temp3 = WTW[j];
           temp4 = taskNumber[j];

           EXP[j] = EXP[j - 1];
           EXR[j] = EXR[j - 1];
           WTW[j] = WTW[j - 1];
           taskNumber[j] = taskNumber[j - 1];


           EXP[j - 1] = temp1;
           EXR[j - 1] = temp2;
           WTW[j - 1] = temp3;
           taskNumber[j - 1] = temp4;

          }
         }
        }

        for(int i =0;i<1000;i++)
        {
            System.out.println("test exp" + EXP[i]);
        }

        System.out.println("Worker recommended task is task number  "+ taskNumber[1000-1]);
       // System.out.println("with expected payment  "+ EXP[0]);



        ////////////////////////////////////////////////////////



       wb.close();

    }

    public static double round(double value, int places) {
        if (places < 0) throw new IllegalArgumentException();

        long factor = (long) Math.pow(10, places);
        value = value * factor;
        long tmp = Math.round(value);
        return (double) tmp / factor;
    }

}
```

Class case 1 is for Case 1.

Case1.java

```
package Recommendation;

import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.util.Random;

import org.apache.poi.xssf.usermodel.XSSFSheet;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;
import java.util.Arrays;


public class Case1 {

    // Case 1: Workers works on a set of tasks at the same time

    public static void main(String[] args) throws Exception {
        // TODO Auto-generated method stub

        double prof[] =  new double[5];
        int rating[] = new int[5];
        int TW[] =  new int[5];


        double EmployerComm[] =  new double[50];
        int EmployerRating[] = new int[50];


        int TaskType[] = new int[1000];
        int TaskEmployer[] = new int[1000];
        int TaskReward[] = new int[1000];



File src = new File("C:\\Users\\Eiman\\Documents\\workspace\\Book1.xlsx");

        FileInputStream fis = new FileInputStream(src);

        XSSFWorkbook wb = new XSSFWorkbook(fis);

        XSSFSheet sheet1 = wb.getSheetAt(0);

        XSSFSheet sheet2 = wb.getSheetAt(1);

        XSSFSheet sheet3 = wb.getSheetAt(2);

        XSSFSheet sheet4 = wb.getSheetAt(3);

        XSSFSheet sheet5 = wb.getSheetAt(4);

         Random rr = new Random();
         int worker1= 1+ rr.nextInt(50);

        System.out.println(worker1);

         for(int i = 1 ; i<=5; i++)
            {
```

```
                double data0 = sheet1.getRow(worker1).getCell(i).getNumericCellValue();
                data0 = round(data0, 2);
                prof[i-1]= data0;
                //System.out.println("The data is" + prof[i-1]);
            }


        for(int i = 1 ; i<=5; i++)
            {
                double data1 = sheet2.getRow(worker1).getCell(i).getNumericCellValue();
                //data1 = round(data1, 2);
                rating[i-1]= (int)data1;
                //System.out.println("The data is" + rating[i-1]);
            }

        for(int i = 1 ; i<=5; i++)
            {
                double data2 = sheet3.getRow(worker1).getCell(i).getNumericCellValue();
                //data2 = round(data2, 2);
                TW[i-1]= (int)data2;
            //  System.out.println("The data is" + TW[i-1]);
            }

        for(int i = 1 ; i<=50; i++)
        {
             double data3 = sheet4.getRow(i).getCell(1).getNumericCellValue();
                data3 = round(data3, 2);
                EmployerComm[i-1]= data3;
            //  System.out.println("The data is" + EmployerComm[i-1]);

        }


        for(int i = 1 ; i<=50; i++)
        {
             double data4 = sheet4.getRow(i).getCell(2).getNumericCellValue();
                //data3 = round(data3, 2);
             EmployerRating[i-1]= (int)data4;
            //  System.out.println("The data is" + EmployerRating[i-1]);

        }


        for(int i = 1 ; i<=1000; i++)
        {
             double data5 = sheet5.getRow(i).getCell(1).getNumericCellValue();
                //data3 = round(data3, 2);
             TaskType[i-1]= (int)data5;
                //System.out.println("The data is" + TaskType[i-1]);

        }

        for(int i = 1 ; i<=1000; i++)
        {
             double data6 = sheet5.getRow(i).getCell(2).getNumericCellValue();
                //data3 = round(data3, 2);
             TaskEmployer[i-1]= (int)data6;
```

```
                //System.out.println("The data is" + TaskType[i-1]);

        }

        for(int i = 1 ; i<=1000; i++)
        {
            double data6 = sheet5.getRow(i).getCell(3).getNumericCellValue();
               //data3 = round(data3, 2);
            TaskReward[i-1]= (int)data6;
               //System.out.println("The data is" + TaskType[i-1]);

        }


        int a,b,c,e,g,f;
        double d,h,t;
        int EXP[] = new int[1000];
        int EXR[] = new int[1000];
        int WTW[] = new int[1000];



        for(int i = 0 ; i<1000; i++)
        {

            a  = TaskType[i];
            b = TaskEmployer[i];
            c=  TaskReward[i];
            d = prof[a-1];
            e = rating[a-1];
            f = TW[a-1];
            h = EmployerComm[b-1];
            g= EmployerRating[b-1];

           // EXP[i] = d*h*c;
            t = d*h*c;
            EXP[i] = (int)t;
            EXR[i] = (e+g)/2;
            WTW[i] = f;

        }

       ///////////////////////////////////////////////////////
        int taskNumber[] = new int[1000];

        for(int i = 0;i<1000;i++)
        {
            taskNumber[i]= i+1;
        }


        ////////////////////////////////////////////////////////
        // Dynamic Programing //

        int answer;

       answer= knapSack(100, WTW, EXP, 1000);
```

```java

        System.out.println("Worker expexted rewards  "+ answer);


       wb.close();

    }

    public static double round(double value, int places) {
        if (places < 0) throw new IllegalArgumentException();

        long factor = (long) Math.pow(10, places);
        value = value * factor;
        long tmp = Math.round(value);
        return (double) tmp / factor;
    }


    ///////////////////////////////////////

    // A utility function that returns maximum of two integers
    static int max(int a, int b) { return (a > b)? a : b; }

// Returns the maximum value that can be put in a knapsack of capacity W
    static int knapSack(int W, int wt[], int val[], int n)
    {
        int i, w;
    int K[][] = new int[n+1][W+1];

    // Build table K[][] in bottom up manner
    for (i = 0; i <= n; i++)
    {
        for (w = 0; w <= W; w++)
        {
            if (i==0 || w==0)
                K[i][w] = 0;
            else if (wt[i-1] <= w)
                K[i][w] = max(val[i-1] + K[i-1][w-wt[i-1]], K[i-1][w]);
            else
                K[i][w] = K[i-1][w];
        }
    }

    return K[n][W];
    }


    /////////////////////////////////////

}

```

Appendix B: Dataset

Datasets

Dataset D1

Workers	Programmi	Analysis	Web desigr	Algorithm	Graphics
1	0.13	0.04	0.75	0.42	0.56
2	0.70	0.95	0.49	0.98	0.80
3	0.00	0.68	0.62	0.94	0.03
4	0.53	0.28	0.35	0.72	0.37
5	0.77	0.12	0.73	0.01	0.16
6	0.68	0.17	0.49	0.88	0.15
7	0.26	0.54	0.23	0.85	0.09
8	0.38	0.80	0.45	0.31	0.04
9	0.49	0.63	0.42	0.94	0.81
10	0.17	0.67	0.21	1.00	0.35
11	0.42	0.24	0.36	0.48	0.66
12	0.33	0.07	0.08	0.04	0.96
13	0.37	0.69	0.72	0.89	0.76
14	0.06	0.68	0.87	0.60	0.69
15	0.86	0.00	0.15	0.82	0.24
16	0.52	0.28	0.99	0.29	0.97
17	0.82	0.90	0.49	0.35	0.85
18	0.25	0.63	0.21	0.76	0.99
19	0.07	0.35	0.98	0.31	0.80
20	0.47	0.90	0.60	0.15	0.40
21	0.66	0.10	0.15	0.15	0.82
22	0.47	0.51	0.20	0.09	0.38
23	0.66	0.18	0.62	0.36	0.49
24	0.28	0.23	0.28	0.13	0.60
25	0.28	0.10	0.17	0.07	0.86
26	0.01	0.41	0.42	0.95	0.12
27	1.00	0.23	0.98	0.34	0.44
28	0.82	0.46	0.68	0.30	0.05
29	0.66	0.44	0.99	0.91	0.30
30	0.75	0.77	0.68	0.32	0.30
31	0.54	0.48	0.61	0.56	0.61
32	0.89	0.24	0.77	0.42	0.53
33	0.47	0.95	0.20	0.84	0.36
34	0.90	0.38	0.33	0.01	0.19
35	0.34	0.28	0.73	0.35	0.05
36	0.74	0.87	0.53	0.84	0.94
37	0.19	0.18	0.50	0.17	0.72
38	0.93	0.65	0.81	0.26	0.05
39	0.85	0.65	0.78	0.71	0.48
40	0.85	0.03	0.85	0.72	0.15
41	0.30	0.43	0.14	0.51	0.14
42	0.96	0.21	0.11	0.60	0.05
43	0.48	0.66	0.22	0.59	0.63
44	0.43	0.21	0.70	0.30	0.66
45	0.58	0.49	0.56	0.45	0.58
46	0.13	0.27	0.32	0.86	0.35

Workers proficiency Score in five Skills.

Uniform Distribution ranging between 0 - 1.

47	0.91	0.32	0.81	0.28	0.20
48	0.40	0.77	0.04	0.78	0.29
49	0.56	0.75	0.77	0.57	0.25
50	0.28	0.99	0.34	0.20	0.81

Workers	Programmin	Analysis	Web desigr	Algorithm	Graphics
1	4	2	1	3	0
2	4	2	3	1	1
3	4	2	1	5	1
4	0	4	5	2	3
5	1	4	2	4	4
6	5	3	2	5	3
7	2	1	0	2	3
8	5	3	4	1	0
9	2	2	1	0	4
10	0	5	2	2	5
11	4	1	1	4	0
12	1	2	1	0	2
13	2	4	3	2	1
14	0	3	0	3	0
15	3	0	1	2	2
16	5	1	3	4	0
17	1	1	3	4	4
18	3	5	1	5	4
19	1	3	4	1	4
20	0	1	3	1	5
21	3	5	2	1	3
22	4	4	0	4	2
23	1	3	5	3	4
24	4	0	0	1	1
25	3	1	2	1	5
26	5	5	2	0	4
27	4	0	3	0	5
28	2	3	5	3	4
29	4	0	4	4	2
30	4	4	5	3	4
31	2	1	5	1	5
32	1	3	2	0	0
33	3	1	0	4	1
34	0	1	3	4	4
35	5	1	4	4	4
36	2	2	1	3	4
37	5	1	0	5	1
38	5	5	4	4	1
39	3	0	5	4	2
40	2	3	3	3	4
41	2	2	2	3	5
42	3	2	2	3	4
43	4	0	3	2	1
44	3	2	3	3	0
45	5	4	4	3	2
46	4	3	0	0	2

Workers Rating Scores in five skills.

Binomial Distribution ranging between 0 - 5.

47	4	4	1	3	1
48	4	1	1	2	0
49	1	3	5	1	1
50	1	2	1	2	1

Workers	Programmi	Analysis	Web desigr	Algorithm	Graphics
1	100.00	25.00	50.00	50.00	100.00
2	25.00	33.00	33.00	100.00	50.00
3	100.00	33.00	50.00	33.00	33.00
4	100.00	33.00	25.00	33.00	33.00
5	50.00	25.00	50.00	50.00	100.00
6	100.00	33.00	50.00	33.00	100.00
7	100.00	50.00	33.00	50.00	100.00
8	25.00	100.00	33.00	100.00	100.00
9	25.00	100.00	50.00	100.00	25.00
10	100.00	25.00	100.00	100.00	100.00
11	33.00	33.00	33.00	100.00	25.00
12	50.00	50.00	50.00	25.00	33.00
13	25.00	33.00	33.00	33.00	33.00
14	25.00	25.00	100.00	50.00	100.00
15	50.00	100.00	33.00	50.00	25.00
16	100.00	50.00	50.00	25.00	50.00
17	50.00	33.00	50.00	25.00	100.00
18	25.00	25.00	25.00	25.00	100.00
19	33.00	50.00	25.00	33.00	50.00
20	33.00	100.00	50.00	33.00	100.00
21	33.00	25.00	100.00	100.00	50.00
22	100.00	50.00	25.00	33.00	25.00
23	50.00	50.00	25.00	33.00	33.00
24	100.00	50.00	50.00	50.00	100.00
25	33.00	100.00	50.00	50.00	25.00
26	100.00	50.00	50.00	50.00	100.00
27	25.00	50.00	33.00	33.00	25.00
28	25.00	33.00	50.00	100.00	100.00
29	100.00	25.00	33.00	33.00	33.00
30	33.00	50.00	100.00	33.00	33.00
31	50.00	100.00	25.00	33.00	33.00
32	33.00	100.00	25.00	25.00	33.00
33	25.00	100.00	33.00	33.00	100.00
34	100.00	25.00	33.00	100.00	33.00
35	33.00	50.00	25.00	25.00	50.00
36	33.00	33.00	25.00	25.00	25.00
37	100.00	33.00	100.00	50.00	50.00
38	25.00	25.00	50.00	50.00	25.00
39	25.00	33.00	25.00	50.00	33.00
40	100.00	50.00	25.00	33.00	25.00
41	25.00	100.00	100.00	25.00	25.00
42	50.00	33.00	50.00	50.00	33.00
43	100.00	50.00	100.00	50.00	100.00
44	50.00	100.00	100.00	100.00	33.00
45	25.00	50.00	50.00	25.00	25.00
46	33.00	33.00	100.00	33.00	33.00

Workers Task weight for five skills

Random number generation between (100, 25, 33, 50)

47	33.00	100.00	50.00	100.00	50.00
48	50.00	100.00	50.00	33.00	33.00
49	33.00	33.00	25.00	25.00	33.00
50	33.00	25.00	33.00	25.00	50.00

Employer	commetment	Rating
1	0.16	4
2	0.54	4
3	0.16	4
4	0.83	4
5	0.42	5
6	0.90	5
7	0.99	3
8	0.74	3
9	0.27	2
10	0.30	4
11	0.74	2
12	0.09	3
13	0.94	2
14	0.50	4
15	0.45	4
16	0.44	5
17	0.24	4
18	0.13	3
19	0.39	5
20	0.96	3
21	0.31	5
22	0.19	5
23	0.76	4
24	0.79	4
25	0.78	3
26	0.19	4
27	0.51	2
28	0.49	2
29	0.53	5
30	0.43	3
31	0.34	4
32	0.38	5
33	0.68	2
34	0.66	2
35	0.09	5
36	0.15	4
37	0.04	5
38	0.44	5
39	0.78	3
40	0.37	5
41	0.56	3
42	0.27	3
43	0.17	3
44	0.33	3
45	0.09	3
46	0.12	4

Employers Files
commitment score using uniform distribution.

Rating Score using binomial distribution.

47	0.40	4
48	0.85	5
49	0.89	2
50	0.35	5

Tasks	Type	Employer	Reward	Deadline
1	3	31	996	79
2	4	38	948	67
3	4	21	289	114
4	4	42	192	107
5	4	5	801	64
6	3	36	878	119
7	5	14	321	32
8	3	5	756	38
9	1	19	348	104
10	2	12	570	29
11	1	9	977	99
12	1	34	988	76
13	2	15	568	78
14	1	1	710	41
15	5	30	118	90
16	2	2	790	28
17	1	29	158	112
18	3	25	535	64
19	4	14	107	75
20	4	22	507	53
21	3	31	526	81
22	4	15	655	69
23	1	16	342	81
24	5	19	507	86
25	1	16	841	87
26	1	19	483	83
27	4	37	588	92
28	2	4	730	42
29	5	41	416	110
30	1	4	630	42
31	1	1	206	28
32	2	30	881	115
33	3	10	936	78
34	2	43	399	81
35	5	14	366	60
36	5	47	637	25
37	5	29	547	84
38	3	9	768	118
39	4	30	633	106
40	3	45	564	43
41	1	30	220	72
42	3	31	271	92
43	5	44	626	86
44	4	11	401	54
45	1	13	910	116
46	5	24	467	41

Task Lists (1000 tasks)
contain task id,
type : random number generator.
Employer: random number generator.
Rewards: uniform distribution.
Deadline: uniform distribution.

47	4	26	819	49
48	5	7	808	70
49	3	37	224	42
50	2	28	326	56
51	3	23	608	71
52	3	34	744	65
53	5	13	409	86
54	4	11	700	111
55	2	39	298	33
56	3	37	156	41
57	1	11	378	54
58	1	28	919	100
59	4	36	633	70
60	5	14	159	96
61	2	30	702	79
62	2	41	583	73
63	5	19	231	114
64	4	49	487	113
65	4	37	555	33
66	2	8	282	27
67	5	23	743	112
68	1	34	787	119
69	4	32	261	90
70	3	25	815	113
71	1	12	298	36
72	1	32	136	54
73	2	37	132	101
74	5	2	947	28
75	1	46	353	87
76	2	43	466	89
77	2	12	232	81
78	5	2	473	71
79	4	15	208	93
80	4	28	165	37
81	5	24	766	62
82	4	29	520	94
83	4	42	713	96
84	3	15	635	97
85	2	43	523	109
86	4	26	670	64
87	3	9	417	58
88	2	19	399	61
89	5	18	280	66
90	5	50	122	46
91	1	40	289	52
92	4	17	403	47
93	5	25	490	116

94	5	2	682	25
95	1	32	265	76
96	4	25	532	73
97	1	48	161	27
98	4	31	797	82
99	2	10	149	63
100	5	33	181	35
101	1	6	647	75
102	5	23	927	47
103	2	47	720	117
104	4	12	887	109
105	4	27	570	102
106	1	7	470	28
107	2	9	469	58
108	4	25	477	44
109	5	21	879	107
110	5	16	218	49
111	4	9	934	90
112	3	43	139	49
113	5	41	870	35
114	4	31	484	105
115	5	48	741	89
116	3	2	352	57
117	2	32	577	53
118	1	18	767	47
119	3	21	318	22
120	5	1	807	75
121	5	42	257	69
122	2	35	986	115
123	2	12	106	90
124	4	5	939	73
125	2	32	677	113
126	2	40	263	41
127	4	40	721	104
128	4	2	903	54
129	5	36	315	108
130	5	22	482	27
131	4	11	491	98
132	1	30	525	42
133	5	47	415	50
134	5	29	702	115
135	2	10	630	85
136	4	50	544	26
137	3	48	471	53
138	2	8	452	84
139	3	34	862	41
140	3	22	993	111

141	2	31	498	46
142	1	31	686	47
143	3	10	273	49
144	4	6	126	103
145	2	37	773	95
146	3	4	629	22
147	3	40	587	79
148	3	46	434	81
149	4	30	981	33
150	4	4	885	33
151	3	11	546	33
152	1	27	868	86
153	1	17	274	72
154	3	36	243	101
155	3	11	211	22
156	1	40	710	79
157	3	41	160	80
158	4	28	156	22
159	4	44	239	26
160	5	44	504	93
161	2	36	933	89
162	3	14	718	44
163	3	6	554	93
164	4	23	289	42
165	2	18	777	39
166	2	7	327	69
167	3	49	588	118
168	5	6	352	97
169	5	24	826	96
170	5	33	845	93
171	5	10	100	73
172	1	48	239	108
173	4	49	140	64
174	2	35	751	91
175	2	12	931	120
176	3	2	338	75
177	2	23	967	36
178	5	4	606	80
179	4	14	624	88
180	4	1	884	32
181	2	5	149	84
182	2	26	607	47
183	5	42	775	36
184	1	44	883	78
185	4	19	547	52
186	4	34	165	102
187	1	37	312	78

188	5	44	822	99
189	2	47	342	39
190	3	14	878	118
191	5	8	188	112
192	5	9	853	106
193	4	6	384	50
194	1	30	326	120
195	5	24	499	113
196	1	49	110	62
197	5	13	949	83
198	2	20	583	108
199	5	4	283	33
200	3	47	573	95
201	2	14	179	40
202	3	19	844	55
203	4	49	827	78
204	4	23	405	84
205	4	12	658	44
206	1	48	776	59
207	5	12	239	93
208	5	33	902	20
209	2	1	397	22
210	5	35	204	115
211	1	33	806	26
212	4	34	271	103
213	1	11	797	55
214	4	30	107	70
215	1	46	709	21
216	2	12	171	113
217	3	36	648	48
218	5	18	467	20
219	3	6	590	81
220	1	20	950	94
221	2	39	628	33
222	5	7	839	34
223	5	26	164	73
224	5	19	685	85
225	3	19	887	37
226	2	46	967	44
227	4	45	968	113
228	3	20	991	58
229	5	39	848	39
230	1	36	137	91
231	3	5	332	88
232	5	6	481	70
233	5	13	927	59
234	1	6	681	73

235	2	18	767	75
236	5	48	447	115
237	1	15	908	84
238	5	20	817	83
239	2	49	958	99
240	4	27	959	70
241	4	4	961	81
242	3	30	257	70
243	4	45	662	95
244	5	19	231	114
245	5	35	947	108
246	3	2	150	33
247	3	20	977	115
248	2	31	667	29
249	4	24	400	69
250	3	35	102	59
251	1	8	809	42
252	4	14	231	54
253	2	21	227	116
254	4	16	839	29
255	4	39	693	61
256	1	7	451	103
257	1	37	418	59
258	4	17	843	110
259	4	45	381	42
260	4	24	699	114
261	5	5	197	79
262	2	16	951	24
263	1	19	477	82
264	4	47	169	50
265	5	28	320	93
266	1	26	732	53
267	1	44	819	69
268	4	47	588	103
269	4	21	452	97
270	3	31	618	46
271	3	29	478	109
272	2	34	468	36
273	3	16	310	106
274	3	10	835	89
275	5	16	199	87
276	5	31	963	114
277	1	36	220	25
278	3	15	292	21
279	4	20	437	87
280	1	26	618	45
281	5	42	779	86

282	4	26	867	25
283	4	26	309	63
284	5	46	871	97
285	4	30	850	59
286	3	22	469	113
287	2	49	527	66
288	2	45	515	108
289	1	37	804	94
290	1	29	996	112
291	2	50	334	36
292	1	39	967	44
293	2	15	320	90
294	1	41	176	70
295	1	32	493	55
296	3	39	544	43
297	2	12	361	47
298	2	31	612	103
299	3	13	564	58
300	4	31	513	81
301	4	39	452	111
302	1	2	516	86
303	5	50	716	37
304	1	35	203	106
305	4	12	847	55
306	5	28	353	119
307	2	19	805	85
308	1	47	336	78
309	5	40	744	100
310	4	14	678	73
311	5	49	677	49
312	4	44	891	55
313	4	47	315	37
314	1	40	195	32
315	4	1	707	36
316	4	47	939	69
317	3	49	412	50
318	4	38	810	51
319	3	44	248	22
320	1	10	261	28
321	3	47	526	84
322	1	37	810	51
323	3	24	202	24
324	5	22	294	88
325	3	45	545	76
326	3	37	952	45
327	2	36	180	86
328	4	37	629	33

329	1	2	607	91
330	3	31	977	37
331	5	1	959	32
332	4	15	277	109
333	5	46	974	73
334	4	7	537	94
335	1	22	916	33
336	1	23	852	46
337	4	18	730	55
338	1	26	963	111
339	2	1	743	61
340	2	4	179	112
341	5	43	847	45
342	3	42	246	80
343	1	42	611	49
344	2	26	466	33
345	3	21	491	28
346	5	27	164	91
347	4	9	501	47
348	2	44	672	45
349	4	26	230	91
350	4	3	833	60
351	4	35	465	102
352	3	5	784	69
353	5	41	116	36
354	5	10	708	28
355	4	10	740	25
356	5	20	246	114
357	4	9	904	56
358	1	14	712	115
359	1	25	266	97
360	4	25	262	82
361	4	27	867	63
362	1	33	823	69
363	2	42	739	67
364	3	16	298	76
365	3	1	987	42
366	5	33	398	114
367	2	18	353	93
368	2	24	321	103
369	3	41	169	96
370	5	8	459	118
371	4	18	557	81
372	2	6	891	48
373	3	20	463	87
374	1	27	322	87
375	3	42	983	79

376	2	21	939	32
377	5	42	344	26
378	4	25	290	119
379	4	18	464	69
380	5	7	671	25
381	4	6	525	101
382	3	28	102	110
383	4	29	915	118
384	1	50	398	115
385	4	42	363	23
386	1	34	356	48
387	2	41	478	47
388	1	25	953	119
389	1	11	329	64
390	2	36	178	102
391	4	42	609	48
392	5	17	997	32
393	2	15	304	65
394	3	36	187	118
395	2	45	917	35
396	4	15	460	97
397	2	21	163	94
398	5	29	407	44
399	1	2	127	49
400	4	20	930	98
401	2	14	527	33
402	2	22	378	97
403	2	19	366	100
404	1	3	534	47
405	4	4	226	68
406	3	4	125	92
407	1	12	762	71
408	5	22	222	82
409	5	2	963	66
410	4	4	231	38
411	1	28	950	67
412	5	25	102	42
413	1	28	421	113
414	1	33	570	24
415	5	31	465	26
416	3	25	930	29
417	5	17	506	84
418	3	47	458	20
419	4	25	227	91
420	5	22	209	28
421	5	3	259	51
422	2	12	664	81

423	5	32	193	42
424	4	15	704	101
425	4	22	133	64
426	4	40	147	106
427	1	4	287	95
428	5	16	113	117
429	2	41	679	21
430	3	1	563	32
431	4	38	469	77
432	5	31	416	33
433	4	35	722	88
434	5	6	593	108
435	5	21	770	61
436	1	4	238	98
437	5	29	576	47
438	5	13	364	46
439	4	33	748	97
440	4	43	644	90
441	4	28	989	51
442	1	2	684	70
443	5	39	989	96
444	5	4	299	53
445	3	39	645	76
446	2	1	280	81
447	3	38	707	21
448	3	2	525	49
449	4	47	938	24
450	5	5	959	55
451	5	9	373	48
452	5	11	678	94
453	2	50	134	83
454	1	36	474	79
455	5	14	732	22
456	5	22	769	88
457	1	12	924	93
458	4	10	122	57
459	4	2	817	98
460	1	20	772	109
461	3	6	913	31
462	3	42	555	82
463	5	36	736	91
464	5	40	929	87
465	5	21	822	49
466	1	50	221	115
467	4	15	145	102
468	1	4	597	118
469	2	45	855	96

470	5	35	537	52
471	1	29	870	69
472	2	37	846	42
473	1	26	218	44
474	4	16	689	41
475	5	30	991	102
476	2	31	260	44
477	2	17	827	56
478	4	12	567	108
479	5	47	563	38
480	3	49	843	95
481	5	5	506	113
482	2	6	147	67
483	2	16	884	99
484	3	13	886	98
485	2	14	997	40
486	3	24	302	77
487	3	12	368	70
488	4	24	692	67
489	1	15	798	53
490	5	17	571	99
491	4	25	455	86
492	4	18	719	73
493	3	50	846	45
494	2	34	662	52
495	2	44	313	48
496	2	49	939	117
497	4	39	250	69
498	1	38	319	82
499	3	8	946	49
500	3	13	771	53
501	3	50	208	105
502	4	8	735	68
503	2	26	181	43
504	5	39	288	104
505	2	7	499	80
506	5	32	325	81
507	4	24	941	23
508	1	42	248	32
509	3	40	402	33
510	2	9	656	90
511	2	31	784	53
512	3	11	758	62
513	5	3	108	87
514	3	32	948	75
515	3	41	403	115
516	2	37	626	50

517	3	15	994	21
518	4	16	534	108
519	5	15	869	110
520	4	11	791	91
521	5	21	830	107
522	2	19	293	64
523	5	16	328	90
524	4	42	399	39
525	4	32	264	103
526	4	20	855	49
527	5	14	416	44
528	4	38	626	59
529	5	26	483	66
530	5	36	284	79
531	3	45	707	119
532	4	11	285	75
533	3	34	282	79
534	3	37	738	104
535	2	16	465	110
536	4	5	145	118
537	3	16	334	54
538	3	27	712	22
539	2	30	406	85
540	3	6	893	58
541	4	10	404	104
542	5	38	362	63
543	1	20	348	83
544	2	24	862	26
545	5	15	897	46
546	3	31	130	99
547	1	43	753	59
548	5	6	956	87
549	5	42	777	37
550	1	48	345	59
551	4	28	433	108
552	2	19	396	93
553	3	24	549	86
554	1	47	928	84
555	3	13	240	34
556	4	20	627	62
557	4	46	554	59
558	5	23	794	37
559	4	28	485	116
560	1	17	991	27
561	3	18	549	24
562	3	24	554	79
563	4	18	139	20

564	3	7	831	87
565	1	49	729	38
566	4	45	917	24
567	1	46	834	107
568	1	23	605	104
569	5	33	276	30
570	3	14	410	100
571	5	2	329	107
572	3	47	490	94
573	3	48	774	80
574	4	28	992	114
575	1	32	960	86
576	2	17	761	39
577	5	30	679	59
578	3	46	901	64
579	5	24	633	85
580	3	17	601	75
581	2	15	796	118
582	3	4	401	70
583	2	37	887	67
584	5	50	557	81
585	5	36	636	85
586	5	49	658	98
587	4	48	472	30
588	4	4	778	29
589	5	26	208	82
590	5	47	732	32
591	3	22	892	42
592	5	24	248	112
593	1	25	631	93
594	2	1	807	58
595	3	43	431	51
596	4	38	866	118
597	1	27	104	120
598	4	44	434	44
599	1	34	429	109
600	1	15	876	60
601	1	40	235	37
602	1	3	820	54
603	5	4	330	97
604	5	42	401	38
605	2	12	354	90
606	5	1	842	88
607	4	41	755	59
608	5	15	876	32
609	2	1	213	58
610	1	37	310	95

611	5	29	122	36
612	5	30	449	107
613	1	13	410	107
614	5	40	160	34
615	1	50	483	34
616	4	31	437	44
617	4	15	226	67
618	5	16	503	46
619	3	41	886	83
620	2	40	598	52
621	2	43	220	57
622	2	15	610	102
623	5	29	235	41
624	5	2	126	83
625	5	5	263	89
626	1	9	863	94
627	3	40	550	100
628	3	42	111	25
629	3	36	364	74
630	1	37	841	33
631	3	30	216	32
632	1	3	978	65
633	4	10	372	69
634	3	34	785	25
635	3	50	510	55
636	1	33	551	119
637	4	31	491	76
638	2	43	601	79
639	3	43	595	41
640	5	21	647	113
641	5	46	510	41
642	3	19	947	24
643	2	39	677	47
644	4	36	857	105
645	4	38	575	63
646	5	22	570	98
647	2	40	204	95
648	4	28	914	48
649	3	18	506	59
650	2	14	727	71
651	4	29	921	87
652	4	33	195	73
653	5	31	487	87
654	5	50	962	70
655	1	32	740	71
656	3	2	471	90
657	3	22	135	93

658	2	44	681	68
659	5	4	843	47
660	3	24	456	95
661	3	36	570	70
662	2	48	682	115
663	3	20	292	32
664	2	9	849	73
665	1	2	199	87
666	5	48	186	59
667	3	21	547	70
668	5	44	332	91
669	2	7	532	97
670	1	11	645	36
671	2	18	472	112
672	5	49	325	70
673	4	3	135	71
674	5	29	140	83
675	4	33	632	42
676	5	1	746	119
677	5	11	256	62
678	5	35	468	61
679	5	37	159	77
680	5	25	985	37
681	4	4	217	52
682	2	6	831	20
683	3	5	492	27
684	1	18	944	61
685	4	40	711	87
686	4	47	211	110
687	2	32	161	32
688	2	19	144	58
689	5	8	287	52
690	3	9	362	65
691	1	42	333	63
692	3	14	553	41
693	5	2	485	89
694	2	13	889	66
695	4	16	781	106
696	5	17	340	117
697	3	35	386	115
698	1	24	848	105
699	5	28	596	23
700	1	3	402	39
701	4	36	531	27
702	3	30	145	60
703	3	27	403	92
704	5	43	981	69

705	1	5	607	79
706	3	35	762	110
707	2	20	600	38
708	3	50	301	26
709	1	5	917	95
710	4	24	184	75
711	2	50	488	48
712	2	35	458	32
713	5	9	153	106
714	2	15	459	43
715	1	14	895	79
716	2	9	985	114
717	4	41	393	89
718	4	37	971	51
719	2	16	782	64
720	4	33	622	91
721	4	25	114	77
722	4	21	147	75
723	5	32	321	108
724	1	27	323	106
725	5	48	577	31
726	1	8	287	113
727	5	17	934	94
728	4	1	120	108
729	3	31	952	45
730	2	40	233	112
731	3	34	390	85
732	1	14	885	72
733	2	31	343	58
734	2	46	659	107
735	2	7	435	41
736	4	1	898	51
737	1	41	996	59
738	3	49	424	43
739	3	2	773	39
740	1	16	975	34
741	1	34	226	69
742	3	26	637	46
743	5	39	456	115
744	2	39	860	35
745	2	26	461	45
746	2	17	295	92
747	5	17	303	110
748	5	7	625	57
749	5	25	642	42
750	2	8	146	74
751	2	27	413	49

752	3	49	597	72
753	3	37	165	46
754	2	19	417	87
755	2	40	789	34
756	1	6	950	59
757	3	32	215	110
758	1	3	992	41
759	4	4	934	73
760	3	49	773	36
761	1	11	370	34
762	5	32	445	36
763	4	2	142	117
764	5	5	469	33
765	2	40	525	93
766	4	3	738	84
767	4	28	643	60
768	3	18	977	39
769	2	5	627	73
770	4	28	477	26
771	1	49	919	45
772	3	42	986	77
773	4	25	155	104
774	4	35	638	32
775	5	16	258	29
776	2	21	933	24
777	2	4	640	59
778	3	34	265	24
779	4	49	966	76
780	3	2	535	45
781	3	5	566	110
782	1	36	632	114
783	1	23	493	107
784	1	44	477	61
785	3	17	340	105
786	5	18	795	95
787	1	7	393	84
788	1	5	385	100
789	2	12	373	56
790	5	45	307	83
791	1	4	988	62
792	5	27	999	68
793	2	41	968	104
794	2	15	131	34
795	5	34	543	72
796	1	25	909	87
797	5	38	539	55
798	2	41	422	52

799	2	12	231	91
800	2	33	234	32
801	1	28	878	47
802	5	48	614	104
803	1	11	385	41
804	3	26	718	117
805	5	49	680	40
806	4	21	487	46
807	2	15	116	77
808	2	17	233	54
809	1	34	445	102
810	3	47	432	97
811	4	17	215	25
812	2	22	853	59
813	2	13	880	70
814	3	31	515	58
815	5	6	690	101
816	3	7	587	96
817	3	20	793	60
818	1	29	144	99
819	2	22	971	90
820	3	13	543	54
821	3	25	927	114
822	3	24	631	28
823	4	3	432	21
824	3	25	311	49
825	1	27	825	44
826	5	2	775	67
827	5	13	960	94
828	5	44	724	32
829	4	41	551	81
830	2	49	342	26
831	4	43	230	117
832	3	15	299	108
833	3	38	678	72
834	1	5	669	26
835	5	7	797	106
836	4	14	918	43
837	4	42	677	29
838	3	12	175	52
839	1	9	281	91
840	5	9	935	62
841	2	18	782	95
842	3	22	103	75
843	1	7	356	96
844	2	36	281	110
845	4	33	299	118

846	4	14	272	47
847	1	39	699	33
848	3	1	265	96
849	2	23	131	88
850	2	25	167	97
851	3	12	337	76
852	4	17	108	77
853	5	36	657	45
854	5	32	448	118
855	1	43	432	51
856	2	31	842	105
857	1	24	974	107
858	3	29	404	98
859	4	4	340	115
860	5	32	601	91
861	2	20	526	23
862	4	40	821	73
863	5	39	798	61
864	2	49	922	23
865	4	27	113	107
866	1	22	615	45
867	3	19	658	110
868	5	2	903	77
869	2	4	628	40
870	1	42	285	68
871	5	25	133	41
872	4	6	625	59
873	1	10	109	26
874	4	46	524	21
875	3	13	575	117
876	5	17	589	35
877	5	21	943	65
878	1	45	969	75
879	1	38	243	47
880	1	33	566	53
881	2	49	814	29
882	4	11	908	94
883	2	43	382	55
884	2	22	826	51
885	4	18	844	26
886	1	50	638	48
887	4	33	380	92
888	3	39	137	36
889	3	15	902	112
890	4	24	784	95
891	2	33	807	60
892	5	1	691	117

893	2	20	237	50
894	2	28	970	112
895	2	41	324	86
896	2	43	863	110
897	2	39	418	74
898	5	24	567	115
899	5	45	750	110
900	1	12	351	63
901	4	23	817	59
902	2	38	190	67
903	4	31	877	38
904	3	28	409	37
905	3	12	740	46
906	2	36	384	105
907	4	31	535	109
908	4	12	310	101
909	2	11	234	36
910	1	40	665	89
911	2	9	873	28
912	4	36	661	42
913	4	36	157	89
914	1	9	752	96
915	3	25	779	33
916	4	3	311	87
917	2	23	775	50
918	4	48	867	64
919	3	22	598	27
920	4	42	159	22
921	5	10	347	45
922	1	8	223	28
923	4	24	465	44
924	5	25	611	91
925	4	28	157	44
926	5	20	493	119
927	2	6	822	96
928	3	10	601	82
929	3	45	383	63
930	2	37	779	95
931	2	45	532	58
932	1	37	210	77
933	2	36	431	84
934	4	1	752	71
935	1	9	149	35
936	5	45	406	65
937	5	31	741	73
938	4	28	308	98
939	3	26	378	80

940	5	50	144	102
941	2	1	796	73
942	3	46	708	89
943	5	37	569	98
944	5	37	535	113
945	5	4	225	39
946	4	12	925	113
947	4	40	775	105
948	3	3	141	24
949	5	22	438	25
950	5	44	494	69
951	4	30	534	77
952	3	39	984	22
953	5	34	202	56
954	3	4	768	69
955	2	43	593	73
956	2	4	344	119
957	4	22	564	106
958	2	6	584	119
959	3	9	737	37
960	3	31	531	28
961	1	45	852	27
962	2	27	748	47
963	5	13	842	94
964	3	48	891	21
965	4	33	260	33
966	4	21	725	58
967	4	8	113	23
968	4	26	652	90
969	3	32	230	91
970	1	21	166	76
971	1	12	725	33
972	2	38	256	61
973	1	25	560	33
974	1	50	337	28
975	3	5	940	25
976	2	34	991	95
977	2	1	638	117
978	3	6	993	58
979	4	14	998	56
980	4	36	460	28
981	4	49	526	74
982	3	25	369	116
983	4	36	566	106
984	3	41	806	72
985	4	36	712	62
986	2	43	110	96

987	4	23	251	41
988	2	31	238	41
989	5	12	637	51
990	2	30	327	93
991	2	23	341	95
992	2	31	962	117
993	1	50	221	88
994	3	28	110	34
995	1	17	297	49
996	3	25	471	28
997	4	33	898	36
998	1	23	355	60
999	2	27	259	37
1000	4	39	197	37

Dataset D2

Workers proficiency Score in Ten Skills.

Uniform Distribution ranging between 0 - 1.

Workers	Skill 1	Skill 2	Skill 3	Skill 4	Skill 5	Skill 6	Skill 7	Skill 8
1	0.559984	0.913785	0.86108	0.249214	0.26899	0.911313	0.590747	0.098239
2	0.772851	0.40141	0.185644	0.261299	0.859401	0.683493	0.557909	0.854091
3	0.983215	0.078066	0.68685	0.799005	0.158116	0.556261	0.820276	0.564867
4	0.464949	0.587725	0.357494	0.825007	0.411054	0.734733	0.904416	0.003052
5	0.202277	0.064394	0.525468	0.247108	0.628162	0.214728	0.449843	0.960845
6	0.379681	0.244087	0.873959	0.727927	0.279275	0.010987	0.685202	0.818873
7	0.368206	0.496353	0.594989	0.801325	0.228645	0.565294	0.06888	0.905728
8	0.273537	0.003876	0.661214	0.261757	0.742088	0.617695	0.0347	0.597369
9	0.145451	0.273659	0.688314	0.654439	0.407605	0.18128	0.127812	0.360607
10	0.66393	0.830775	0.688894	0.206763	0.421949	0.228034	0.352886	0.971679
11	0.511826	0.329295	0.663533	0.702109	0.961089	0.675466	0.236183	0.689596
12	0.839747	0.016968	0.64388	0.229163	0.252937	0.669118	0.195379	0.622486
13	0.496628	0.825556	0.989532	0.698111	0.182165	0.883236	0.136631	0.844539
14	0.338084	0.020783	0.593982	0.372814	0.881771	0.547838	0.418195	0.773919
15	0.67568	0.364757	0.852077	0.908536	0.115757	0.456801	0.071474	0.726096
16	0.682058	0.666311	0.376751	0.70925	0.532182	0.050417	0.675985	0.568102
17	0.290628	0.893582	0.866482	0.321299	0.077364	0.721366	0.49501	0.687979
18	0.02646	0.919156	0.824641	0.229072	0.10886	0.612293	0.663533	0.665426
19	0.299631	0.911222	0.312265	0.413892	0.362255	0.950621	0.212256	0.397351
20	0.041658	0.655934	0.33079	0.214667	0.0112	0.235908	0.109592	0.68276
21	0.620685	0.977325	0.627186	0.007721	0.978973	0.978698	0.167211	0.460463
22	0.498856	0.691702	0.792962	0.75869	0.405499	0.955809	0.104434	0.492782
23	0.426435	0.977477	0.29841	0.466689	0.415448	0.904019	0.385784	0.99823
24	0.234321	0.940306	0.771111	0.906919	0.180456	0.433943	0.49266	0.078677
25	0.708213	0.770684	0.893063	0.053133	0.010102	0.180853	0.12949	0.335368
26	0.085604	0.085177	0.885403	0.234931	0.119358	0.048555	0.416486	0.705222
27	0.581133	0.337352	0.399884	0.264992	0.580401	0.264107	0.253517	0.773186
28	0.636555	0.655324	0.184088	0.637379	0.580432	0.229255	0.276711	0.745415
29	0.068484	0.993225	0.941588	0.065523	0.682455	0.794671	0.219001	0.674917
30	0.80578	0.058809	0.558885	0.846034	0.815699	0.32667	0.012329	0.724418
31	0.511917	0.260353	0.877041	0.707663	0.361126	0.713462	0.304392	0.432386
32	0.08829	0.05182	0.394391	0.819575	0.599872	0.102298	0.313944	0.127201
33	0.428358	0.215247	0.981048	0.975036	0.627155	0.295511	0.422681	0.921873
34	0.239082	0.433241	0.642384	0.790368	0.708945	0.114566	0.800897	0.394238
35	0.698508	0.189306	0.382305	0.733116	0.290445	0.52739	0.702933	0.427015
36	0.992584	0.586901	0.52736	0.536821	0.61156	0.981414	0.937346	0.421827
37	0.551927	0.208747	0.887783	0.334758	0.312204	0.135228	0.76162	0.329295
38	0.673849	0.963408	0.249458	0.498825	0.135685	0.846217	0.158849	0.917081
39	0.034516	0.146916	0.846858	0.27781	0.636586	0.535173	0.498947	0.765862
40	0.893704	0.829646	0.473556	0.793054	0.477371	0.484603	0.956908	0.255806
41	0.630848	0.854579	0.437819	0.194128	0.652669	0.641682	0.427259	0.559221
42	0.56093	0.741569	0.208716	0.140233	0.349528	0.491684	0.146977	0.182287
43	0.589679	0.151769	0.814753	0.384442	0.142979	0.465835	0.104709	0.654073
44	0.815363	0.337992	0.011444	0.312082	0.953612	0.007294	0.853175	0.893002
45	0.395306	0.137761	0.56389	0.325327	0.919187	0.381115	0.326243	0.681753
46	0.805933	0.336589	0.417798	0.066439	0.419782	0.594714	0.501083	0.894131

47	0.858577	0.425642	0.005524	0.492233	0.457869	0.460097	0.445906	0.735496
48	0.139653	0.162023	0.244514	0.792352	0.027955	0.311686	0.949187	0.632771
49	0.601947	0.261879	0.668966	0.084872	0.852016	0.014893	0.689169	0.605914
50	0.930509	0.575701	0.027711	0.352794	0.514664	0.828181	0.430219	0.404035
51	0.401715	0.67275	0.386761	0.440046	0.408918	0.029237	0.69274	0.219794
52	0.308359	0.130955	0.509079	0.022492	0.719687	0.774285	0.84225	0.553728
53	0.054933	0.413831	0.527726	0.691183	0.386364	0.18128	0.900388	0.473342
54	0.63393	0.375286	0.863277	0.090823	0.350749	0.472976	0.752495	0.654866
55	0.285897	0.212775	0.537156	0.213446	0.780541	0.242988	0.977355	0.675466
56	0.127598	0.579119	0.409101	0.765069	0.373394	0.948332	0.876675	0.259468
57	0.816614	0.210211	0.807276	0.199011	0.894986	0.434095	0.204657	0.046052
58	0.786431	0.526627	0.986206	0.589129	0.089541	0.7275	0.400067	0.142277
59	0.229041	0.248665	0.582476	0.967376	0.748863	0.426679	0.003723	0.770226
60	0.957732	0.667898	0.13419	0.813501	0.978729	0.596393	0.938963	0.782464
61	0.763604	0.35667	0.73806	0.953124	0.354839	0.76928	0.447066	0.619678
62	0.385266	0.842952	0.807154	0.138432	0.4626	0.252968	0.82458	0.300272
63	0.856868	0.05005	0.484695	0.355358	0.631489	0.149876	0.956664	0.739921
64	0.568621	0.959716	0.717093	0.129643	0.546251	0.850307	0.35612	0.302469
65	0.145299	0.551347	0.966643	0.800867	0.343944	0.770379	0.502335	0.493271
66	0.505325	0.574175	0.652699	0.497818	0.406384	0.86166	0.334025	0.216376
67	0.976775	0.334635	0.965789	0.696554	0.10773	0.786859	0.615497	0.867031
68	0.942686	0.399915	0.3867	0.467299	0.071627	0.654073	0.179296	0.835658
69	0.300089	0.16361	0.121067	0.846492	0.2743	0.904691	0.54973	0.527146
70	0.774407	0.495346	0.193609	0.757317	0.98822	0.753716	0.892972	0.083773
71	0.275887	0.627247	0.847865	0.012146	0.566027	0.43083	0.079409	0.750511
72	0.806238	0.363414	0.67333	0.388562	0.118717	0.146306	0.632862	0.550523
73	0.186285	0.719687	0.120823	0.027314	0.767449	0.000824	0.850215	0.819636
74	0.405896	0.575243	0.640828	0.734336	0.904355	0.209296	0.416303	0.295358
75	0.348949	0.879513	0.090335	0.461867	0.734336	0.443068	0.010285	0.647603
76	0.170568	0.921842	0.421857	0.078799	0.552995	0.138524	0.011078	0.867336
77	0.883755	0.725913	0.761742	0.293649	0.578997	0.439436	0.181463	0.034394
78	0.318979	0.631306	0.115268	0.559038	0.331858	0.141911	0.089022	0.587085
79	0.462508	0.129795	0.305673	0.212287	0.250038	0.417585	0.799188	0.669271
80	0.664205	0.987701	0.36961	0.866817	0.456709	0.345073	0.803369	0.996612
81	0.722404	0.861232	0.607685	0.673238	0.978118	0.087649	0.587695	0.409467
82	0.377819	0.877102	0.097537	0.605701	0.414899	0.419233	0.742058	0.136174
83	0.961913	0.486038	0.034913	0.268075	0.325755	0.843654	0.563128	0.110416
84	0.937101	0.491745	0.444838	0.856258	0.757134	0.5215	0.328959	0.073641
85	0.193976	0.070162	0.720298	0.001892	0.988403	0.509354	0.59975	0.97882
86	0.281533	0.768578	0.364177	0.425855	0.16718	0.357952	0.81634	0.020844
87	0.72689	0.211585	0.564165	0.297281	0.166784	0.246193	0.378124	0.696097
88	0.918882	0.249794	0.441908	0.574847	0.598468	0.167302	0.663015	0.544939
89	0.303262	0.587359	0.257698	0.601398	0.396222	0.481796	0.582598	0.038392
90	0.135502	0.055544	0.316324	0.83993	0.07181	0.298532	0.945982	0.947264
91	0.582385	0.196142	0.563433	0.981964	0.297037	0.086764	0.654866	0.024964
92	0.21131	0.675375	0.230262	0.04709	0.989746	0.949553	0.134617	0.962462
93	0.514908	0.424879	0.473586	0.685842	0.933073	0.350352	0.365795	0.387646

94	0.235694	0.444472	0.107425	0.842738	0.159459	0.666494	0.691794	0.650655
95	0.070711	0.592578	0.328928	0.925657	0.449171	0.204688	0.06299	0.991424
96	0.397137	0.993988	0.152226	0.197302	0.758721	0.502915	0.178625	0.843013
97	0.9523	0.353893	0.148106	0.777764	0.537828	0.252113	0.813318	0.377178
98	0.09714	0.778527	0.475661	0.480544	0.846431	0.425916	0.456374	0.876461
99	0.491043	0.900967	0.910154	0.648549	0.175665	0.080966	0.910794	0.467116
100	0.546007	0.115818	0.766839	0.9888	0.344249	0.763695	0.821986	0.747673

Skill 9	Skill 10
0.468581	0.616657
0.162084	0.136357
0.669698	0.850124
0.245155	0.81753
0.135441	0.307779
0.379681	0.004669
0.106326	0.875179
0.434462	0.487136
0.767296	0.160161
0.793237	0.576647
0.213904	0.63329
0.241005	0.206763
0.438093	0.771233
0.073763	0.177648
0.799951	0.295236
0.280038	0.523118
0.689474	0.823634
0.934996	0.291452
0.20835	0.314676
0.565081	0.861019
0.844661	0.18601
0.815729	0.382946
0.396435	0.584368
0.902524	0.179205
0.949644	0.570299
0.682028	0.051241
0.654073	0.602405
0.358684	0.597186
0.497391	0.276925
0.479354	0.918912
0.37199	0.429426
0.580218	0.648457
0.59154	0.059267
0.266427	0.163732
0.765099	0.717948
0.249214	0.317423
0.705741	0.638447
0.562151	0.099582
0.956908	0.595233
0.454482	0.51558
0.695151	0.899625
0.93646	0.712119
0.331828	0.681143
0.260048	0.153386
0.215827	0.301279
0.376659	0.555742

0.561174	0.02179
0.252144	0.464156
0.208808	0.528947
0.460494	0.918424
0.209204	0.220557
0.345958	0.735984
0.604877	0.923704
0.866878	0.660207
0.32548	0.547441
0.633381	0.108432
0.906125	0.717704
0.255989	0.631703
0.930021	0.308481
0.553056	0.589709
0.800897	0.607807
0.049654	0.079043
0.796838	0.255989
0.164068	0.950621
0.468032	0.697775
0.143376	0.941435
0.443159	0.321116
0.804102	0.652272
0.727012	0.341258
0.081179	0.447218
0.729392	0.449843
0.913816	0.46263
0.689871	0.916501
0.724906	0.915616
0.97705	0.823817
0.308542	0.919156
0.071261	0.063417
0.948332	0.169195
0.010254	0.247261
0.678304	0.293283
0.813379	0.415601
0.388592	0.464309
0.02765	0.981109
0.597766	0.213111
0.180944	0.356853
0.784997	0.345042
0.710288	0.805383
0.827113	0.600452
0.983795	0.835108
0.937193	0.327647
0.472121	0.591906
0.185186	0.078677
0.082125	0.763695

0.813684	0.820948
0.259743	0.167516
0.776299	0.507736
0.831721	0.042116
0.841975	0.189367
0.806665	0.749779
0.851497	0.801782

Workers	Skill 1	Skill 2	Skill 3	Skill 4	Skill 5	Skill 6	Skill 7	Skill 8
1	0	3	5	2	0	1	5	5
2	3	3	2	2	2	4	1	1
3	2	4	5	1	1	1	1	1
4	4	1	2	3	4	2	0	5
5	2	4	3	0	3	4	1	3
6	1	3	5	4	2	3	3	2
7	3	3	2	1	4	0	0	2
8	4	2	2	1	2	1	4	0
9	5	1	1	4	0	3	0	3
10	4	4	4	1	0	3	1	2
11	3	2	2	4	1	5	3	4
12	2	1	1	5	4	5	3	4
13	0	4	3	5	2	4	1	1
14	4	0	3	3	5	2	2	0
15	4	1	1	4	0	4	5	3
16	2	5	0	2	0	4	2	4
17	4	3	3	4	0	3	3	2
18	0	2	2	4	5	2	4	2
19	3	0	0	2	5	1	4	2
20	2	1	3	2	4	4	2	2
21	2	3	2	0	0	2	2	0
22	2	3	1	0	2	1	2	3
23	5	3	0	4	2	2	1	3
24	3	0	1	1	4	4	5	3
25	2	0	3	1	4	1	4	5
26	2	2	3	5	3	2	5	3
27	2	4	3	1	0	3	4	1
28	0	4	2	4	4	1	0	3
29	4	5	3	4	0	3	4	1
30	1	3	4	2	1	2	3	1
31	3	1	3	5	4	3	1	0
32	3	3	1	4	1	3	3	3
33	5	5	3	3	4	4	1	3
34	3	0	3	3	3	0	4	2
35	1	4	2	3	4	1	2	3
36	1	0	1	0	3	3	2	4
37	5	4	1	5	1	3	3	5
38	4	5	5	4	2	3	4	0
39	3	1	3	4	4	1	3	5
40	4	3	3	4	2	3	4	3
41	0	4	1	2	2	3	5	2
42	4	0	4	2	4	4	1	1
43	1	5	3	0	3	2	4	5
44	2	1	5	3	5	1	3	1
45	5	4	4	4	3	5	5	4
46	3	2	3	3	1	3	1	3

Workers Rating Scores in Ten skills.

Binomial Distribution ranging between 0 - 5.

47	1	2	4	2	1	1	0	0
48	3	1	3	0	3	1	2	0
49	1	1	2	1	4	3	0	3
50	3	4	4	1	1	2	3	1
51	4	5	1	1	4	2	2	3
52	3	1	3	2	3	2	1	3
53	1	2	1	3	4	4	2	2
54	3	4	1	2	3	4	1	1
55	2	4	3	4	4	4	3	1
56	2	5	2	3	2	1	1	2
57	3	2	4	2	1	5	5	4
58	2	4	1	2	5	2	4	2
59	3	5	2	5	2	4	2	2
60	2	1	3	1	4	0	0	4
61	2	1	0	3	3	3	0	4
62	1	3	3	4	4	2	2	4
63	4	4	4	2	2	3	4	2
64	5	3	3	2	4	3	1	0
65	2	4	4	4	2	3	3	5
66	5	0	1	3	5	4	5	1
67	4	1	1	1	1	3	4	0
68	3	4	2	4	4	3	1	5
69	2	4	4	2	2	4	5	2
70	4	2	3	5	3	2	4	5
71	5	4	4	3	4	3	4	2
72	3	2	2	1	1	4	0	5
73	5	1	2	4	5	0	5	2
74	1	2	4	2	1	1	1	1
75	3	1	4	5	5	2	2	2
76	2	1	2	1	4	3	3	4
77	3	4	1	1	1	4	3	5
78	2	2	3	3	1	2	4	0
79	2	1	1	1	1	3	3	2
80	3	3	5	4	2	3	4	4
81	2	4	4	1	2	4	1	3
82	4	3	2	0	3	3	4	3
83	4	5	2	4	0	1	2	4
84	1	1	4	2	4	0	2	1
85	2	1	1	2	1	3	2	2
86	5	1	1	1	1	5	5	2
87	2	4	5	0	3	5	1	2
88	2	0	2	2	3	4	1	5
89	1	3	4	3	3	1	4	5
90	1	2	3	2	0	2	2	3
91	1	5	4	1	2	4	4	4
92	1	2	2	2	1	2	4	0
93	1	3	1	3	3	2	2	0

94	4	1	2	1	0	5	4	2
95	4	4	2	4	2	0	3	5
96	4	1	2	5	1	4	2	0
97	4	2	2	2	3	0	3	4
98	4	3	3	1	0	2	0	2
99	1	0	4	0	4	1	3	0
100	4	3	1	3	5	4	0	1

Skill 9	Skill 10
1	5
4	4
4	5
0	1
4	4
1	4
2	2
3	0
0	1
3	3
4	1
3	4
3	3
2	3
1	3
4	1
2	4
3	4
1	5
1	3
2	2
2	3
4	4
4	3
2	0
4	1
4	0
0	1
0	1
5	4
5	2
3	3
3	4
0	2
3	1
5	2
1	5
2	2
0	4
4	2
1	1
3	5
3	0
4	4
4	2
2	3

3	3
3	2
2	0
1	4
5	3
4	2
1	4
3	5
2	1
3	2
3	5
2	3
3	4
1	5
2	4
5	4
0	4
1	2
1	3
5	0
2	0
4	4
1	3
0	1
1	0
1	1
2	3
0	4
3	1
1	1
5	2
4	1
1	1
4	5
2	5
1	2
2	3
4	2
2	1
4	2
2	2
1	5
4	2
3	1
4	2
1	1
1	2

3	0
3	2
3	3
1	3
3	0
1	2
2	1

Workers Task weight for Ten skills

Random number generation between (100, 25, 33, 50)

Workers	Skill 1	Skill 2	Skill 3	Skill 4	Skill 5	Skill 6	Skill 7	Skill 8
1	50.00	25.00	25.00	33.00	25.00	50.00	100.00	33.00
2	25.00	25.00	33.00	33.00	33.00	33.00	33.00	50.00
3	33.00	50.00	50.00	100.00	25.00	50.00	25.00	25.00
4	100.00	50.00	25.00	25.00	33.00	100.00	33.00	100.00
5	25.00	100.00	50.00	100.00	25.00	100.00	100.00	33.00
6	100.00	25.00	100.00	33.00	25.00	100.00	33.00	33.00
7	100.00	33.00	100.00	33.00	25.00	25.00	33.00	100.00
8	50.00	33.00	33.00	50.00	100.00	25.00	25.00	100.00
9	100.00	50.00	50.00	25.00	100.00	25.00	25.00	100.00
10	25.00	100.00	100.00	50.00	50.00	100.00	33.00	100.00
11	100.00	100.00	100.00	25.00	33.00	33.00	33.00	33.00
12	33.00	100.00	100.00	25.00	50.00	25.00	33.00	33.00
13	33.00	25.00	50.00	50.00	25.00	100.00	25.00	50.00
14	50.00	100.00	33.00	50.00	50.00	100.00	33.00	50.00
15	50.00	50.00	33.00	100.00	100.00	50.00	33.00	100.00
16	100.00	100.00	100.00	33.00	33.00	25.00	50.00	25.00
17	33.00	33.00	25.00	50.00	100.00	100.00	100.00	50.00
18	33.00	100.00	50.00	33.00	50.00	100.00	100.00	50.00
19	25.00	100.00	25.00	100.00	33.00	100.00	100.00	100.00
20	25.00	100.00	100.00	50.00	50.00	33.00	100.00	100.00
21	33.00	25.00	50.00	25.00	33.00	100.00	25.00	33.00
22	50.00	50.00	100.00	33.00	33.00	25.00	25.00	25.00
23	25.00	50.00	100.00	50.00	50.00	100.00	100.00	25.00
24	100.00	25.00	25.00	33.00	50.00	33.00	100.00	33.00
25	25.00	33.00	50.00	100.00	33.00	25.00	25.00	50.00
26	33.00	33.00	33.00	100.00	50.00	100.00	25.00	100.00
27	50.00	25.00	25.00	33.00	100.00	100.00	50.00	100.00
28	100.00	50.00	50.00	25.00	25.00	33.00	100.00	25.00
29	50.00	25.00	100.00	25.00	25.00	50.00	100.00	25.00
30	50.00	25.00	33.00	25.00	25.00	33.00	25.00	100.00
31	33.00	33.00	50.00	33.00	100.00	25.00	100.00	50.00
32	33.00	50.00	33.00	33.00	25.00	100.00	25.00	33.00
33	100.00	50.00	33.00	25.00	25.00	100.00	50.00	25.00
34	100.00	33.00	100.00	50.00	100.00	100.00	50.00	33.00
35	25.00	50.00	100.00	100.00	33.00	100.00	50.00	100.00
36	25.00	50.00	100.00	50.00	100.00	100.00	100.00	100.00
37	50.00	100.00	25.00	33.00	50.00	25.00	33.00	100.00
38	33.00	25.00	100.00	33.00	100.00	50.00	100.00	50.00
39	25.00	25.00	50.00	33.00	100.00	33.00	50.00	33.00
40	25.00	100.00	33.00	33.00	33.00	50.00	25.00	33.00
41	25.00	25.00	50.00	25.00	25.00	33.00	33.00	33.00
42	33.00	33.00	25.00	33.00	25.00	100.00	25.00	33.00
43	100.00	25.00	33.00	33.00	25.00	50.00	100.00	33.00
44	100.00	25.00	33.00	25.00	25.00	50.00	25.00	50.00
45	50.00	25.00	100.00	33.00	25.00	100.00	33.00	100.00
46	25.00	50.00	25.00	100.00	50.00	50.00	50.00	50.00

47	100.00	100.00	50.00	50.00	25.00	50.00	33.00	25.00
48	33.00	33.00	50.00	50.00	50.00	33.00	25.00	100.00
49	25.00	100.00	33.00	25.00	25.00	100.00	50.00	33.00
50	25.00	50.00	50.00	25.00	50.00	50.00	50.00	50.00
51	25.00	25.00	33.00	33.00	50.00	50.00	33.00	100.00
52	25.00	33.00	25.00	50.00	33.00	25.00	25.00	33.00
53	100.00	100.00	25.00	100.00	25.00	25.00	25.00	33.00
54	50.00	33.00	100.00	25.00	25.00	50.00	100.00	50.00
55	50.00	33.00	100.00	50.00	25.00	25.00	33.00	100.00
56	25.00	100.00	33.00	100.00	25.00	50.00	33.00	33.00
57	100.00	25.00	50.00	25.00	33.00	100.00	33.00	33.00
58	25.00	25.00	50.00	25.00	25.00	50.00	100.00	50.00
59	100.00	33.00	100.00	100.00	33.00	100.00	33.00	50.00
60	33.00	100.00	33.00	33.00	33.00	25.00	25.00	33.00
61	50.00	33.00	100.00	100.00	50.00	50.00	50.00	50.00
62	25.00	50.00	50.00	33.00	100.00	100.00	100.00	50.00
63	33.00	50.00	50.00	33.00	25.00	50.00	25.00	100.00
64	100.00	50.00	50.00	50.00	25.00	50.00	33.00	100.00
65	33.00	100.00	25.00	100.00	50.00	33.00	33.00	33.00
66	25.00	25.00	33.00	50.00	100.00	100.00	100.00	33.00
67	50.00	25.00	25.00	33.00	50.00	50.00	100.00	100.00
68	50.00	50.00	100.00	33.00	100.00	33.00	33.00	25.00
69	50.00	33.00	33.00	50.00	25.00	100.00	100.00	25.00
70	100.00	33.00	50.00	25.00	100.00	50.00	50.00	100.00
71	100.00	50.00	25.00	25.00	50.00	33.00	25.00	25.00
72	25.00	33.00	33.00	100.00	100.00	25.00	50.00	25.00
73	33.00	33.00	33.00	25.00	50.00	100.00	25.00	25.00
74	33.00	50.00	25.00	33.00	50.00	33.00	33.00	25.00
75	33.00	100.00	50.00	100.00	50.00	25.00	100.00	100.00
76	50.00	100.00	50.00	33.00	33.00	25.00	25.00	100.00
77	25.00	25.00	25.00	50.00	25.00	25.00	50.00	33.00
78	33.00	25.00	100.00	50.00	33.00	50.00	50.00	100.00
79	25.00	33.00	100.00	50.00	50.00	50.00	100.00	33.00
80	50.00	33.00	25.00	50.00	50.00	50.00	25.00	100.00
81	100.00	100.00	25.00	33.00	25.00	33.00	33.00	33.00
82	25.00	25.00	50.00	50.00	100.00	100.00	25.00	100.00
83	33.00	100.00	25.00	50.00	25.00	33.00	100.00	33.00
84	33.00	33.00	100.00	25.00	25.00	25.00	33.00	50.00
85	100.00	33.00	100.00	25.00	100.00	25.00	50.00	100.00
86	25.00	25.00	100.00	100.00	33.00	33.00	50.00	100.00
87	50.00	50.00	33.00	25.00	50.00	100.00	33.00	25.00
88	33.00	50.00	25.00	50.00	25.00	100.00	33.00	25.00
89	33.00	100.00	33.00	100.00	100.00	33.00	33.00	50.00
90	25.00	25.00	25.00	100.00	100.00	50.00	50.00	33.00
91	100.00	25.00	33.00	33.00	50.00	33.00	50.00	33.00
92	25.00	25.00	100.00	100.00	33.00	100.00	50.00	25.00
93	33.00	25.00	100.00	50.00	100.00	50.00	100.00	100.00

94	100.00	33.00	100.00	25.00	50.00	100.00	100.00	33.00
95	25.00	100.00	100.00	50.00	33.00	50.00	100.00	25.00
96	25.00	100.00	100.00	100.00	33.00	50.00	25.00	25.00
97	25.00	33.00	100.00	33.00	25.00	100.00	100.00	50.00
98	100.00	33.00	50.00	25.00	33.00	25.00	25.00	25.00
99	50.00	50.00	50.00	33.00	25.00	100.00	100.00	100.00
100	100.00	50.00	100.00	25.00	25.00	100.00	50.00	33.00

Skill 9	Skill 10
33.00	25.00
33.00	50.00
33.00	25.00
100.00	100.00
100.00	33.00
100.00	33.00
33.00	100.00
100.00	33.00
100.00	25.00
100.00	25.00
100.00	25.00
100.00	100.00
50.00	100.00
25.00	25.00
100.00	50.00
50.00	33.00
33.00	25.00
25.00	25.00
33.00	100.00
33.00	25.00
100.00	50.00
100.00	100.00
33.00	33.00
33.00	33.00
33.00	25.00
33.00	33.00
25.00	25.00
25.00	100.00
100.00	100.00
100.00	100.00
25.00	33.00
33.00	50.00
50.00	100.00
50.00	100.00
25.00	50.00
33.00	50.00
100.00	33.00
100.00	33.00
33.00	100.00
33.00	50.00
33.00	33.00
33.00	33.00
100.00	100.00
33.00	100.00
25.00	25.00
50.00	50.00

50.00	25.00
50.00	25.00
25.00	25.00
50.00	33.00
50.00	50.00
100.00	100.00
100.00	25.00
33.00	33.00
50.00	25.00
100.00	100.00
33.00	33.00
33.00	33.00
50.00	33.00
100.00	33.00
100.00	25.00
25.00	25.00
50.00	50.00
25.00	33.00
100.00	100.00
33.00	100.00
100.00	50.00
100.00	100.00
50.00	100.00
25.00	33.00
50.00	100.00
50.00	100.00
100.00	33.00
25.00	50.00
33.00	33.00
50.00	100.00
33.00	50.00
50.00	33.00
50.00	33.00
33.00	25.00
50.00	25.00
33.00	33.00
25.00	100.00
33.00	50.00
25.00	100.00
33.00	100.00
25.00	100.00
25.00	100.00
33.00	33.00
100.00	100.00
100.00	25.00
25.00	33.00
50.00	50.00

100.00	50.00
25.00	100.00
50.00	100.00
25.00	25.00
50.00	50.00
50.00	100.00
50.00	100.00

Employers Files
Commitment score using uniform distribution.

Rating Score using binomial distribution.

Employer	commitment	Rating
1	0.910855434	1
2	0.996887112	3
3	0.3234962	2
4	0.38001648	4
5	0.229590747	4
6	0.738151189	0
7	0.196539201	0
8	0.902096622	2
9	0.945066683	3
10	0.631763665	4
11	0.741477706	3
12	0.552903836	1
13	0.24713889	4
14	0.627979369	4
15	0.047456282	2
16	0.745109409	0
17	0.357463301	2
18	0.18344676	3
19	0.87981811	4
20	0.277169103	3
21	0.09787286	2
22	0.680837428	3
23	0.046113468	0
24	0.359447005	3
25	0.437452315	1
26	0.912747581	1
27	0.569444868	3
28	0.220252083	4
29	0.299081393	3
30	0.786492508	2
31	0.198980682	3
32	0.698294015	3
33	0.407574694	1
34	0.905758843	5
35	0.290292062	2
36	0.727500229	4
37	0.459395123	0
38	0.185430464	5
39	0.585467086	5
40	0.121097446	4
41	0.396343883	2
42	0.155949583	2
43	0.925290689	4
44	0.560289315	4
45	0.063966796	3
46	0.113315226	4

47	0.424176763	3
48	0.194586016	5
49	0.035798212	3
50	0.771172216	0
51	0.139194922	3
52	0.759544664	2
53	0.884914701	0
54	0.570696127	4
55	0.764519181	2
56	0.618366039	3
57	0.684133427	3
58	0.305612354	4
59	0.382549516	3
60	0.633716849	0
61	0.01596118	4
62	0.217291787	1
63	0.802911466	3
64	0.275673696	2
65	0.273415326	3
66	0.217047639	3
67	0.520554216	1
68	0.87243263	4
69	0.533158361	4
70	0.391735588	1
71	0.824304941	0
72	0.409436323	0
73	0.782250435	0
74	0.448591571	4
75	0.086977752	1
76	0.784417249	3
77	0.779595325	1
78	0.255958739	1
79	0.768578143	2
80	0.922696615	4
81	0.333353679	1
82	0.135898923	1
83	0.163151952	3
84	0.661030915	3
85	0.275765252	5
86	0.06491287	4
87	0.421826838	1
88	0.804620502	1
89	0.916348766	3
90	0.439222388	4
91	0.881710257	0
92	0.264473403	3
93	0.254768517	4

94	0.560838649	0
95	0.23300882	3
96	0.681173132	3
97	0.735190893	3
98	0.500320444	2
99	0.555558947	1
100	0.440961943	4

Tasks	Type	Employer	Reward	Deadline
1	3	15	392	66
2	6	50	924	88
3	10	9	159	30
4	6	74	527	107
5	2	100	539	15
6	7	70	513	48
7	2	69	253	97
8	2	56	854	94
9	10	98	203	40
10	3	30	669	100
11	10	20	789	52
12	9	73	890	54
13	5	92	742	94
14	6	91	779	57
15	3	47	450	79
16	4	17	879	73
17	8	92	218	20
18	4	69	477	14
19	1	60	584	72
20	8	47	700	95
21	7	40	411	85
22	2	24	826	39
23	1	44	493	36
24	9	20	501	83
25	7	58	320	58
26	8	15	943	29
27	6	98	435	82
28	2	47	312	15
29	1	8	436	14
30	8	18	617	113
31	10	55	248	12
32	4	59	869	90
33	4	39	774	59
34	8	19	343	24
35	3	52	639	113
36	2	29	703	56
37	4	6	804	43
38	7	30	807	112
39	8	34	155	106
40	6	87	239	26
41	7	84	425	105
42	7	40	899	26
43	10	54	819	46
44	8	34	296	33
45	1	74	760	30
46	7	26	557	24

Task Lists (5000 tasks)
contain task id,
type : random number generation.
Employer: random number generation.
Rewards: uniform distribution.
Deadline: uniform distribution.

47	9	9	994	17
48	1	93	634	39
49	4	79	526	19
50	1	95	690	78
51	2	83	379	108
52	6	55	400	64
53	6	38	350	91
54	6	9	553	66
55	2	6	776	40
56	10	98	157	53
57	3	55	595	45
58	9	66	973	75
59	6	28	344	52
60	3	72	551	48
61	7	82	799	77
62	3	22	180	51
63	10	33	676	87
64	1	98	709	43
65	1	36	762	113
66	4	85	863	102
67	3	29	623	116
68	3	66	758	93
69	3	94	317	55
70	4	38	750	48
71	1	65	436	52
72	1	57	519	63
73	7	16	383	101
74	8	82	616	76
75	2	16	359	16
76	9	91	459	20
77	4	36	632	28
78	3	15	343	100
79	8	86	254	69
80	10	63	370	111
81	5	94	602	12
82	9	70	769	112
83	1	26	512	40
84	5	8	980	38
85	9	22	466	118
86	2	77	648	27
87	8	80	660	24
88	1	4	594	70
89	7	11	627	42
90	8	34	533	81
91	9	83	163	67
92	1	74	453	56
93	5	1	581	89

94	3	54	313	64
95	6	66	363	62
96	5	93	712	44
97	8	11	502	13
98	6	75	552	115
99	8	37	165	27
100	9	70	736	57
101	4	69	862	113
102	8	45	299	21
103	8	4	582	15
104	9	1	888	106
105	2	14	152	21
106	6	22	657	84
107	8	84	510	55
108	3	2	845	94
109	10	24	476	99
110	10	83	845	62
111	1	79	569	47
112	1	23	935	110
113	7	49	661	116
114	1	100	282	15
115	6	13	787	94
116	5	7	691	33
117	7	47	273	38
118	4	42	233	112
119	6	24	444	109
120	8	90	277	17
121	10	52	981	97
122	5	78	686	38
123	10	89	316	71
124	3	89	665	20
125	9	78	604	35
126	8	92	464	117
127	9	16	330	65
128	6	77	571	64
129	6	54	276	97
130	6	92	480	65
131	1	29	520	87
132	2	67	492	28
133	9	86	532	10
134	8	17	290	22
135	7	38	855	48
136	1	92	783	21
137	9	28	789	80
138	8	40	611	89
139	2	1	166	66
140	7	84	733	21

141	7	63	968	88
142	1	40	248	63
143	3	33	796	67
144	9	21	868	44
145	1	64	200	11
146	9	27	952	114
147	9	95	504	82
148	8	90	994	79
149	4	15	691	115
150	4	93	155	115
151	4	91	347	63
152	3	30	461	69
153	10	37	658	42
154	9	91	398	38
155	2	92	720	38
156	10	11	797	43
157	5	100	546	66
158	10	43	956	110
159	10	86	671	91
160	2	10	220	34
161	7	30	497	47
162	1	23	469	55
163	1	72	733	14
164	2	51	547	101
165	1	79	528	37
166	6	51	734	88
167	5	59	582	72
168	3	42	727	73
169	7	39	935	56
170	1	40	520	14
171	10	11	267	94
172	4	79	914	25
173	9	33	792	14
174	8	57	163	83
175	10	84	564	112
176	6	28	647	100
177	10	90	549	41
178	8	7	701	65
179	2	61	158	73
180	7	12	619	118
181	4	75	651	117
182	5	64	438	92
183	4	88	508	69
184	3	74	940	33
185	8	25	488	43
186	5	57	785	51
187	2	61	786	83

188	3	6	276	27
189	5	26	647	64
190	9	45	439	100
191	9	73	549	112
192	3	63	680	69
193	6	74	862	98
194	5	29	793	66
195	5	77	197	113
196	7	86	386	59
197	10	21	908	102
198	3	89	817	78
199	7	45	715	93
200	3	93	633	50
201	6	32	213	62
202	7	62	874	99
203	4	45	398	103
204	3	8	296	18
205	9	85	890	62
206	3	49	529	34
207	4	40	271	33
208	4	32	988	99
209	6	64	621	112
210	3	60	452	67
211	4	14	589	52
212	4	43	657	104
213	4	78	490	92
214	10	56	284	36
215	8	94	989	67
216	1	74	896	119
217	3	69	587	49
218	5	26	866	38
219	1	46	273	22
220	6	72	889	113
221	1	3	850	77
222	10	13	988	77
223	6	90	749	74
224	2	38	663	25
225	5	56	350	49
226	3	7	475	71
227	8	44	350	53
228	2	38	158	32
229	5	82	588	62
230	3	87	603	118
231	9	13	422	94
232	3	29	653	34
233	10	55	930	104
234	5	15	920	23

235	2	52	489	92
236	3	69	151	37
237	8	78	210	33
238	8	15	459	93
239	2	25	236	94
240	2	95	297	42
241	3	36	312	24
242	7	7	732	85
243	6	68	433	38
244	2	13	313	11
245	4	32	252	38
246	1	49	619	96
247	8	46	918	82
248	9	72	600	59
249	3	26	353	12
250	5	28	959	59
251	9	22	985	32
252	8	66	841	17
253	5	64	810	103
254	8	56	456	75
255	2	66	274	11
256	1	41	818	51
257	3	79	622	32
258	7	90	693	30
259	6	20	427	90
260	9	39	676	57
261	1	72	743	55
262	9	15	185	46
263	8	19	495	33
264	3	10	329	67
265	10	74	424	80
266	10	55	351	118
267	4	5	817	87
268	2	5	713	27
269	9	36	809	11
270	6	78	795	43
271	1	72	699	95
272	2	71	618	115
273	3	19	500	22
274	7	27	817	76
275	10	50	370	80
276	5	99	308	51
277	10	29	541	119
278	9	94	698	23
279	2	77	197	114
280	1	37	564	97
281	6	7	165	108

282	3	89	987	53
283	3	44	818	102
284	7	17	697	22
285	5	61	738	88
286	6	44	974	64
287	1	35	599	42
288	6	23	336	36
289	6	45	925	73
290	6	95	525	114
291	3	45	854	68
292	5	67	866	79
293	5	74	553	54
294	1	57	528	79
295	1	74	795	64
296	9	8	291	44
297	7	10	417	114
298	8	17	166	43
299	3	63	302	36
300	1	49	331	61
301	6	7	204	25
302	3	43	705	41
303	10	3	969	34
304	1	76	836	24
305	5	61	496	22
306	5	70	436	95
307	6	69	577	38
308	2	41	326	11
309	9	76	817	119
310	3	9	651	54
311	7	75	346	32
312	10	54	688	108
313	7	91	538	77
314	6	19	472	61
315	3	40	568	29
316	6	50	840	100
317	3	99	551	37
318	4	2	446	58
319	2	66	568	41
320	6	42	343	54
321	5	28	635	89
322	8	61	349	45
323	5	87	705	81
324	2	15	320	59
325	6	76	608	91
326	10	25	927	34
327	4	51	251	97
328	1	92	557	42

329	6	85	214	111
330	5	93	962	107
331	2	86	761	29
332	1	72	569	118
333	4	16	716	54
334	6	95	232	67
335	6	90	276	104
336	9	37	922	99
337	10	94	594	56
338	2	39	620	79
339	6	9	725	21
340	1	77	714	26
341	5	56	248	100
342	6	98	635	31
343	6	97	258	117
344	6	100	259	41
345	3	63	416	100
346	7	43	240	112
347	5	13	523	84
348	2	76	615	19
349	10	3	728	11
350	9	22	205	116
351	9	12	648	80
352	8	40	509	31
353	8	45	963	114
354	6	81	998	83
355	2	89	251	36
356	7	71	992	86
357	4	37	357	39
358	3	83	702	106
359	6	2	267	18
360	3	65	537	100
361	4	5	953	91
362	2	64	391	60
363	8	30	955	76
364	7	25	255	85
365	10	42	975	25
366	4	61	297	38
367	9	2	320	34
368	5	44	240	79
369	8	40	372	47
370	9	10	301	62
371	10	90	832	63
372	5	43	785	29
373	7	36	262	73
374	5	43	835	59
375	3	51	442	81

376	8	39	506	26
377	8	33	579	36
378	7	37	715	114
379	6	88	468	17
380	5	57	880	114
381	4	84	603	24
382	7	9	760	12
383	4	78	372	94
384	6	73	738	72
385	8	48	702	73
386	6	3	353	38
387	3	96	955	90
388	1	86	402	46
389	9	48	653	47
390	5	53	900	94
391	9	57	463	20
392	1	9	728	111
393	6	50	503	77
394	6	22	804	79
395	10	72	842	109
396	10	58	578	87
397	10	82	172	34
398	8	45	998	26
399	3	92	173	77
400	5	41	940	109
401	10	90	388	33
402	5	23	562	69
403	10	99	408	115
404	9	93	514	80
405	8	57	291	106
406	7	9	609	69
407	4	57	482	46
408	8	35	346	46
409	7	33	239	72
410	9	51	226	76
411	1	97	532	50
412	2	100	610	78
413	9	93	917	33
414	7	50	948	114
415	5	43	896	107
416	2	78	327	28
417	7	45	478	117
418	2	8	1000	39
419	6	73	378	102
420	7	70	983	62
421	10	71	360	103
422	6	89	895	59

423	8	79	381	50
424	2	80	166	37
425	2	52	267	95
426	10	20	754	115
427	1	95	723	89
428	9	7	846	44
429	7	3	601	15
430	4	49	759	36
431	2	87	523	25
432	3	87	311	75
433	2	19	486	55
434	1	50	388	115
435	2	45	162	52
436	2	61	543	34
437	8	74	328	107
438	1	43	195	37
439	7	95	729	114
440	9	22	635	46
441	1	57	439	72
442	7	42	951	107
443	8	52	340	105
444	7	13	292	61
445	5	70	421	58
446	1	62	479	64
447	6	15	748	118
448	2	59	510	40
449	10	39	200	66
450	7	88	873	97
451	1	78	673	70
452	8	48	732	54
453	6	61	599	33
454	1	14	936	111
455	5	82	334	31
456	8	68	677	25
457	2	71	932	65
458	1	40	947	104
459	1	5	398	119
460	8	14	176	115
461	9	17	866	69
462	3	44	580	117
463	2	65	284	85
464	4	60	983	43
465	2	30	640	46
466	3	75	915	94
467	7	39	296	107
468	4	98	474	96
469	3	29	923	41

470	2	3	707	94
471	2	98	605	32
472	9	40	768	42
473	6	29	657	13
474	5	90	422	84
475	4	74	713	27
476	7	17	227	119
477	3	30	973	66
478	3	86	793	101
479	6	27	922	101
480	4	82	210	103
481	7	41	760	78
482	3	87	911	99
483	9	30	943	25
484	2	86	860	70
485	6	100	499	53
486	6	14	450	72
487	6	70	555	85
488	9	92	925	73
489	5	46	766	20
490	6	47	838	84
491	10	71	858	113
492	9	3	904	35
493	3	25	980	29
494	8	48	638	36
495	8	70	835	109
496	10	69	724	44
497	5	9	959	56
498	9	11	332	111
499	7	45	509	107
500	9	14	578	48
501	7	72	633	42
502	10	36	734	84
503	3	36	305	73
504	5	51	380	14
505	5	42	504	94
506	7	72	870	96
507	5	4	266	18
508	8	3	531	108
509	5	11	738	80
510	7	14	805	67
511	9	92	636	116
512	8	62	291	61
513	5	37	633	92
514	7	17	802	36
515	9	67	632	27
516	10	73	693	64

517	9	4	595	105
518	3	6	590	28
519	1	39	345	86
520	7	7	664	30
521	1	91	720	106
522	9	56	336	84
523	10	60	891	17
524	8	24	926	108
525	2	98	662	112
526	4	80	286	41
527	1	81	930	35
528	4	30	942	70
529	9	50	834	25
530	5	11	679	38
531	7	64	844	17
532	3	100	935	16
533	3	36	274	84
534	3	41	592	116
535	2	22	789	46
536	7	18	192	14
537	1	6	572	79
538	3	59	730	109
539	8	16	511	96
540	3	32	284	117
541	7	9	991	75
542	1	71	307	69
543	7	93	726	89
544	7	76	154	53
545	1	97	975	50
546	4	19	687	106
547	1	61	381	58
548	10	28	221	119
549	1	59	283	82
550	3	36	837	113
551	5	29	630	107
552	7	40	738	41
553	8	79	179	35
554	2	92	651	10
555	4	53	483	26
556	5	75	505	102
557	1	32	190	47
558	5	52	606	98
559	3	70	940	108
560	5	42	732	75
561	3	13	638	118
562	4	41	197	70
563	7	2	839	55

564	1	32	592	38
565	7	58	368	88
566	7	37	625	16
567	2	82	393	64
568	2	25	318	54
569	1	51	736	65
570	3	26	587	81
571	1	32	753	86
572	5	93	929	36
573	8	16	162	83
574	2	67	817	38
575	3	92	250	35
576	1	41	290	70
577	8	89	527	10
578	5	43	307	56
579	7	11	327	21
580	10	34	946	113
581	1	63	364	45
582	3	54	696	63
583	2	95	551	84
584	6	48	527	98
585	8	5	360	69
586	9	63	894	90
587	1	5	810	37
588	9	48	565	25
589	3	12	399	61
590	5	73	153	104
591	10	100	307	108
592	9	47	280	85
593	10	97	878	64
594	3	74	978	59
595	6	53	378	30
596	1	58	961	54
597	7	68	297	32
598	6	48	353	91
599	10	47	770	109
600	8	66	260	111
601	2	29	156	42
602	1	44	628	67
603	4	35	644	65
604	8	59	324	100
605	7	85	641	84
606	8	22	481	61
607	7	96	736	49
608	4	6	578	64
609	7	77	712	32
610	8	81	707	66

611	3	29	952	70
612	2	74	786	53
613	10	45	497	111
614	5	79	550	98
615	8	65	873	42
616	5	78	502	49
617	7	95	913	35
618	5	82	627	99
619	2	22	849	87
620	10	57	997	12
621	1	1	981	107
622	9	42	910	115
623	5	99	944	12
624	7	73	860	90
625	8	83	494	33
626	10	7	467	69
627	4	16	541	110
628	6	40	715	26
629	4	57	849	65
630	10	30	170	36
631	4	74	944	28
632	5	95	606	12
633	10	83	672	113
634	7	46	641	116
635	1	97	181	32
636	6	67	391	13
637	9	49	274	116
638	4	79	869	49
639	9	39	230	98
640	5	51	288	60
641	1	57	536	68
642	10	77	831	82
643	6	37	532	59
644	7	74	872	72
645	8	48	902	18
646	7	91	809	45
647	1	34	581	89
648	1	87	524	34
649	4	40	771	94
650	7	28	715	24
651	1	18	767	26
652	3	5	655	104
653	7	37	489	88
654	8	6	638	118
655	7	95	435	101
656	7	77	586	39
657	1	74	257	90

658	2	94	810	41
659	6	60	203	56
660	4	25	893	41
661	7	100	822	63
662	8	46	501	29
663	1	91	708	82
664	9	14	487	98
665	3	76	859	57
666	1	30	867	43
667	4	41	557	16
668	4	81	316	44
669	3	86	867	58
670	1	71	181	68
671	5	38	952	11
672	3	59	627	45
673	7	25	837	75
674	10	100	927	41
675	10	16	239	95
676	2	44	452	101
677	10	4	652	112
678	3	80	208	105
679	4	15	278	92
680	4	37	691	42
681	8	65	698	80
682	10	91	774	62
683	9	18	231	63
684	6	14	756	29
685	4	18	580	77
686	1	92	524	36
687	10	69	532	84
688	6	3	360	67
689	7	96	223	85
690	9	39	354	18
691	2	48	283	98
692	4	54	647	92
693	8	59	418	71
694	9	13	945	73
695	6	82	629	69
696	1	3	972	11
697	5	88	273	92
698	5	44	298	93
699	7	46	758	77
700	7	43	798	41
701	1	3	532	64
702	8	49	890	118
703	7	29	789	33
704	2	5	493	77

705	1	28	886	13
706	5	62	842	91
707	3	89	742	93
708	5	22	283	88
709	1	23	232	26
710	8	67	911	83
711	4	79	490	112
712	3	23	492	29
713	1	44	744	49
714	5	98	871	67
715	10	86	949	108
716	1	63	633	83
717	4	95	920	92
718	6	18	929	57
719	9	51	359	85
720	5	69	473	48
721	8	40	579	89
722	3	93	450	31
723	9	23	346	59
724	8	7	646	81
725	5	33	402	83
726	2	79	276	105
727	2	38	340	96
728	1	5	817	15
729	3	99	305	114
730	5	89	925	63
731	9	10	967	85
732	1	92	652	87
733	10	64	489	19
734	10	6	398	25
735	5	75	951	58
736	1	19	721	69
737	1	71	938	67
738	9	90	877	39
739	3	36	503	111
740	8	66	940	107
741	10	72	683	113
742	2	66	856	49
743	1	27	812	75
744	10	9	503	18
745	6	78	240	64
746	10	55	675	105
747	1	78	729	103
748	4	85	281	31
749	8	88	886	103
750	3	62	543	84
751	6	97	660	94

752	4	28	607	39
753	10	21	796	67
754	10	38	346	103
755	5	7	304	12
756	7	74	518	36
757	3	90	392	39
758	4	60	698	109
759	6	30	248	21
760	10	17	720	118
761	8	6	777	20
762	7	45	945	67
763	7	6	863	54
764	8	50	207	66
765	10	9	817	86
766	1	98	382	97
767	5	22	867	64
768	10	14	687	113
769	9	47	530	93
770	3	15	932	11
771	9	30	881	18
772	9	71	261	31
773	6	61	204	14
774	3	71	560	118
775	9	4	758	106
776	4	59	691	109
777	6	3	349	76
778	4	43	306	11
779	7	25	269	21
780	3	54	361	90
781	3	11	407	26
782	8	64	633	64
783	10	69	400	77
784	8	21	681	61
785	10	23	477	99
786	9	2	596	24
787	8	20	412	61
788	9	5	560	114
789	6	41	566	51
790	8	79	190	111
791	7	27	677	10
792	4	26	767	67
793	10	78	795	40
794	8	80	704	100
795	2	10	470	25
796	4	40	453	94
797	7	96	452	44
798	2	67	848	31

799	7	65	736	113
800	1	84	856	46
801	5	99	182	61
802	7	25	512	12
803	2	4	178	43
804	4	93	998	79
805	7	20	951	89
806	1	83	563	119
807	5	90	616	83
808	9	67	411	30
809	1	65	669	46
810	2	50	338	51
811	4	57	618	84
812	9	95	497	61
813	5	8	825	56
814	4	26	792	99
815	6	99	162	86
816	8	81	854	94
817	6	86	209	46
818	5	76	503	15
819	1	87	420	120
820	5	32	881	47
821	8	3	811	66
822	5	26	157	73
823	10	11	174	97
824	4	40	431	14
825	9	47	446	79
826	8	93	998	37
827	2	62	155	72
828	5	21	498	90
829	10	6	587	65
830	7	78	976	114
831	9	7	678	27
832	3	42	229	105
833	5	99	546	56
834	1	77	419	39
835	5	3	444	115
836	5	62	617	48
837	2	7	538	94
838	6	40	586	37
839	8	14	590	82
840	10	89	957	25
841	8	73	928	43
842	1	94	943	37
843	6	31	615	108
844	3	82	978	85
845	1	7	638	118

846	10	78	637	115
847	2	58	453	39
848	8	50	674	84
849	6	33	711	103
850	6	86	672	67
851	2	16	427	25
852	8	75	552	26
853	4	11	719	91
854	8	21	331	112
855	5	90	604	69
856	5	23	277	73
857	6	15	801	45
858	9	5	529	72
859	5	6	633	81
860	10	42	961	95
861	9	66	425	45
862	9	25	823	21
863	1	93	733	40
864	3	76	601	49
865	10	99	708	55
866	2	70	212	76
867	3	60	673	88
868	7	60	647	81
869	4	85	539	67
870	7	7	373	120
871	8	24	808	43
872	4	44	463	77
873	7	87	990	31
874	3	56	197	105
875	9	87	298	47
876	10	52	204	107
877	8	44	600	82
878	9	30	666	84
879	2	50	846	49
880	2	9	776	45
881	8	8	284	31
882	3	33	461	50
883	3	98	873	84
884	2	30	162	38
885	10	93	442	83
886	2	26	472	58
887	3	27	355	19
888	4	90	258	99
889	8	8	515	69
890	2	69	997	59
891	3	79	413	77
892	8	87	577	61

893	4	90	574	52
894	8	65	473	21
895	2	53	861	35
896	2	73	601	43
897	5	3	736	38
898	6	79	746	100
899	1	80	205	62
900	6	56	733	10
901	6	33	534	102
902	7	15	547	111
903	6	59	466	104
904	8	89	701	24
905	3	56	849	90
906	8	83	876	47
907	5	31	899	107
908	10	50	644	84
909	1	57	325	100
910	8	26	550	21
911	6	89	838	96
912	2	93	339	67
913	1	24	764	32
914	7	36	528	44
915	2	87	845	83
916	5	87	961	87
917	10	87	690	55
918	2	15	363	117
919	4	75	283	19
920	8	19	565	18
921	5	59	924	65
922	3	52	305	61
923	1	71	901	116
924	5	18	444	64
925	8	74	503	99
926	2	72	888	96
927	4	58	943	89
928	2	100	801	99
929	5	20	831	100
930	4	47	782	75
931	4	75	423	61
932	6	77	333	13
933	1	98	660	22
934	9	9	906	20
935	9	77	380	48
936	7	100	283	103
937	5	55	634	79
938	10	36	260	89
939	6	87	595	85

940	4	5	326	103
941	8	25	835	50
942	1	23	839	97
943	7	27	695	27
944	8	28	475	90
945	8	9	717	91
946	4	37	282	118
947	9	93	627	61
948	8	25	614	85
949	10	13	178	79
950	7	40	667	101
951	7	49	316	114
952	8	39	781	52
953	1	99	947	62
954	3	6	206	23
955	7	9	445	67
956	2	9	625	90
957	10	72	236	40
958	10	21	503	55
959	3	90	780	37
960	9	43	310	85
961	1	17	476	119
962	1	85	380	65
963	6	16	509	29
964	2	76	420	64
965	7	37	763	39
966	2	25	424	118
967	8	24	439	72
968	10	6	609	75
969	9	42	606	84
970	4	12	327	112
971	6	24	459	31
972	7	64	497	47
973	2	74	308	90
974	8	41	407	88
975	1	76	602	101
976	8	69	841	16
977	4	7	544	88
978	4	88	436	113
979	10	49	845	88
980	2	5	852	71
981	6	70	732	67
982	2	14	372	84
983	8	61	300	105
984	7	17	724	53
985	2	60	487	91
986	6	7	357	96

987	10	10	292	47
988	2	79	997	30
989	10	76	355	116
990	10	42	334	81
991	8	53	943	16
992	9	56	176	49
993	1	40	990	86
994	9	62	903	14
995	6	17	838	51
996	3	1	929	47
997	2	30	398	92
998	2	5	655	64
999	3	82	450	23
1000	8	43	768	105
1001	6	11	308	84
1002	6	60	960	99
1003	1	64	208	22
1004	3	34	513	36
1005	6	26	882	106
1006	9	1	707	33
1007	4	100	666	118
1008	9	73	477	41
1009	9	98	778	16
1010	1	60	453	76
1011	7	50	773	44
1012	8	92	667	101
1013	6	82	483	96
1014	5	45	194	76
1015	2	89	854	101
1016	3	8	454	53
1017	7	28	429	77
1018	2	17	865	94
1019	1	60	332	96
1020	8	97	399	14
1021	4	29	303	78
1022	2	42	570	70
1023	8	66	224	34
1024	1	52	838	112
1025	9	7	421	105
1026	9	79	764	77
1027	6	15	430	22
1028	3	3	761	81
1029	6	11	524	71
1030	8	73	625	109
1031	7	47	991	51
1032	4	62	940	96
1033	2	58	266	59

1034	3	90	792	24
1035	1	24	757	58
1036	8	12	399	99
1037	10	18	150	42
1038	8	100	668	17
1039	6	80	914	71
1040	9	16	368	102
1041	10	100	386	91
1042	6	85	247	47
1043	2	71	880	60
1044	5	89	566	29
1045	5	6	674	86
1046	6	53	873	22
1047	7	81	816	76
1048	7	47	269	50
1049	5	84	941	59
1050	6	76	588	39
1051	2	31	801	65
1052	6	1	887	78
1053	1	49	927	64
1054	7	94	578	74
1055	2	59	735	72
1056	7	19	422	27
1057	5	81	352	64
1058	3	23	530	67
1059	2	91	351	90
1060	9	83	330	83
1061	1	66	962	31
1062	4	23	239	111
1063	3	31	860	62
1064	5	48	692	27
1065	6	34	291	106
1066	5	90	606	68
1067	3	38	813	61
1068	10	15	840	60
1069	9	33	516	11
1070	6	72	944	59
1071	2	97	443	38
1072	5	66	884	97
1073	10	55	165	92
1074	6	61	185	55
1075	10	54	278	80
1076	6	79	872	54
1077	7	24	551	11
1078	4	7	628	21
1079	3	94	643	88
1080	1	66	706	113

1081	8	62	623	78
1082	9	78	537	53
1083	2	71	580	55
1084	3	57	928	55
1085	4	33	234	51
1086	10	20	602	81
1087	8	64	258	86
1088	7	12	694	61
1089	8	92	797	74
1090	7	89	366	116
1091	1	70	313	103
1092	2	95	715	97
1093	2	95	389	88
1094	4	13	280	81
1095	10	10	476	50
1096	9	99	681	57
1097	6	12	323	101
1098	8	44	790	73
1099	5	75	281	91
1100	4	3	638	92
1101	5	63	689	73
1102	2	38	517	114
1103	5	52	820	75
1104	4	33	934	108
1105	9	54	198	24
1106	10	84	234	116
1107	6	4	977	76
1108	5	35	869	65
1109	4	60	771	17
1110	5	49	253	107
1111	1	73	716	64
1112	4	43	842	52
1113	2	27	759	69
1114	8	66	475	28
1115	3	20	596	88
1116	5	94	507	17
1117	4	39	187	75
1118	10	15	911	12
1119	6	72	745	10
1120	4	56	258	110
1121	2	95	205	31
1122	5	19	892	19
1123	9	4	372	49
1124	6	22	258	75
1125	9	31	590	87
1126	2	100	704	75
1127	7	72	177	92

1128	3	99	953	88
1129	2	76	439	38
1130	2	53	481	91
1131	1	73	260	23
1132	8	14	957	54
1133	3	9	788	19
1134	10	19	946	70
1135	5	12	983	105
1136	9	68	520	70
1137	6	80	698	112
1138	4	100	814	19
1139	7	13	232	53
1140	7	10	507	77
1141	8	72	847	87
1142	7	53	570	28
1143	8	89	482	14
1144	4	67	843	79
1145	2	96	215	81
1146	2	71	614	65
1147	8	50	420	108
1148	7	53	270	51
1149	3	35	760	119
1150	8	74	953	52
1151	9	37	670	84
1152	9	35	509	33
1153	9	21	556	53
1154	4	32	628	108
1155	7	8	931	58
1156	9	62	516	94
1157	8	70	506	38
1158	6	91	647	85
1159	10	60	223	24
1160	6	31	942	94
1161	6	71	845	22
1162	10	45	969	53
1163	10	91	494	89
1164	9	24	226	42
1165	8	90	638	27
1166	7	39	415	81
1167	5	96	640	22
1168	6	95	729	97
1169	6	54	313	36
1170	9	88	246	91
1171	3	29	741	45
1172	9	40	400	11
1173	4	66	851	41
1174	9	22	725	19

1175	3	77	162	37
1176	9	50	262	41
1177	10	18	541	57
1178	7	36	769	35
1179	7	72	821	88
1180	7	16	678	42
1181	5	24	781	115
1182	1	55	199	103
1183	10	11	976	82
1184	6	44	530	94
1185	6	60	565	42
1186	4	39	246	44
1187	3	64	186	70
1188	7	10	310	30
1189	7	66	490	13
1190	10	37	968	37
1191	5	7	793	95
1192	7	81	323	11
1193	4	94	182	52
1194	8	50	594	100
1195	5	41	223	115
1196	4	58	937	60
1197	1	68	624	112
1198	5	99	536	14
1199	2	42	269	32
1200	6	4	339	48
1201	1	18	374	31
1202	6	39	703	100
1203	6	9	450	118
1204	8	86	975	26
1205	3	66	563	90
1206	10	41	744	26
1207	1	32	745	96
1208	7	93	593	74
1209	8	73	867	51
1210	6	76	358	20
1211	8	37	359	33
1212	2	73	305	112
1213	3	78	489	54
1214	6	4	700	110
1215	6	82	825	86
1216	4	49	492	76
1217	10	22	777	12
1218	10	72	869	30
1219	1	70	931	58
1220	2	23	824	107
1221	3	47	598	31

1222	4	8	857	119
1223	10	26	288	92
1224	1	91	709	68
1225	2	94	624	19
1226	1	32	698	33
1227	10	27	732	65
1228	9	48	299	18
1229	4	36	837	21
1230	2	76	396	113
1231	3	97	270	93
1232	10	96	916	73
1233	8	87	642	112
1234	2	84	557	22
1235	3	54	199	41
1236	1	45	774	80
1237	9	57	319	62
1238	7	29	677	44
1239	10	66	590	20
1240	9	78	168	58
1241	4	63	460	14
1242	8	28	949	89
1243	6	43	494	90
1244	5	59	941	72
1245	9	83	678	96
1246	5	54	283	64
1247	3	78	153	43
1248	4	78	807	55
1249	7	51	171	93
1250	10	81	671	38
1251	7	53	353	23
1252	5	27	508	14
1253	2	90	598	59
1254	5	8	376	19
1255	8	71	426	107
1256	1	69	552	15
1257	2	18	226	40
1258	8	50	526	81
1259	3	32	253	46
1260	2	10	432	66
1261	8	49	225	105
1262	7	72	811	115
1263	6	39	960	30
1264	5	79	865	64
1265	9	37	163	81
1266	3	23	336	44
1267	10	95	324	50
1268	1	37	704	36

1269	8	84	165	63
1270	3	82	602	108
1271	4	70	529	55
1272	8	87	960	55
1273	3	33	417	116
1274	3	23	311	82
1275	7	57	479	106
1276	7	65	788	77
1277	7	36	152	17
1278	1	6	199	108
1279	2	10	513	96
1280	7	31	424	70
1281	4	91	411	86
1282	3	99	830	29
1283	2	95	235	109
1284	4	55	173	119
1285	4	67	813	13
1286	7	82	467	15
1287	6	69	300	99
1288	1	93	243	115
1289	10	76	579	14
1290	2	81	624	26
1291	7	77	760	86
1292	8	37	721	37
1293	8	30	847	118
1294	10	66	921	22
1295	1	19	879	84
1296	8	97	533	96
1297	9	38	666	70
1298	6	88	166	12
1299	7	69	655	85
1300	5	8	633	103
1301	5	62	703	39
1302	3	93	623	82
1303	1	42	462	91
1304	3	77	930	88
1305	8	42	267	95
1306	6	52	915	51
1307	2	78	761	68
1308	2	96	415	81
1309	8	99	822	94
1310	7	56	879	103
1311	1	87	829	34
1312	5	68	234	12
1313	5	38	740	17
1314	4	40	423	64
1315	10	41	156	50

1316	4	30	447	35
1317	8	32	424	88
1318	7	51	396	101
1319	8	19	618	112
1320	7	54	412	44
1321	3	10	852	63
1322	5	22	288	76
1323	4	77	325	23
1324	3	66	365	115
1325	4	5	217	88
1326	3	23	743	101
1327	5	26	188	32
1328	9	35	501	96
1329	1	1	612	108
1330	4	4	626	92
1331	3	74	518	115
1332	4	72	247	72
1333	1	78	545	104
1334	9	69	869	92
1335	2	8	209	106
1336	3	95	825	33
1337	1	88	525	19
1338	8	1	585	34
1339	4	69	434	105
1340	4	6	202	56
1341	4	95	908	111
1342	3	41	228	62
1343	10	34	796	61
1344	4	60	514	63
1345	5	58	536	24
1346	3	16	679	83
1347	3	28	889	52
1348	3	79	921	55
1349	4	1	872	96
1350	1	67	308	98
1351	8	59	314	56
1352	3	99	229	97
1353	2	69	352	94
1354	1	29	967	102
1355	4	26	764	56
1356	6	66	928	60
1357	5	37	659	106
1358	3	82	323	114
1359	4	6	665	67
1360	4	97	744	63
1361	2	66	295	114
1362	10	36	196	110

1363	5	53	436	31
1364	10	29	742	76
1365	4	33	558	99
1366	4	14	649	90
1367	8	32	362	53
1368	7	49	787	60
1369	2	77	986	72
1370	8	55	995	60
1371	2	17	639	47
1372	8	76	344	99
1373	7	41	884	21
1374	9	69	428	45
1375	10	79	880	29
1376	7	84	932	38
1377	3	98	160	31
1378	2	5	731	18
1379	8	50	155	22
1380	5	72	1000	47
1381	8	73	754	18
1382	1	71	374	89
1383	6	24	787	37
1384	8	31	801	95
1385	1	67	216	15
1386	4	7	312	72
1387	10	95	839	86
1388	3	55	386	87
1389	7	70	801	98
1390	6	89	581	87
1391	2	57	610	27
1392	7	26	623	67
1393	4	71	992	13
1394	1	34	359	100
1395	7	46	425	85
1396	3	49	581	48
1397	5	31	469	34
1398	1	80	996	71
1399	7	40	582	81
1400	6	81	342	77
1401	10	75	728	109
1402	10	66	872	39
1403	1	47	308	85
1404	9	56	852	107
1405	1	71	937	56
1406	5	5	289	46
1407	9	35	615	94
1408	7	80	544	45
1409	9	55	954	73

1410	5	29	160	117
1411	2	31	266	115
1412	3	82	753	117
1413	4	38	925	115
1414	3	47	789	91
1415	8	67	809	26
1416	1	50	927	39
1417	8	8	863	71
1418	7	2	674	54
1419	9	24	507	43
1420	9	22	782	101
1421	5	24	684	49
1422	4	63	764	90
1423	10	54	796	58
1424	3	25	588	32
1425	6	36	713	63
1426	8	89	497	87
1427	1	11	846	10
1428	4	28	591	47
1429	1	12	562	89
1430	7	54	419	77
1431	2	79	452	95
1432	8	93	957	69
1433	9	28	665	100
1434	1	99	242	88
1435	1	100	422	110
1436	1	38	697	101
1437	2	53	958	65
1438	8	38	347	102
1439	6	20	307	58
1440	5	36	345	44
1441	8	31	996	79
1442	4	97	701	72
1443	9	67	484	80
1444	5	72	913	58
1445	6	76	248	44
1446	3	5	927	60
1447	4	33	774	82
1448	4	99	493	112
1449	6	79	297	66
1450	3	66	606	109
1451	4	74	920	110
1452	5	73	571	23
1453	2	1	903	113
1454	4	45	995	20
1455	5	89	211	60
1456	6	48	481	64

1457	3	62	1000	84
1458	9	88	446	87
1459	8	84	709	119
1460	8	39	560	41
1461	6	30	829	61
1462	6	40	857	83
1463	3	71	323	85
1464	8	18	939	89
1465	10	73	990	18
1466	5	68	435	97
1467	6	66	606	45
1468	1	7	368	12
1469	6	54	547	66
1470	10	95	313	76
1471	5	32	183	32
1472	6	57	721	65
1473	10	2	194	91
1474	7	93	289	112
1475	7	81	413	27
1476	4	35	327	11
1477	8	42	693	71
1478	3	97	189	101
1479	7	74	992	19
1480	8	94	475	21
1481	2	88	903	31
1482	9	85	814	63
1483	7	55	177	60
1484	7	20	572	12
1485	5	43	912	63
1486	2	39	577	37
1487	6	32	318	19
1488	5	87	380	11
1489	3	98	978	11
1490	4	51	493	104
1491	6	29	214	97
1492	8	34	944	13
1493	6	6	970	26
1494	4	100	872	109
1495	5	95	468	19
1496	10	59	533	45
1497	10	67	330	60
1498	2	77	681	54
1499	10	5	830	64
1500	8	27	166	116
1501	8	80	713	26
1502	6	86	776	97
1503	3	32	388	98

1504	9	81	594	102
1505	3	34	289	47
1506	4	23	712	15
1507	6	47	854	96
1508	6	13	208	23
1509	8	33	541	83
1510	7	87	551	39
1511	9	22	846	67
1512	8	42	386	94
1513	2	71	951	110
1514	9	22	754	64
1515	2	86	969	26
1516	6	66	568	23
1517	3	12	723	60
1518	7	95	535	15
1519	6	51	798	118
1520	6	11	603	11
1521	2	60	756	111
1522	10	70	525	21
1523	9	9	761	99
1524	9	58	311	74
1525	10	71	392	118
1526	8	7	754	110
1527	3	67	230	60
1528	6	63	222	26
1529	9	82	845	95
1530	5	1	664	45
1531	1	22	559	40
1532	2	3	207	47
1533	7	29	471	80
1534	3	62	651	116
1535	8	70	990	98
1536	6	57	828	112
1537	9	49	606	105
1538	8	47	810	100
1539	7	65	402	16
1540	8	92	729	48
1541	5	2	325	21
1542	10	85	617	69
1543	1	93	490	101
1544	6	75	831	71
1545	9	52	542	118
1546	4	83	334	31
1547	3	37	385	65
1548	10	33	158	29
1549	8	19	905	103
1550	1	65	223	94

1551	3	16	402	76
1552	3	16	857	58
1553	7	9	574	106
1554	7	7	882	42
1555	9	28	692	92
1556	10	38	444	22
1557	3	1	761	90
1558	7	9	819	86
1559	4	77	571	15
1560	4	93	731	82
1561	1	28	579	107
1562	6	99	516	39
1563	9	81	455	100
1564	3	46	420	118
1565	9	43	244	92
1566	2	39	581	20
1567	1	61	274	68
1568	6	53	823	13
1569	5	99	603	99
1570	4	47	721	17
1571	1	30	548	101
1572	5	24	181	59
1573	4	51	293	109
1574	2	98	715	105
1575	7	51	847	82
1576	9	19	952	95
1577	6	75	759	97
1578	2	40	764	89
1579	9	1	960	40
1580	7	25	453	19
1581	10	34	887	29
1582	3	26	721	19
1583	1	71	353	27
1584	5	64	238	106
1585	4	62	806	50
1586	5	58	656	32
1587	8	72	397	72
1588	6	66	387	114
1589	3	11	899	82
1590	5	100	562	81
1591	6	58	676	111
1592	2	91	470	119
1593	8	2	257	83
1594	2	64	823	102
1595	3	90	675	69
1596	2	19	550	105
1597	10	8	165	93

1598	9	57	384	71
1599	2	11	699	21
1600	3	89	222	14
1601	2	93	720	85
1602	8	79	190	62
1603	4	12	579	32
1604	1	22	916	45
1605	6	65	394	76
1606	10	27	353	103
1607	7	9	210	59
1608	3	35	982	49
1609	1	77	271	89
1610	9	95	305	27
1611	1	40	471	71
1612	1	73	678	111
1613	3	35	653	65
1614	5	1	587	19
1615	6	26	943	82
1616	2	19	633	70
1617	1	77	259	113
1618	4	50	239	62
1619	5	57	609	69
1620	7	50	704	71
1621	5	83	187	74
1622	6	23	350	75
1623	6	38	498	106
1624	7	43	642	111
1625	7	78	737	13
1626	6	12	923	72
1627	8	65	868	26
1628	6	55	153	26
1629	7	63	922	63
1630	1	3	754	26
1631	5	13	392	54
1632	5	73	603	69
1633	1	76	679	12
1634	10	89	258	86
1635	1	49	977	74
1636	2	90	853	12
1637	7	42	289	41
1638	7	55	763	63
1639	7	19	193	56
1640	10	62	541	112
1641	10	81	427	31
1642	9	37	220	60
1643	6	43	443	83
1644	3	22	724	109

1645	1	30	746	56
1646	8	34	952	38
1647	1	36	685	106
1648	3	95	765	51
1649	6	85	423	24
1650	1	91	475	85
1651	6	47	708	31
1652	7	22	917	74
1653	7	19	266	104
1654	8	100	581	90
1655	4	11	760	69
1656	6	87	987	107
1657	6	87	461	111
1658	10	43	603	104
1659	6	50	183	77
1660	9	54	303	23
1661	10	88	361	43
1662	5	77	714	16
1663	2	59	800	23
1664	8	67	386	93
1665	1	51	336	54
1666	4	78	721	33
1667	7	70	867	82
1668	9	57	274	93
1669	3	32	196	71
1670	1	18	654	33
1671	7	68	631	70
1672	9	36	338	91
1673	1	29	160	84
1674	4	89	522	66
1675	4	15	196	36
1676	1	99	295	55
1677	6	14	568	14
1678	2	2	427	63
1679	4	39	562	52
1680	10	4	448	107
1681	5	90	368	70
1682	7	3	961	45
1683	9	79	418	49
1684	6	14	365	99
1685	9	34	780	49
1686	2	63	827	85
1687	5	51	408	40
1688	7	24	634	116
1689	10	100	553	43
1690	5	11	954	15
1691	4	24	662	32

1692	6	29	183	36
1693	7	59	289	18
1694	10	96	161	106
1695	6	54	743	86
1696	9	80	778	72
1697	1	56	530	59
1698	7	67	977	63
1699	3	4	574	51
1700	7	10	973	109
1701	5	17	890	72
1702	2	75	464	102
1703	7	73	427	79
1704	6	96	474	89
1705	3	100	235	81
1706	2	63	829	19
1707	9	23	244	29
1708	5	79	608	118
1709	7	42	927	11
1710	7	78	199	19
1711	5	87	627	67
1712	3	37	584	43
1713	4	40	151	98
1714	2	100	895	93
1715	8	83	659	23
1716	8	99	752	31
1717	7	97	234	63
1718	10	76	687	63
1719	4	99	856	40
1720	1	88	652	112
1721	10	29	570	31
1722	8	2	967	50
1723	8	83	441	62
1724	1	66	496	22
1725	5	4	517	34
1726	6	23	721	77
1727	1	68	658	51
1728	3	66	678	65
1729	1	30	533	95
1730	1	47	872	102
1731	5	40	763	117
1732	1	45	310	118
1733	6	29	305	118
1734	2	54	653	77
1735	8	35	957	33
1736	7	66	436	58
1737	4	7	701	81
1738	7	60	357	106

1739	4	20	597	84
1740	9	41	995	42
1741	6	58	639	66
1742	1	34	631	92
1743	6	21	853	58
1744	5	71	614	64
1745	9	89	809	72
1746	10	29	775	40
1747	10	70	997	106
1748	6	84	193	102
1749	3	84	242	29
1750	8	61	637	72
1751	2	42	889	61
1752	2	43	649	27
1753	2	19	202	78
1754	10	85	936	116
1755	7	75	555	37
1756	2	66	904	66
1757	7	13	567	87
1758	9	26	643	84
1759	4	2	904	29
1760	8	90	919	93
1761	4	25	599	17
1762	9	70	708	67
1763	3	28	749	118
1764	5	80	560	85
1765	2	19	332	11
1766	1	1	862	27
1767	10	76	966	80
1768	7	16	801	25
1769	4	19	889	33
1770	6	30	475	62
1771	10	62	762	30
1772	5	25	603	32
1773	7	64	779	39
1774	7	72	411	76
1775	3	62	286	12
1776	3	80	220	117
1777	9	95	963	65
1778	7	91	701	20
1779	9	5	491	41
1780	3	19	870	119
1781	2	50	421	110
1782	5	66	330	94
1783	10	45	912	75
1784	9	54	470	90
1785	9	72	971	82

1786	10	4	756	106
1787	5	6	680	79
1788	10	94	886	50
1789	10	44	515	117
1790	1	35	757	107
1791	6	14	911	109
1792	6	77	879	31
1793	5	40	263	11
1794	10	22	386	84
1795	5	74	188	33
1796	1	71	410	102
1797	10	50	869	53
1798	3	42	756	66
1799	7	53	819	13
1800	7	37	685	87
1801	6	67	262	112
1802	3	66	454	52
1803	4	27	589	62
1804	3	27	941	87
1805	2	16	430	65
1806	6	3	806	27
1807	8	18	438	116
1808	2	67	172	112
1809	7	1	216	46
1810	9	43	377	42
1811	2	10	249	41
1812	2	81	528	19
1813	5	7	952	85
1814	9	92	295	47
1815	5	69	401	27
1816	5	21	203	49
1817	6	29	283	54
1818	7	92	772	17
1819	4	20	841	11
1820	1	29	584	91
1821	2	9	999	17
1822	1	85	216	60
1823	8	34	877	24
1824	8	83	171	44
1825	10	74	898	49
1826	3	67	258	49
1827	7	23	786	91
1828	9	54	912	59
1829	6	19	674	52
1830	4	53	877	26
1831	4	46	802	41
1832	9	28	294	83

1833	4	33	968	56
1834	5	40	866	10
1835	8	62	273	72
1836	6	56	784	115
1837	5	17	930	11
1838	9	23	559	47
1839	7	42	195	92
1840	8	39	627	30
1841	5	4	854	99
1842	7	45	806	107
1843	2	32	871	22
1844	9	88	975	84
1845	5	97	869	13
1846	10	30	237	112
1847	2	37	452	60
1848	7	5	171	54
1849	1	92	775	117
1850	6	22	933	53
1851	3	23	557	18
1852	3	85	803	107
1853	6	7	662	65
1854	8	24	752	12
1855	8	70	923	35
1856	1	72	348	39
1857	10	15	604	104
1858	5	8	279	68
1859	1	27	340	111
1860	7	95	382	81
1861	9	92	760	96
1862	9	45	944	108
1863	9	74	271	19
1864	10	12	922	37
1865	6	92	187	45
1866	2	11	185	82
1867	7	93	357	94
1868	2	70	420	41
1869	10	22	778	19
1870	6	86	989	64
1871	3	4	912	79
1872	2	30	283	35
1873	10	81	197	88
1874	10	29	893	96
1875	10	85	753	39
1876	9	61	435	65
1877	5	35	615	12
1878	6	77	738	75
1879	2	21	380	71

1880	3	32	939	110
1881	7	46	970	71
1882	4	82	788	77
1883	7	80	540	93
1884	3	89	465	31
1885	10	22	406	50
1886	6	90	721	78
1887	6	4	238	46
1888	10	81	653	107
1889	7	41	168	43
1890	4	3	855	17
1891	4	46	395	57
1892	7	5	350	93
1893	4	29	151	113
1894	4	66	703	13
1895	8	97	203	108
1896	4	31	702	42
1897	9	88	330	44
1898	2	3	736	71
1899	3	75	877	84
1900	5	44	802	25
1901	5	46	729	73
1902	9	29	891	47
1903	10	35	463	79
1904	3	86	629	34
1905	7	76	796	103
1906	8	90	845	100
1907	4	24	337	68
1908	7	45	344	59
1909	9	18	342	35
1910	7	85	706	98
1911	5	5	274	46
1912	4	61	759	72
1913	4	55	370	45
1914	2	55	338	54
1915	3	9	152	42
1916	7	80	854	33
1917	9	56	840	37
1918	9	65	209	86
1919	8	55	484	50
1920	8	22	567	79
1921	2	15	508	67
1922	2	90	294	89
1923	7	30	290	28
1924	6	84	460	36
1925	7	63	976	85
1926	6	49	775	34

1927	1	93	644	116
1928	1	30	556	46
1929	6	93	543	20
1930	6	93	196	100
1931	10	23	515	27
1932	5	7	995	29
1933	9	80	397	101
1934	3	13	788	14
1935	9	59	894	72
1936	10	78	838	115
1937	7	93	236	16
1938	4	58	470	89
1939	1	72	411	119
1940	7	81	254	71
1941	3	4	759	98
1942	9	34	778	57
1943	6	93	988	53
1944	7	75	673	87
1945	6	28	312	65
1946	1	8	886	48
1947	6	82	795	32
1948	1	83	518	80
1949	10	73	581	42
1950	10	20	175	20
1951	10	85	690	15
1952	8	20	661	82
1953	8	91	974	51
1954	8	79	756	31
1955	2	71	561	50
1956	3	34	171	88
1957	9	84	824	57
1958	4	70	959	69
1959	2	56	711	44
1960	8	8	797	66
1961	9	47	954	56
1962	5	62	943	65
1963	3	55	849	52
1964	10	35	730	26
1965	1	79	803	94
1966	10	12	379	24
1967	1	98	772	15
1968	10	99	328	108
1969	9	36	485	33
1970	4	2	880	21
1971	1	38	407	118
1972	3	22	380	12
1973	1	86	583	52

1974	3	4	765	92
1975	7	62	751	28
1976	5	26	262	46
1977	8	36	566	49
1978	9	87	787	111
1979	4	30	822	41
1980	2	93	370	47
1981	8	89	506	28
1982	3	27	756	16
1983	1	8	657	103
1984	4	80	893	90
1985	5	18	202	92
1986	8	28	600	17
1987	5	2	386	65
1988	9	92	454	114
1989	9	65	390	20
1990	1	30	230	61
1991	2	93	291	93
1992	7	87	274	83
1993	3	50	973	16
1994	6	65	707	66
1995	7	81	401	96
1996	4	85	401	12
1997	1	7	976	99
1998	7	32	239	73
1999	2	89	282	58
2000	7	82	451	27
2001	2	10	242	34
2002	5	34	234	66
2003	8	33	953	26
2004	9	39	751	32
2005	1	51	789	29
2006	1	40	505	58
2007	5	90	260	104
2008	10	28	197	19
2009	7	59	184	27
2010	8	32	544	68
2011	10	3	626	78
2012	6	55	286	47
2013	10	24	346	16
2014	10	9	417	39
2015	2	82	958	16
2016	1	63	614	43
2017	8	60	357	71
2018	4	56	340	100
2019	3	82	992	49
2020	5	72	396	106

2021	4	90	928	24
2022	10	23	989	107
2023	6	87	892	106
2024	9	2	628	42
2025	10	94	995	98
2026	4	38	220	88
2027	10	25	589	17
2028	9	91	218	106
2029	3	73	502	111
2030	1	3	970	91
2031	3	9	380	109
2032	9	11	250	65
2033	10	5	891	11
2034	7	30	547	54
2035	3	66	471	54
2036	8	64	361	75
2037	8	31	359	108
2038	2	58	713	75
2039	10	41	983	59
2040	10	18	536	42
2041	6	96	901	36
2042	2	53	267	59
2043	5	89	948	70
2044	4	93	806	98
2045	9	8	392	86
2046	2	85	198	95
2047	6	13	383	88
2048	7	94	260	61
2049	9	32	339	93
2050	2	47	939	105
2051	5	71	549	118
2052	2	8	598	75
2053	4	93	849	67
2054	9	70	566	20
2055	8	88	543	15
2056	3	67	337	66
2057	2	27	696	64
2058	8	21	664	103
2059	3	34	627	37
2060	7	10	202	116
2061	8	79	377	75
2062	2	31	652	53
2063	1	96	244	52
2064	8	72	708	50
2065	7	84	548	71
2066	1	17	883	23
2067	3	51	281	111

2068	8	91	567	107
2069	8	63	534	95
2070	9	55	431	89
2071	2	57	907	85
2072	9	52	728	110
2073	5	91	335	114
2074	5	81	514	109
2075	10	20	325	90
2076	4	90	586	16
2077	5	9	643	47
2078	2	24	315	16
2079	10	75	194	80
2080	5	80	508	119
2081	6	63	880	46
2082	6	66	412	66
2083	5	15	974	89
2084	3	83	623	47
2085	9	66	823	96
2086	6	33	564	96
2087	1	88	590	57
2088	2	13	668	24
2089	3	20	736	119
2090	3	85	275	74
2091	7	42	922	75
2092	7	51	191	87
2093	8	12	453	103
2094	2	46	788	88
2095	6	73	670	76
2096	2	68	497	102
2097	7	20	862	103
2098	8	89	758	111
2099	7	6	345	32
2100	3	30	840	106
2101	1	20	561	78
2102	10	81	426	87
2103	10	88	492	23
2104	7	13	460	73
2105	6	79	487	81
2106	9	23	595	118
2107	4	40	187	18
2108	10	69	643	76
2109	7	22	805	96
2110	8	4	163	119
2111	4	2	724	60
2112	4	14	574	40
2113	5	81	296	41
2114	7	77	628	114

2115	2	63	277	28
2116	2	42	700	67
2117	7	97	377	117
2118	5	26	914	18
2119	3	19	604	113
2120	2	59	579	75
2121	3	48	205	31
2122	3	31	289	61
2123	8	78	530	84
2124	8	89	685	82
2125	4	58	290	37
2126	7	1	361	40
2127	1	46	677	17
2128	6	36	226	81
2129	7	20	214	110
2130	2	100	990	117
2131	6	57	973	119
2132	9	77	466	71
2133	9	72	247	91
2134	6	81	643	105
2135	10	90	966	17
2136	6	59	508	71
2137	9	90	940	94
2138	9	20	272	89
2139	7	87	611	97
2140	4	39	962	18
2141	9	64	293	49
2142	4	76	860	105
2143	5	49	525	72
2144	2	82	762	54
2145	4	19	434	83
2146	7	86	504	45
2147	10	28	216	36
2148	1	23	871	85
2149	9	79	445	83
2150	10	15	725	62
2151	1	22	922	43
2152	7	38	966	11
2153	8	39	880	12
2154	5	14	692	117
2155	5	97	549	46
2156	3	99	300	69
2157	8	21	856	107
2158	5	38	929	18
2159	3	41	900	15
2160	4	24	745	32
2161	4	93	518	99

2162	3	18	299	48
2163	5	11	269	46
2164	3	79	245	17
2165	3	86	805	116
2166	4	39	204	110
2167	6	25	928	81
2168	5	22	188	96
2169	8	62	560	91
2170	3	55	609	13
2171	4	50	320	49
2172	3	62	488	40
2173	1	12	778	108
2174	10	19	154	51
2175	5	40	621	24
2176	2	46	768	19
2177	7	1	726	70
2178	9	80	260	79
2179	3	99	341	22
2180	3	19	994	70
2181	8	70	821	116
2182	2	9	832	108
2183	9	81	954	10
2184	1	58	412	25
2185	2	79	417	59
2186	1	51	226	62
2187	3	13	723	86
2188	4	6	863	34
2189	8	46	277	49
2190	5	33	676	65
2191	3	27	200	78
2192	4	70	592	70
2193	2	37	422	81
2194	1	73	405	12
2195	10	85	933	69
2196	3	42	789	86
2197	3	9	607	86
2198	7	35	806	117
2199	3	76	597	91
2200	3	83	755	105
2201	1	31	897	15
2202	2	45	571	92
2203	10	83	930	70
2204	1	57	533	26
2205	8	84	241	96
2206	1	9	918	77
2207	1	40	253	22
2208	6	60	523	88

2209	9	31	734	15
2210	2	3	569	99
2211	10	63	551	37
2212	2	57	739	118
2213	9	100	154	68
2214	7	65	394	83
2215	5	13	880	46
2216	3	13	293	56
2217	9	85	858	59
2218	2	81	630	39
2219	7	81	293	26
2220	4	16	617	93
2221	10	31	637	102
2222	10	56	365	49
2223	5	82	753	80
2224	8	30	245	53
2225	2	97	406	68
2226	1	39	932	105
2227	3	57	910	33
2228	1	47	827	107
2229	1	46	283	100
2230	5	56	773	72
2231	6	78	963	120
2232	3	56	300	14
2233	1	22	232	30
2234	9	39	424	75
2235	6	5	153	53
2236	9	65	521	100
2237	7	68	618	115
2238	1	71	951	18
2239	2	27	287	37
2240	7	36	199	44
2241	7	80	156	42
2242	6	26	853	88
2243	8	18	239	51
2244	1	72	441	11
2245	4	85	204	66
2246	10	86	301	66
2247	3	88	799	46
2248	10	49	522	109
2249	8	81	976	15
2250	6	65	696	67
2251	6	83	544	54
2252	1	53	171	45
2253	5	9	327	119
2254	6	9	783	104
2255	9	97	410	88

2256	3	36	420	22
2257	10	79	232	77
2258	5	26	252	85
2259	5	47	189	36
2260	1	53	601	36
2261	6	63	170	95
2262	9	25	156	80
2263	6	36	386	108
2264	9	7	558	33
2265	7	13	381	87
2266	7	57	887	34
2267	7	63	299	35
2268	1	28	542	71
2269	2	39	156	102
2270	7	63	629	86
2271	6	80	655	16
2272	6	94	209	74
2273	9	63	520	59
2274	1	18	988	17
2275	5	32	841	68
2276	5	72	248	99
2277	8	67	735	79
2278	1	55	615	73
2279	6	100	731	104
2280	3	68	399	47
2281	1	24	527	43
2282	9	78	520	47
2283	3	61	556	56
2284	6	74	794	39
2285	6	80	847	14
2286	10	45	194	76
2287	4	38	337	39
2288	6	15	373	89
2289	6	40	646	19
2290	6	68	593	29
2291	8	85	807	96
2292	2	54	166	66
2293	8	3	312	70
2294	9	98	733	67
2295	3	66	547	80
2296	5	92	950	114
2297	8	3	741	27
2298	9	30	580	28
2299	5	16	618	83
2300	2	66	499	95
2301	2	63	206	56
2302	1	93	356	94

2303	2	76	361	102
2304	2	93	351	106
2305	10	79	938	90
2306	2	1	450	67
2307	1	96	742	11
2308	7	63	973	91
2309	5	92	373	22
2310	1	37	577	53
2311	8	40	617	67
2312	3	10	318	22
2313	1	90	250	42
2314	4	27	646	89
2315	1	83	605	58
2316	9	12	173	20
2317	1	21	854	89
2318	10	34	291	78
2319	8	78	776	63
2320	5	16	449	108
2321	4	84	827	27
2322	9	12	380	111
2323	4	48	895	41
2324	9	39	243	80
2325	9	40	462	79
2326	10	54	907	115
2327	6	77	496	38
2328	4	59	287	60
2329	7	33	842	25
2330	10	46	274	108
2331	10	81	712	75
2332	9	75	958	16
2333	7	31	982	33
2334	10	73	560	12
2335	5	96	881	84
2336	3	20	666	31
2337	2	9	656	49
2338	6	25	867	79
2339	9	98	369	54
2340	6	10	694	39
2341	6	78	849	90
2342	3	25	306	111
2343	2	18	317	89
2344	6	53	907	39
2345	1	4	167	24
2346	1	59	372	73
2347	10	95	463	10
2348	8	70	905	59
2349	4	85	412	44

2350	7	66	239	46
2351	2	98	185	47
2352	5	17	381	116
2353	5	94	936	20
2354	1	12	193	59
2355	3	64	627	61
2356	9	63	515	84
2357	4	29	932	15
2358	10	2	810	21
2359	8	60	903	68
2360	6	12	168	78
2361	4	86	347	43
2362	2	69	341	93
2363	9	27	522	30
2364	7	50	601	25
2365	2	47	701	36
2366	3	64	848	74
2367	4	51	209	57
2368	4	74	581	33
2369	2	45	751	75
2370	5	51	918	105
2371	10	29	497	72
2372	2	37	700	16
2373	5	93	202	96
2374	8	73	794	67
2375	1	9	464	81
2376	9	51	486	66
2377	9	69	432	116
2378	10	15	297	87
2379	8	89	245	111
2380	6	70	305	28
2381	8	14	995	78
2382	1	26	934	73
2383	1	54	344	74
2384	4	35	992	112
2385	4	5	416	26
2386	5	5	212	28
2387	10	80	527	48
2388	1	76	480	104
2389	10	90	890	51
2390	2	28	598	67
2391	4	76	660	13
2392	4	6	732	116
2393	1	9	753	42
2394	6	48	757	13
2395	4	91	338	107
2396	5	100	474	111

2397	1	29	690	20
2398	7	69	852	10
2399	8	96	509	91
2400	3	4	613	35
2401	10	46	733	91
2402	10	94	585	73
2403	2	52	545	83
2404	5	31	398	110
2405	2	96	429	20
2406	3	25	511	113
2407	6	80	757	38
2408	2	37	164	30
2409	9	82	483	116
2410	1	100	620	102
2411	2	74	413	40
2412	3	85	600	33
2413	1	92	383	117
2414	8	94	747	75
2415	4	93	404	93
2416	10	89	794	58
2417	2	25	545	81
2418	7	36	219	53
2419	4	56	611	33
2420	8	97	300	72
2421	9	81	911	87
2422	5	44	454	29
2423	10	85	279	23
2424	1	16	658	83
2425	9	8	288	98
2426	1	66	247	105
2427	7	21	794	57
2428	7	51	585	58
2429	10	1	283	104
2430	5	80	232	33
2431	5	30	468	14
2432	1	1	670	35
2433	8	60	246	22
2434	7	6	300	16
2435	3	61	276	23
2436	6	22	706	98
2437	2	83	687	13
2438	6	95	506	49
2439	1	8	819	43
2440	9	37	438	45
2441	10	8	740	49
2442	1	4	293	75
2443	9	87	924	60

2444	6	44	430	87
2445	3	73	315	37
2446	9	15	624	11
2447	1	79	285	29
2448	3	64	239	96
2449	7	13	183	112
2450	1	64	447	48
2451	10	2	390	78
2452	8	69	801	48
2453	1	54	430	81
2454	3	86	590	58
2455	3	42	191	113
2456	7	1	561	74
2457	6	31	715	71
2458	7	35	539	50
2459	4	71	321	102
2460	7	92	759	110
2461	3	33	224	30
2462	2	56	372	70
2463	7	40	390	68
2464	4	55	896	103
2465	4	10	766	18
2466	6	95	947	62
2467	10	15	652	80
2468	1	21	658	102
2469	6	84	688	62
2470	9	30	151	47
2471	7	92	191	43
2472	9	43	344	73
2473	4	35	906	49
2474	3	25	420	66
2475	1	17	382	22
2476	6	98	802	112
2477	3	18	536	103
2478	8	83	689	109
2479	9	69	677	111
2480	8	33	865	112
2481	2	76	871	50
2482	2	91	732	30
2483	4	39	573	57
2484	1	15	924	109
2485	10	4	290	53
2486	9	59	391	19
2487	3	56	749	53
2488	2	29	448	32
2489	5	51	526	45
2490	9	42	412	65

2491	4	21	243	55
2492	10	9	631	86
2493	6	84	961	34
2494	9	12	985	96
2495	1	14	503	32
2496	2	18	827	50
2497	10	34	501	12
2498	1	92	463	110
2499	8	7	427	33
2500	9	51	801	43
2501	7	70	703	105
2502	9	36	547	106
2503	10	83	887	73
2504	6	4	858	41
2505	10	54	816	38
2506	4	55	856	13
2507	3	20	282	28
2508	6	97	835	46
2509	6	99	500	86
2510	5	12	244	88
2511	8	23	493	108
2512	1	32	203	64
2513	9	63	886	49
2514	6	11	510	111
2515	7	81	512	95
2516	6	55	274	18
2517	10	10	189	76
2518	2	69	972	42
2519	8	43	525	30
2520	2	72	565	85
2521	6	95	900	105
2522	7	19	796	28
2523	9	11	530	88
2524	5	26	505	43
2525	7	86	953	85
2526	2	25	959	18
2527	3	67	952	100
2528	7	33	236	90
2529	7	42	344	49
2530	8	37	635	51
2531	5	62	402	22
2532	5	73	645	78
2533	3	88	710	25
2534	5	43	269	48
2535	5	32	588	32
2536	6	92	645	57
2537	10	16	441	48

2538	5	32	207	62
2539	5	49	769	105
2540	3	54	313	65
2541	8	42	221	38
2542	8	44	314	43
2543	2	81	263	90
2544	8	17	270	92
2545	10	84	335	47
2546	7	82	520	66
2547	10	42	755	16
2548	3	45	817	106
2549	5	41	385	65
2550	8	58	922	82
2551	8	32	562	98
2552	6	78	545	101
2553	4	25	597	12
2554	7	18	525	27
2555	1	30	648	76
2556	3	43	496	55
2557	8	94	916	96
2558	1	83	911	117
2559	6	68	953	112
2560	6	60	834	70
2561	2	92	468	69
2562	2	4	301	93
2563	6	88	871	109
2564	2	5	367	52
2565	7	77	396	97
2566	4	16	472	70
2567	2	47	298	15
2568	1	15	308	80
2569	9	84	730	115
2570	7	30	932	23
2571	4	52	631	85
2572	8	67	533	31
2573	8	76	266	68
2574	5	20	254	114
2575	1	84	440	108
2576	3	88	773	77
2577	8	11	309	95
2578	9	42	250	100
2579	1	27	496	119
2580	5	69	937	62
2581	8	92	844	103
2582	9	33	557	30
2583	5	55	973	64
2584	2	24	261	23

2585	5	93	209	78
2586	3	93	582	29
2587	6	76	408	36
2588	2	78	532	104
2589	9	4	423	41
2590	4	44	631	62
2591	4	23	494	106
2592	7	29	326	16
2593	3	58	332	15
2594	10	32	452	104
2595	10	32	777	54
2596	6	86	806	47
2597	1	27	855	103
2598	2	18	635	84
2599	6	96	940	96
2600	7	68	690	34
2601	2	39	219	61
2602	8	60	839	21
2603	1	77	701	55
2604	9	60	907	45
2605	5	89	914	13
2606	3	38	294	46
2607	7	22	545	74
2608	9	10	809	85
2609	8	34	332	29
2610	7	85	492	74
2611	8	90	973	69
2612	9	54	530	31
2613	5	30	388	112
2614	8	71	221	38
2615	1	5	941	44
2616	3	74	676	84
2617	7	75	463	72
2618	10	3	702	101
2619	10	18	817	12
2620	5	89	359	78
2621	3	97	455	63
2622	5	91	789	113
2623	3	36	333	94
2624	5	97	900	29
2625	2	51	375	54
2626	10	83	831	52
2627	4	40	256	92
2628	6	34	915	52
2629	4	54	339	102
2630	8	91	264	48
2631	10	78	807	100

2632	8	38	323	24
2633	5	74	974	38
2634	5	72	741	65
2635	10	65	458	22
2636	10	90	660	114
2637	5	88	450	99
2638	8	26	690	66
2639	9	13	870	100
2640	9	49	563	31
2641	3	39	911	16
2642	3	75	789	62
2643	1	47	370	115
2644	7	80	156	67
2645	10	55	949	69
2646	4	38	283	87
2647	8	60	420	17
2648	8	99	442	39
2649	3	75	517	92
2650	9	88	223	81
2651	1	84	676	82
2652	9	29	383	102
2653	5	37	852	31
2654	10	54	379	27
2655	1	98	296	63
2656	8	91	736	63
2657	1	22	321	25
2658	8	15	780	117
2659	1	18	638	45
2660	3	2	312	73
2661	9	62	209	103
2662	7	13	592	71
2663	10	58	664	53
2664	6	94	528	115
2665	7	50	946	94
2666	4	22	199	40
2667	3	34	577	24
2668	8	76	533	41
2669	4	5	267	62
2670	7	67	876	10
2671	6	50	779	52
2672	5	77	459	14
2673	10	68	984	117
2674	2	3	286	17
2675	3	38	617	97
2676	9	55	191	18
2677	4	44	766	14
2678	7	4	289	90

2679	7	44	984	51
2680	2	90	997	47
2681	10	28	511	24
2682	2	19	632	14
2683	6	82	831	25
2684	6	90	827	101
2685	6	57	311	95
2686	6	98	972	46
2687	7	67	981	89
2688	4	83	805	32
2689	10	7	875	102
2690	6	69	945	24
2691	7	1	202	98
2692	5	57	978	24
2693	7	53	681	64
2694	1	85	329	91
2695	6	36	343	62
2696	4	2	314	117
2697	6	67	766	58
2698	7	86	929	43
2699	6	25	376	19
2700	1	83	229	89
2701	6	68	616	24
2702	6	61	311	86
2703	10	68	405	102
2704	10	25	311	96
2705	10	63	475	69
2706	8	62	278	102
2707	10	79	588	104
2708	4	48	822	81
2709	7	53	334	43
2710	4	44	662	114
2711	1	43	728	78
2712	3	29	623	117
2713	3	71	721	83
2714	7	29	471	46
2715	1	93	775	90
2716	10	51	876	13
2717	1	88	636	21
2718	3	49	768	15
2719	6	69	357	111
2720	4	65	613	32
2721	9	20	325	21
2722	4	91	723	42
2723	3	62	970	28
2724	3	48	460	55
2725	4	82	831	110

2726	8	50	761	87
2727	8	34	451	104
2728	9	84	631	21
2729	8	70	351	95
2730	8	72	846	50
2731	7	13	371	106
2732	5	85	966	96
2733	6	11	605	56
2734	9	97	182	106
2735	8	31	650	68
2736	1	2	172	78
2737	3	38	287	48
2738	7	62	813	28
2739	8	61	353	38
2740	8	26	353	32
2741	9	5	709	90
2742	5	98	152	54
2743	6	46	215	117
2744	6	44	387	112
2745	10	92	703	50
2746	8	63	430	94
2747	9	50	346	116
2748	8	3	381	17
2749	4	15	790	75
2750	8	32	540	44
2751	10	19	562	45
2752	9	31	758	13
2753	3	24	494	72
2754	6	85	811	97
2755	2	7	208	79
2756	5	24	366	114
2757	5	18	293	95
2758	8	79	649	86
2759	5	15	662	21
2760	10	18	966	111
2761	10	28	600	42
2762	1	48	867	26
2763	7	72	572	87
2764	10	68	376	28
2765	5	17	751	69
2766	4	50	852	39
2767	7	15	639	75
2768	8	85	332	56
2769	4	16	610	18
2770	10	66	690	47
2771	2	14	327	50
2772	4	61	300	101

2773	7	38	953	40
2774	1	17	934	67
2775	5	46	782	51
2776	9	66	747	76
2777	7	87	997	40
2778	3	44	803	63
2779	3	17	576	85
2780	10	78	706	86
2781	6	72	294	32
2782	8	10	240	69
2783	3	66	254	114
2784	6	47	375	106
2785	6	38	785	110
2786	1	29	661	107
2787	6	19	630	91
2788	8	93	421	56
2789	3	68	958	42
2790	7	65	484	47
2791	4	97	571	68
2792	5	34	197	83
2793	8	13	842	70
2794	10	13	662	37
2795	1	7	485	13
2796	9	8	156	87
2797	10	85	429	31
2798	5	46	161	116
2799	10	2	265	112
2800	3	89	472	88
2801	8	45	913	54
2802	9	45	859	71
2803	9	87	421	104
2804	6	51	294	98
2805	2	5	685	52
2806	4	8	960	26
2807	7	27	283	71
2808	1	50	700	17
2809	1	43	576	60
2810	4	31	634	68
2811	7	72	293	118
2812	2	72	477	102
2813	7	64	926	54
2814	1	22	695	69
2815	1	74	564	76
2816	1	10	541	75
2817	5	14	735	101
2818	6	9	173	87
2819	2	63	195	41

2820	2	57	162	84
2821	8	35	175	31
2822	8	53	781	85
2823	4	89	543	40
2824	5	52	840	33
2825	5	22	542	23
2826	9	31	351	28
2827	9	26	810	75
2828	4	89	965	55
2829	9	59	420	79
2830	9	10	224	66
2831	1	49	193	35
2832	6	91	513	83
2833	2	55	800	13
2834	1	3	174	112
2835	9	94	891	84
2836	8	36	629	67
2837	3	55	934	19
2838	9	74	760	66
2839	9	2	745	13
2840	9	4	332	13
2841	6	15	244	11
2842	2	75	271	104
2843	3	96	371	40
2844	7	78	689	74
2845	4	82	771	32
2846	4	87	210	69
2847	6	47	840	104
2848	10	60	869	85
2849	1	44	865	15
2850	7	37	550	43
2851	7	74	605	49
2852	10	6	641	86
2853	3	92	949	93
2854	2	59	382	29
2855	4	75	864	36
2856	9	11	551	22
2857	8	55	998	78
2858	6	31	506	43
2859	9	50	893	59
2860	6	18	728	59
2861	9	65	365	78
2862	8	59	633	99
2863	4	44	157	116
2864	2	56	613	24
2865	2	46	857	89
2866	8	39	487	59

2867	4	1	638	13
2868	10	68	566	72
2869	4	12	735	108
2870	5	11	889	41
2871	8	83	712	20
2872	7	50	816	104
2873	5	48	941	16
2874	7	95	510	46
2875	1	51	329	32
2876	1	54	447	28
2877	6	18	174	25
2878	4	6	516	28
2879	2	89	353	19
2880	3	1	362	45
2881	5	66	976	47
2882	3	59	897	82
2883	3	58	510	46
2884	9	52	175	79
2885	2	4	897	94
2886	9	9	707	84
2887	9	3	890	21
2888	1	98	622	72
2889	7	6	238	87
2890	10	85	452	118
2891	10	94	426	107
2892	3	11	582	22
2893	5	28	461	61
2894	8	47	160	33
2895	1	13	662	54
2896	5	6	320	72
2897	7	70	953	37
2898	1	20	703	14
2899	6	70	609	59
2900	10	31	877	83
2901	6	11	532	106
2902	5	21	229	67
2903	4	51	351	98
2904	9	54	168	75
2905	1	82	336	95
2906	6	62	902	87
2907	3	57	881	91
2908	9	54	370	118
2909	5	1	886	42
2910	3	44	821	61
2911	10	51	841	56
2912	9	82	813	43
2913	6	26	155	64

2914	1	60	770	78
2915	4	65	603	101
2916	9	33	294	98
2917	4	25	740	71
2918	7	73	649	59
2919	1	47	748	46
2920	3	22	889	58
2921	9	21	674	13
2922	10	36	812	55
2923	7	23	296	66
2924	4	62	630	108
2925	4	19	614	10
2926	2	8	998	59
2927	7	10	433	71
2928	7	72	270	26
2929	8	73	240	57
2930	4	65	181	66
2931	2	76	395	88
2932	1	25	448	87
2933	10	99	476	78
2934	9	40	239	85
2935	7	22	599	14
2936	9	91	889	110
2937	2	51	485	47
2938	9	13	597	82
2939	5	51	536	21
2940	4	31	894	73
2941	10	68	967	38
2942	7	36	358	105
2943	5	27	567	96
2944	2	72	853	23
2945	1	48	169	47
2946	9	19	176	117
2947	3	4	225	98
2948	6	76	641	82
2949	3	85	909	118
2950	4	17	832	27
2951	8	61	485	27
2952	10	76	572	36
2953	7	83	603	46
2954	3	50	965	89
2955	5	78	884	31
2956	3	65	789	55
2957	4	25	437	77
2958	5	76	270	83
2959	2	52	667	38
2960	2	77	490	84

2961	7	11	553	20
2962	2	28	428	76
2963	2	53	783	107
2964	4	46	532	89
2965	1	18	426	40
2966	10	22	704	80
2967	10	67	612	56
2968	7	78	622	28
2969	10	73	173	118
2970	1	37	898	94
2971	9	93	698	100
2972	7	80	568	81
2973	8	58	734	28
2974	10	33	938	34
2975	10	77	259	109
2976	9	33	203	108
2977	3	14	370	90
2978	9	83	426	54
2979	7	79	759	60
2980	6	59	674	101
2981	4	71	691	59
2982	10	95	204	105
2983	9	42	916	79
2984	4	87	835	24
2985	2	19	152	45
2986	1	28	738	114
2987	10	23	368	19
2988	7	8	789	59
2989	6	32	951	30
2990	3	61	226	55
2991	10	2	778	20
2992	2	21	394	75
2993	5	55	460	23
2994	2	67	854	114
2995	9	95	356	45
2996	3	20	493	111
2997	10	5	798	62
2998	2	19	586	83
2999	6	61	318	50
3000	4	6	648	48
3001	6	72	870	116
3002	1	51	160	67
3003	2	3	569	85
3004	8	43	302	17
3005	7	37	213	86
3006	5	95	998	95
3007	2	41	570	37

3008	3	91	522	56
3009	1	80	241	71
3010	1	28	729	63
3011	9	33	537	40
3012	6	25	517	18
3013	6	9	782	106
3014	3	7	290	18
3015	8	80	230	69
3016	8	21	526	115
3017	9	21	432	97
3018	4	54	411	12
3019	6	48	670	78
3020	3	6	174	114
3021	10	19	336	27
3022	3	98	591	80
3023	5	29	952	59
3024	4	19	487	13
3025	3	19	360	54
3026	1	59	472	18
3027	5	39	591	82
3028	1	95	360	81
3029	3	66	869	56
3030	7	56	573	60
3031	4	39	414	18
3032	10	74	787	22
3033	9	85	779	72
3034	2	98	587	44
3035	1	59	542	95
3036	1	56	825	104
3037	8	88	835	13
3038	2	4	701	33
3039	4	97	370	19
3040	6	68	309	26
3041	5	53	250	90
3042	4	79	748	88
3043	6	46	783	14
3044	10	40	955	48
3045	7	60	736	88
3046	7	71	941	82
3047	2	73	785	63
3048	5	81	437	29
3049	8	40	987	69
3050	2	32	716	96
3051	10	88	660	68
3052	1	35	854	120
3053	10	96	728	62
3054	10	23	266	89

3055	8	24	448	62
3056	9	22	662	92
3057	8	42	786	108
3058	2	89	443	57
3059	2	52	762	98
3060	9	91	828	56
3061	1	98	469	100
3062	10	84	530	21
3063	6	75	668	68
3064	6	14	248	95
3065	5	1	783	21
3066	7	56	588	57
3067	7	10	510	58
3068	3	70	976	29
3069	1	80	343	110
3070	7	42	239	73
3071	4	84	151	62
3072	8	25	852	27
3073	10	10	995	33
3074	2	36	597	63
3075	7	34	518	98
3076	3	41	888	89
3077	10	64	665	113
3078	9	6	336	110
3079	3	23	607	101
3080	8	33	745	114
3081	2	55	643	53
3082	6	68	286	119
3083	7	91	399	99
3084	6	6	300	105
3085	1	7	572	82
3086	8	27	730	56
3087	7	52	990	23
3088	5	88	201	30
3089	3	98	706	70
3090	1	11	683	52
3091	10	31	487	114
3092	7	28	702	108
3093	4	79	840	114
3094	4	78	348	19
3095	5	2	768	63
3096	3	100	180	42
3097	5	57	202	112
3098	1	7	720	18
3099	2	65	707	46
3100	3	72	260	50
3101	5	65	434	74

3102	9	40	679	48
3103	8	33	325	37
3104	2	22	276	34
3105	6	13	660	117
3106	5	12	841	21
3107	9	26	808	108
3108	4	63	731	59
3109	9	17	387	19
3110	10	70	929	69
3111	2	16	196	89
3112	8	39	168	16
3113	1	92	910	34
3114	6	29	756	42
3115	4	7	296	90
3116	4	6	899	113
3117	7	36	571	88
3118	1	77	939	111
3119	1	13	826	22
3120	8	84	325	55
3121	8	56	916	49
3122	6	92	708	28
3123	8	51	580	73
3124	5	4	306	109
3125	1	27	380	72
3126	8	61	796	45
3127	5	80	322	64
3128	2	96	269	42
3129	2	87	186	55
3130	10	1	295	51
3131	2	83	865	50
3132	7	10	546	110
3133	1	23	814	102
3134	9	6	562	54
3135	7	7	375	11
3136	4	16	349	92
3137	4	96	959	13
3138	2	67	798	96
3139	8	1	515	112
3140	2	23	713	111
3141	4	79	280	30
3142	8	8	952	81
3143	9	35	820	22
3144	8	65	214	116
3145	6	94	879	110
3146	3	37	811	39
3147	8	80	256	105
3148	3	68	602	98

3149	4	89	284	31
3150	7	29	724	98
3151	3	76	674	113
3152	2	56	794	30
3153	5	35	650	50
3154	2	89	486	116
3155	8	14	499	59
3156	7	61	623	56
3157	9	14	595	118
3158	6	67	968	21
3159	10	11	560	106
3160	10	80	442	17
3161	10	9	318	118
3162	4	39	775	45
3163	1	35	210	92
3164	7	67	967	60
3165	7	72	899	11
3166	1	57	160	14
3167	6	49	553	28
3168	7	71	525	97
3169	6	29	340	58
3170	3	33	236	82
3171	9	90	545	100
3172	3	45	875	88
3173	6	27	429	102
3174	7	89	363	91
3175	7	30	268	86
3176	9	38	927	94
3177	6	87	625	54
3178	3	34	440	51
3179	6	98	527	15
3180	10	75	574	28
3181	1	8	680	31
3182	10	68	794	15
3183	2	80	322	108
3184	4	38	758	107
3185	3	71	971	78
3186	1	62	436	103
3187	10	31	900	74
3188	2	35	754	74
3189	1	78	997	17
3190	1	76	836	28
3191	1	31	930	90
3192	5	39	864	68
3193	3	92	315	18
3194	10	70	672	104
3195	7	13	866	114

3196	9	40	469	97
3197	4	44	659	113
3198	9	10	619	112
3199	4	2	181	106
3200	6	28	497	21
3201	8	69	784	38
3202	5	64	225	89
3203	1	54	450	90
3204	9	77	225	66
3205	1	90	626	24
3206	10	65	846	91
3207	8	13	498	117
3208	10	14	894	35
3209	5	69	357	81
3210	1	17	932	118
3211	5	75	855	55
3212	6	1	943	109
3213	2	78	734	47
3214	4	87	182	16
3215	7	100	326	14
3216	4	44	456	74
3217	1	100	526	34
3218	10	17	579	95
3219	5	71	231	45
3220	3	76	463	85
3221	1	24	810	30
3222	6	65	396	78
3223	5	100	801	113
3224	9	79	240	42
3225	6	38	876	27
3226	8	49	653	98
3227	3	54	197	93
3228	9	32	364	107
3229	8	25	625	11
3230	3	5	562	31
3231	9	69	203	21
3232	2	55	199	15
3233	9	57	155	76
3234	7	59	588	111
3235	5	37	978	24
3236	6	86	984	32
3237	7	80	372	89
3238	8	34	712	116
3239	4	13	842	35
3240	6	93	637	96
3241	5	96	414	79
3242	10	7	627	51

3243	8	12	478	48
3244	4	74	804	16
3245	6	78	831	94
3246	2	62	501	79
3247	1	75	316	34
3248	10	59	365	117
3249	1	62	645	41
3250	10	100	538	80
3251	3	89	687	38
3252	2	81	177	28
3253	10	51	663	35
3254	8	92	524	119
3255	3	67	313	30
3256	10	15	913	75
3257	2	34	283	91
3258	2	4	983	108
3259	4	96	168	31
3260	10	43	926	101
3261	1	71	185	26
3262	6	81	339	44
3263	7	46	633	74
3264	10	100	652	83
3265	1	48	698	76
3266	7	70	747	20
3267	6	22	671	93
3268	4	92	935	99
3269	10	31	575	96
3270	10	49	849	27
3271	4	21	548	66
3272	3	1	920	64
3273	2	96	421	51
3274	5	79	869	93
3275	9	50	683	71
3276	9	67	761	99
3277	6	98	211	27
3278	3	88	839	116
3279	8	85	691	18
3280	5	26	350	57
3281	1	49	242	116
3282	10	35	387	36
3283	1	23	560	39
3284	8	65	988	80
3285	10	87	892	83
3286	1	92	420	34
3287	10	5	377	110
3288	8	8	366	46
3289	7	56	350	49

3290	1	77	682	94
3291	3	77	536	76
3292	7	10	544	111
3293	2	91	983	97
3294	1	22	326	33
3295	7	71	554	103
3296	1	73	569	81
3297	10	20	544	58
3298	4	27	576	117
3299	10	95	966	79
3300	9	70	229	67
3301	10	57	150	11
3302	6	59	470	59
3303	3	44	486	83
3304	2	63	195	21
3305	9	42	983	114
3306	6	44	901	32
3307	7	77	549	11
3308	9	80	388	67
3309	10	52	375	90
3310	3	34	310	86
3311	6	43	918	10
3312	10	30	519	116
3313	5	75	912	59
3314	2	79	652	38
3315	3	49	184	66
3316	5	57	401	103
3317	1	68	693	58
3318	10	5	162	81
3319	4	64	972	47
3320	10	54	570	19
3321	7	69	476	73
3322	8	31	918	116
3323	5	90	554	75
3324	4	68	822	71
3325	10	54	973	111
3326	1	60	925	32
3327	4	100	670	30
3328	1	22	598	49
3329	3	58	504	44
3330	10	31	404	34
3331	1	87	248	16
3332	1	65	526	82
3333	6	17	274	81
3334	3	27	517	51
3335	10	65	597	44
3336	1	66	553	118

3337	9	38	289	55
3338	5	64	418	88
3339	6	46	353	113
3340	6	33	768	84
3341	1	4	829	35
3342	4	71	605	100
3343	5	25	388	32
3344	2	82	366	39
3345	7	39	983	114
3346	8	66	609	43
3347	8	86	240	59
3348	6	88	837	90
3349	3	20	668	17
3350	5	80	703	11
3351	2	36	299	63
3352	1	96	337	15
3353	3	75	190	119
3354	2	85	763	115
3355	5	24	671	14
3356	8	40	626	46
3357	6	34	366	15
3358	4	89	867	11
3359	6	6	756	84
3360	7	61	780	95
3361	1	84	674	58
3362	7	52	157	51
3363	5	71	644	78
3364	7	58	756	90
3365	4	39	976	60
3366	4	46	257	109
3367	9	54	742	103
3368	1	100	925	31
3369	5	66	911	107
3370	6	30	418	32
3371	4	11	715	108
3372	10	6	252	113
3373	6	65	788	111
3374	8	95	892	96
3375	5	45	960	80
3376	8	64	474	83
3377	2	93	617	88
3378	10	69	837	106
3379	3	81	904	98
3380	10	43	279	47
3381	1	4	278	72
3382	7	26	474	62
3383	9	18	729	26

3384	2	46	415	23
3385	1	11	856	38
3386	1	36	590	20
3387	10	42	830	24
3388	1	43	338	117
3389	2	73	781	32
3390	8	45	607	94
3391	3	29	506	29
3392	3	49	542	74
3393	9	92	430	21
3394	2	82	215	110
3395	1	78	379	31
3396	9	36	508	52
3397	4	92	296	91
3398	9	18	683	51
3399	10	57	702	61
3400	8	9	480	58
3401	5	40	454	67
3402	7	74	650	63
3403	1	28	902	83
3404	4	51	401	24
3405	1	18	875	78
3406	9	38	365	54
3407	9	7	165	17
3408	6	23	817	27
3409	3	60	957	12
3410	1	25	668	54
3411	7	80	998	72
3412	1	20	776	111
3413	10	54	393	67
3414	8	30	482	74
3415	2	4	303	107
3416	2	74	945	99
3417	7	2	569	10
3418	6	39	373	80
3419	2	36	470	44
3420	4	72	154	52
3421	10	91	995	116
3422	5	34	627	10
3423	1	90	386	48
3424	10	40	402	21
3425	9	55	983	73
3426	2	52	560	31
3427	10	58	571	110
3428	5	73	887	56
3429	10	87	232	44
3430	3	35	266	70

3431	10	34	178	22
3432	3	7	325	16
3433	3	59	904	64
3434	6	6	464	40
3435	10	36	526	50
3436	5	77	894	33
3437	3	29	571	83
3438	3	67	794	111
3439	6	83	551	12
3440	8	61	755	51
3441	5	30	731	30
3442	6	26	732	31
3443	6	15	539	14
3444	1	83	790	103
3445	3	37	737	10
3446	10	9	911	44
3447	2	51	383	17
3448	8	81	603	44
3449	2	1	959	113
3450	4	25	540	94
3451	1	40	226	44
3452	9	37	932	77
3453	4	63	342	58
3454	1	33	587	82
3455	10	20	780	76
3456	6	20	267	42
3457	10	71	276	32
3458	9	22	774	63
3459	6	81	985	34
3460	2	66	263	99
3461	2	63	790	71
3462	5	2	903	77
3463	2	96	491	70
3464	6	50	959	19
3465	2	68	984	11
3466	10	97	511	32
3467	6	58	394	49
3468	6	75	372	44
3469	2	29	764	111
3470	8	45	578	119
3471	7	84	340	99
3472	1	5	741	79
3473	3	22	498	70
3474	3	13	413	62
3475	7	21	739	97
3476	6	43	296	83
3477	9	68	745	82

3478	1	74	271	13
3479	5	86	550	100
3480	1	93	857	61
3481	10	92	387	96
3482	6	55	266	71
3483	5	73	228	26
3484	2	2	795	104
3485	6	91	414	38
3486	9	87	172	19
3487	7	78	296	27
3488	3	3	281	97
3489	3	93	636	46
3490	3	7	890	118
3491	2	30	882	100
3492	3	98	219	83
3493	3	52	831	49
3494	3	100	267	22
3495	3	100	335	42
3496	4	92	569	30
3497	4	16	394	45
3498	6	37	196	97
3499	4	9	808	43
3500	6	53	690	89
3501	5	82	350	95
3502	1	69	690	82
3503	4	78	222	70
3504	3	86	614	107
3505	6	46	950	62
3506	8	66	396	52
3507	1	24	605	84
3508	4	66	786	18
3509	10	92	409	80
3510	3	72	498	66
3511	1	21	307	18
3512	3	21	862	95
3513	6	71	748	43
3514	8	63	881	119
3515	9	52	950	23
3516	8	18	232	63
3517	5	73	812	74
3518	2	42	795	14
3519	5	36	858	118
3520	5	97	829	108
3521	1	91	272	48
3522	9	25	546	93
3523	6	89	715	87
3524	8	26	453	40

3525	8	62	627	23
3526	9	18	308	16
3527	8	14	870	82
3528	4	81	978	84
3529	8	1	671	84
3530	5	60	222	63
3531	1	79	715	25
3532	8	85	968	107
3533	9	100	483	29
3534	2	90	430	51
3535	3	73	810	21
3536	1	42	453	93
3537	9	50	364	51
3538	6	26	968	116
3539	2	25	340	99
3540	2	30	162	111
3541	2	94	279	79
3542	3	42	157	113
3543	2	11	776	66
3544	8	17	864	52
3545	1	77	669	39
3546	2	36	767	116
3547	1	5	663	98
3548	6	61	394	10
3549	4	27	843	22
3550	5	53	494	84
3551	9	50	912	105
3552	7	39	835	68
3553	6	82	925	84
3554	4	31	678	99
3555	1	7	436	25
3556	1	77	474	16
3557	2	55	157	101
3558	3	23	454	100
3559	10	28	631	89
3560	9	82	853	70
3561	10	41	935	52
3562	2	92	897	79
3563	2	96	260	18
3564	8	47	990	51
3565	1	65	325	73
3566	9	31	825	10
3567	5	87	933	24
3568	7	82	576	65
3569	8	50	546	84
3570	3	78	317	27
3571	3	32	492	78

3572	3	51	394	34
3573	8	91	612	104
3574	3	54	387	111
3575	5	83	451	78
3576	4	19	497	22
3577	4	90	610	64
3578	7	78	455	40
3579	8	57	533	16
3580	4	1	547	21
3581	3	97	373	17
3582	4	59	734	72
3583	7	29	527	47
3584	3	31	312	44
3585	9	19	219	93
3586	2	16	976	14
3587	10	83	340	85
3588	8	53	459	76
3589	5	60	807	114
3590	6	97	157	31
3591	2	43	619	51
3592	2	100	797	58
3593	6	43	436	97
3594	8	19	428	61
3595	10	100	779	78
3596	7	79	352	116
3597	7	24	446	117
3598	1	9	380	101
3599	9	40	194	19
3600	2	54	691	17
3601	5	65	889	104
3602	10	45	482	99
3603	6	22	255	98
3604	9	34	714	26
3605	2	87	524	16
3606	4	3	655	57
3607	1	90	293	82
3608	8	48	484	56
3609	7	1	313	103
3610	1	89	929	78
3611	10	4	370	12
3612	1	51	619	75
3613	6	47	676	34
3614	6	64	457	84
3615	6	93	537	92
3616	5	28	361	63
3617	8	48	163	20
3618	8	69	729	37

3619	4	97	217	29
3620	10	29	397	83
3621	2	65	270	64
3622	6	86	596	51
3623	3	92	908	37
3624	2	49	802	80
3625	10	16	258	37
3626	9	65	876	27
3627	8	82	557	70
3628	5	98	167	71
3629	8	8	273	108
3630	9	5	174	66
3631	2	70	665	28
3632	3	58	744	12
3633	1	88	915	54
3634	9	13	556	86
3635	9	94	866	45
3636	9	36	169	10
3637	8	76	537	68
3638	9	2	452	106
3639	8	33	337	84
3640	6	40	937	58
3641	8	50	505	32
3642	9	30	224	77
3643	2	63	330	23
3644	9	20	355	63
3645	6	50	182	104
3646	4	74	332	52
3647	8	17	851	32
3648	10	88	623	116
3649	7	89	345	29
3650	8	39	530	26
3651	10	46	206	90
3652	2	56	942	24
3653	8	80	200	13
3654	3	40	429	117
3655	5	77	642	97
3656	10	42	251	21
3657	5	17	769	27
3658	3	28	500	94
3659	10	22	772	112
3660	7	87	511	33
3661	4	58	641	52
3662	6	97	463	25
3663	6	69	939	54
3664	9	42	919	77
3665	6	50	282	103

3666	3	29	930	60
3667	6	24	511	59
3668	2	91	168	41
3669	5	88	883	19
3670	3	87	150	18
3671	1	98	536	63
3672	3	90	509	114
3673	4	59	345	60
3674	8	4	229	94
3675	6	23	916	15
3676	7	1	163	109
3677	9	99	432	97
3678	3	40	204	64
3679	3	9	447	75
3680	4	49	977	46
3681	10	28	490	71
3682	4	66	573	49
3683	8	15	786	91
3684	2	61	861	17
3685	1	93	256	78
3686	7	17	888	11
3687	5	97	584	33
3688	1	15	464	59
3689	3	55	768	114
3690	4	80	565	38
3691	9	10	400	20
3692	2	77	423	27
3693	2	19	394	15
3694	10	53	682	32
3695	4	35	377	71
3696	10	71	792	91
3697	2	32	478	92
3698	1	49	749	84
3699	7	80	269	87
3700	3	23	236	77
3701	7	46	650	16
3702	5	31	995	36
3703	10	27	764	86
3704	6	33	638	51
3705	9	41	822	73
3706	4	71	730	65
3707	5	10	424	97
3708	4	40	264	27
3709	9	16	973	53
3710	8	4	794	29
3711	2	41	771	76
3712	5	7	694	97

3713	10	51	452	99
3714	8	17	652	54
3715	9	74	236	108
3716	2	3	436	85
3717	9	29	656	107
3718	8	57	683	108
3719	3	92	569	65
3720	5	89	451	111
3721	7	9	889	17
3722	7	47	235	66
3723	2	3	459	84
3724	4	91	456	95
3725	3	96	632	118
3726	10	37	289	75
3727	10	51	814	33
3728	8	65	176	114
3729	1	25	576	87
3730	7	57	459	101
3731	3	97	713	112
3732	1	28	563	97
3733	5	39	336	46
3734	6	36	858	118
3735	3	22	816	86
3736	4	11	454	100
3737	2	52	512	66
3738	8	27	268	27
3739	8	23	897	79
3740	6	95	694	86
3741	6	85	208	67
3742	8	94	300	14
3743	3	14	643	84
3744	10	76	980	37
3745	8	62	221	70
3746	2	70	165	87
3747	1	58	577	25
3748	10	2	612	48
3749	4	9	475	116
3750	7	47	249	59
3751	9	99	351	58
3752	1	52	311	60
3753	2	67	197	11
3754	6	93	824	44
3755	10	8	613	73
3756	8	49	980	73
3757	7	82	464	104
3758	10	60	472	77
3759	2	54	602	87

3760	5	86	824	60
3761	8	73	631	45
3762	4	28	958	43
3763	8	41	214	32
3764	7	63	301	94
3765	6	63	994	45
3766	8	17	190	47
3767	1	26	406	81
3768	9	3	177	12
3769	5	65	672	41
3770	2	26	385	115
3771	10	16	471	19
3772	9	39	454	24
3773	4	14	501	77
3774	7	44	347	19
3775	1	20	500	15
3776	1	30	733	33
3777	2	94	768	54
3778	9	40	659	76
3779	5	91	777	90
3780	1	44	450	75
3781	8	34	197	68
3782	2	83	899	109
3783	9	49	455	74
3784	7	25	383	78
3785	5	68	440	43
3786	6	53	702	47
3787	8	8	875	118
3788	6	81	476	40
3789	1	8	407	102
3790	9	84	744	117
3791	6	23	565	27
3792	4	43	476	24
3793	2	47	830	39
3794	5	14	195	52
3795	3	67	888	111
3796	1	71	966	80
3797	3	2	836	114
3798	8	4	315	91
3799	6	12	870	65
3800	6	74	263	53
3801	6	53	990	111
3802	3	62	525	15
3803	3	19	180	119
3804	4	54	907	36
3805	4	44	521	76
3806	4	58	887	89

3807	6	51	702	93
3808	1	18	789	109
3809	10	50	648	15
3810	5	71	735	48
3811	2	52	998	30
3812	1	15	522	20
3813	8	2	862	95
3814	6	11	573	109
3815	10	30	476	39
3816	10	48	392	81
3817	5	93	950	64
3818	1	88	386	31
3819	1	2	748	50
3820	10	71	640	87
3821	7	14	685	81
3822	1	48	643	96
3823	6	6	598	64
3824	3	2	514	62
3825	7	12	645	34
3826	1	38	820	38
3827	5	3	944	92
3828	6	38	489	83
3829	3	77	333	86
3830	6	96	888	13
3831	7	60	911	119
3832	2	36	560	119
3833	2	76	441	65
3834	1	81	581	62
3835	3	20	552	62
3836	2	75	682	112
3837	6	10	348	117
3838	6	53	196	94
3839	9	81	678	74
3840	4	97	523	26
3841	6	5	246	30
3842	3	64	294	18
3843	4	90	900	43
3844	4	45	367	86
3845	9	46	670	60
3846	7	68	635	63
3847	1	26	780	78
3848	5	19	307	57
3849	5	39	339	28
3850	10	28	847	50
3851	5	65	933	62
3852	3	98	432	105
3853	8	89	379	67

3854	6	4	586	71
3855	1	50	513	53
3856	10	24	857	85
3857	2	2	528	109
3858	6	20	835	15
3859	5	6	639	78
3860	4	62	866	39
3861	2	46	513	71
3862	3	56	735	59
3863	8	23	857	78
3864	1	20	303	69
3865	6	40	679	21
3866	7	78	900	31
3867	6	75	760	109
3868	7	28	770	42
3869	7	56	618	91
3870	5	71	830	59
3871	1	10	626	23
3872	1	65	400	61
3873	3	1	933	69
3874	5	19	538	103
3875	10	52	486	29
3876	1	30	187	51
3877	5	72	928	101
3878	7	62	779	20
3879	7	27	801	70
3880	4	90	331	67
3881	9	30	756	112
3882	4	15	958	40
3883	7	79	778	46
3884	9	8	327	112
3885	8	17	831	36
3886	10	43	165	36
3887	9	22	897	47
3888	1	99	814	75
3889	7	26	217	16
3890	4	26	395	91
3891	4	16	574	95
3892	2	26	504	11
3893	7	28	411	18
3894	6	5	414	82
3895	2	39	953	86
3896	5	9	666	53
3897	9	50	942	107
3898	7	32	580	16
3899	10	79	323	116
3900	9	94	277	72

3901	5	46	823	57
3902	7	36	269	52
3903	10	49	668	89
3904	6	25	271	12
3905	9	46	812	106
3906	8	84	574	79
3907	9	41	953	28
3908	10	47	847	43
3909	10	64	949	86
3910	4	80	723	75
3911	8	38	747	89
3912	3	8	909	21
3913	4	96	230	58
3914	7	9	893	34
3915	7	5	823	39
3916	5	52	612	31
3917	5	42	538	21
3918	9	100	700	25
3919	3	52	553	73
3920	4	88	783	85
3921	7	63	429	61
3922	1	75	954	38
3923	4	42	845	87
3924	5	68	178	79
3925	1	11	471	43
3926	2	70	506	88
3927	3	20	516	40
3928	6	32	514	77
3929	1	70	601	117
3930	7	72	871	101
3931	4	61	805	75
3932	6	67	675	25
3933	9	22	166	22
3934	1	46	271	79
3935	6	85	842	69
3936	7	48	231	78
3937	7	69	667	118
3938	2	12	806	41
3939	7	70	448	111
3940	2	6	478	95
3941	4	73	607	51
3942	7	28	212	37
3943	8	35	744	77
3944	6	92	174	15
3945	4	3	323	50
3946	9	71	425	58
3947	2	27	254	103

3948	6	52	541	26
3949	8	23	255	115
3950	6	65	987	11
3951	9	45	757	25
3952	9	55	551	21
3953	1	54	317	112
3954	6	17	170	57
3955	9	10	194	32
3956	3	86	475	11
3957	8	24	844	103
3958	1	78	769	17
3959	3	51	481	34
3960	2	5	649	103
3961	1	11	861	75
3962	10	61	924	32
3963	1	28	716	14
3964	2	58	697	71
3965	9	57	957	55
3966	8	36	920	76
3967	3	95	259	64
3968	5	67	610	92
3969	2	28	569	87
3970	6	19	392	74
3971	5	66	858	51
3972	10	20	424	38
3973	7	10	331	91
3974	7	90	718	89
3975	1	95	839	60
3976	2	46	749	105
3977	7	8	181	56
3978	3	37	633	14
3979	2	73	304	116
3980	5	63	878	54
3981	5	84	445	28
3982	6	66	697	119
3983	4	88	155	103
3984	1	90	993	82
3985	1	46	866	43
3986	4	3	401	116
3987	4	62	259	48
3988	2	85	944	31
3989	7	8	539	100
3990	4	27	990	76
3991	5	88	855	28
3992	2	59	414	61
3993	9	53	507	115
3994	6	65	341	90

3995	2	5	610	99
3996	10	89	592	68
3997	9	35	962	58
3998	3	24	626	28
3999	6	69	501	41
4000	3	75	279	69
4001	9	81	716	107
4002	5	5	642	33
4003	7	42	169	90
4004	3	98	993	51
4005	7	76	260	29
4006	6	7	340	19
4007	7	91	997	42
4008	2	4	396	89
4009	4	15	780	47
4010	3	100	494	16
4011	4	32	851	96
4012	10	63	505	92
4013	6	67	431	78
4014	10	37	380	80
4015	8	11	712	42
4016	6	88	675	99
4017	2	68	643	58
4018	4	4	207	66
4019	9	48	471	62
4020	5	29	953	51
4021	4	57	822	107
4022	7	56	979	76
4023	10	66	717	68
4024	7	52	239	61
4025	9	41	160	44
4026	9	100	876	110
4027	6	26	537	88
4028	2	90	422	111
4029	10	60	207	106
4030	2	99	377	72
4031	4	1	517	56
4032	9	43	897	94
4033	3	56	594	54
4034	7	33	766	92
4035	7	86	962	63
4036	3	12	608	108
4037	8	62	239	77
4038	5	1	603	100
4039	10	71	258	111
4040	9	19	514	59
4041	8	40	683	108

4042	5	74	289	58
4043	5	37	417	89
4044	6	13	667	23
4045	6	55	940	94
4046	8	73	612	109
4047	10	40	914	105
4048	2	91	950	84
4049	10	53	898	39
4050	5	21	299	76
4051	5	30	609	35
4052	3	24	529	109
4053	7	13	970	36
4054	10	58	576	91
4055	8	19	762	65
4056	3	63	795	63
4057	8	45	571	115
4058	8	19	486	24
4059	5	40	892	98
4060	4	71	914	92
4061	6	54	975	109
4062	5	44	336	62
4063	7	100	879	26
4064	3	38	963	108
4065	5	23	669	31
4066	4	42	427	86
4067	5	96	210	54
4068	4	48	904	92
4069	2	68	672	64
4070	2	7	509	100
4071	7	47	977	112
4072	6	22	194	68
4073	2	67	283	108
4074	10	61	749	11
4075	9	7	420	64
4076	8	22	721	78
4077	5	1	710	71
4078	6	38	293	26
4079	3	84	869	88
4080	2	50	863	11
4081	4	100	464	86
4082	4	81	991	86
4083	1	24	795	67
4084	5	71	366	42
4085	9	85	813	74
4086	1	16	494	22
4087	9	2	760	18
4088	1	35	441	103

4089	6	80	926	32
4090	9	64	976	86
4091	3	93	717	60
4092	8	51	908	29
4093	7	54	156	39
4094	1	20	694	114
4095	1	35	379	68
4096	4	87	914	119
4097	2	83	690	32
4098	4	52	588	59
4099	1	90	336	70
4100	4	38	781	13
4101	4	19	821	99
4102	3	86	785	51
4103	5	56	335	86
4104	7	35	464	21
4105	7	96	957	27
4106	7	69	507	69
4107	9	86	922	23
4108	7	43	691	65
4109	3	67	736	63
4110	2	93	903	26
4111	1	79	602	94
4112	9	33	545	39
4113	2	47	858	88
4114	7	15	498	23
4115	10	78	649	69
4116	5	49	972	34
4117	4	35	743	31
4118	1	32	626	33
4119	7	84	398	11
4120	3	54	324	66
4121	4	94	542	53
4122	5	25	357	97
4123	1	52	247	44
4124	2	10	759	70
4125	1	55	299	33
4126	8	44	817	60
4127	3	23	279	53
4128	7	53	578	103
4129	8	94	541	55
4130	5	69	966	43
4131	2	8	704	38
4132	3	11	655	118
4133	9	36	504	19
4134	10	54	486	30
4135	4	42	527	50

4136	10	82	892	115
4137	6	11	813	71
4138	8	33	348	85
4139	2	72	634	105
4140	1	69	412	29
4141	1	47	870	74
4142	6	94	547	19
4143	8	97	911	95
4144	3	13	366	66
4145	10	100	328	43
4146	2	70	886	26
4147	9	92	982	94
4148	9	1	968	64
4149	5	45	858	101
4150	6	1	889	113
4151	8	53	984	102
4152	6	70	982	22
4153	6	32	571	111
4154	10	77	489	70
4155	7	89	913	41
4156	5	39	636	47
4157	4	47	260	69
4158	3	75	950	108
4159	4	50	911	48
4160	7	56	870	104
4161	7	27	235	101
4162	8	71	877	62
4163	9	53	177	24
4164	8	35	754	11
4165	10	11	949	49
4166	10	1	395	48
4167	2	22	274	106
4168	9	91	434	70
4169	6	88	644	119
4170	5	80	658	118
4171	2	25	309	43
4172	3	92	387	31
4173	1	41	671	54
4174	8	80	757	68
4175	1	83	813	34
4176	5	1	939	63
4177	9	79	656	64
4178	2	24	421	113
4179	10	48	406	116
4180	10	8	490	20
4181	4	57	964	100
4182	5	30	379	77

4183	8	17	348	108
4184	4	8	642	110
4185	1	59	595	30
4186	2	86	283	116
4187	6	46	243	73
4188	1	89	522	32
4189	5	90	304	71
4190	4	3	511	68
4191	10	25	828	97
4192	6	36	395	20
4193	8	39	772	65
4194	1	71	940	20
4195	8	2	511	19
4196	9	87	270	83
4197	9	44	541	33
4198	7	52	468	52
4199	2	50	764	113
4200	6	90	837	119
4201	9	29	525	53
4202	8	11	575	113
4203	9	49	195	39
4204	5	74	929	41
4205	10	79	259	13
4206	9	75	541	61
4207	10	36	946	53
4208	2	37	561	75
4209	10	27	356	49
4210	1	46	375	72
4211	8	65	423	27
4212	8	37	336	27
4213	7	14	574	120
4214	10	78	766	112
4215	3	67	488	40
4216	4	61	319	105
4217	2	24	331	80
4218	7	34	804	117
4219	4	87	357	84
4220	2	75	212	112
4221	8	89	402	24
4222	5	50	647	17
4223	7	70	201	109
4224	4	74	546	40
4225	9	67	731	65
4226	10	16	525	28
4227	6	75	409	41
4228	10	68	760	82
4229	6	77	771	92

4230	10	48	689	33
4231	9	82	556	15
4232	8	69	686	14
4233	3	23	663	87
4234	2	86	537	109
4235	3	30	888	106
4236	2	52	468	46
4237	10	30	310	18
4238	2	5	633	43
4239	10	53	168	49
4240	1	36	322	108
4241	4	15	625	116
4242	4	47	768	11
4243	4	17	334	84
4244	8	16	272	118
4245	3	35	610	92
4246	1	47	347	13
4247	4	27	774	105
4248	10	90	417	70
4249	6	91	479	92
4250	7	39	166	54
4251	3	100	323	47
4252	6	80	169	63
4253	1	36	234	79
4254	6	3	832	75
4255	1	99	826	78
4256	6	74	407	97
4257	6	31	526	114
4258	9	80	303	79
4259	9	97	616	32
4260	1	32	194	104
4261	9	30	292	79
4262	3	70	221	26
4263	3	22	679	62
4264	9	40	504	22
4265	2	84	550	111
4266	10	4	587	27
4267	6	34	777	70
4268	1	6	646	48
4269	2	63	355	87
4270	2	37	945	102
4271	7	76	658	116
4272	2	64	699	18
4273	8	25	243	61
4274	4	61	373	26
4275	9	79	633	22
4276	1	79	723	11

4277	5	34	852	118
4278	3	24	848	57
4279	7	39	495	47
4280	4	51	730	34
4281	5	8	169	12
4282	9	46	335	117
4283	9	6	404	79
4284	10	87	667	116
4285	6	73	888	118
4286	10	15	566	78
4287	5	84	164	88
4288	10	13	151	84
4289	9	20	705	116
4290	9	25	549	46
4291	6	45	526	73
4292	10	70	608	42
4293	5	15	861	11
4294	6	38	801	113
4295	3	80	385	42
4296	5	44	204	42
4297	6	67	657	17
4298	9	34	366	109
4299	1	81	298	116
4300	2	99	371	70
4301	3	13	489	74
4302	4	78	565	40
4303	7	42	263	103
4304	7	68	708	67
4305	1	17	673	119
4306	4	96	962	94
4307	8	3	461	48
4308	5	22	233	102
4309	2	73	786	24
4310	2	86	619	18
4311	10	49	561	26
4312	7	1	440	96
4313	9	85	368	119
4314	2	4	331	48
4315	6	26	357	104
4316	4	69	946	81
4317	4	28	610	80
4318	1	61	223	16
4319	6	7	700	83
4320	1	12	187	78
4321	7	89	245	90
4322	9	36	286	119
4323	2	55	727	56

4324	5	10	450	27
4325	4	63	544	76
4326	6	57	792	87
4327	1	11	541	46
4328	4	95	789	45
4329	7	85	743	31
4330	4	58	818	34
4331	7	43	231	21
4332	10	39	913	39
4333	8	57	509	113
4334	3	98	308	35
4335	2	9	153	54
4336	4	85	455	36
4337	8	64	961	90
4338	7	49	513	73
4339	6	14	153	72
4340	9	29	557	36
4341	10	30	458	27
4342	7	69	588	41
4343	7	1	953	66
4344	10	77	670	98
4345	8	70	471	35
4346	2	78	473	16
4347	1	84	286	80
4348	10	44	907	115
4349	3	74	752	72
4350	2	38	208	34
4351	7	22	588	98
4352	7	63	316	118
4353	4	1	809	59
4354	1	45	692	40
4355	9	82	452	14
4356	1	56	533	94
4357	9	96	515	80
4358	6	12	286	119
4359	7	31	240	33
4360	1	67	950	72
4361	9	19	694	52
4362	8	46	789	23
4363	9	7	852	31
4364	4	87	808	120
4365	3	79	772	65
4366	7	17	165	89
4367	10	70	567	100
4368	7	55	285	111
4369	6	20	285	109
4370	5	52	853	27

4371	2	49	390	30
4372	1	50	715	24
4373	8	93	470	73
4374	7	54	858	101
4375	8	53	721	58
4376	2	56	229	62
4377	6	54	861	47
4378	10	86	682	71
4379	8	71	638	105
4380	2	64	269	62
4381	6	88	675	40
4382	3	46	950	101
4383	4	69	453	31
4384	1	63	577	93
4385	6	38	792	46
4386	6	68	844	90
4387	9	2	980	13
4388	2	95	634	79
4389	10	96	806	107
4390	2	40	249	93
4391	3	88	192	46
4392	6	42	467	19
4393	10	23	534	93
4394	9	16	426	65
4395	1	57	232	95
4396	6	68	954	56
4397	3	11	495	71
4398	1	39	152	29
4399	6	69	815	19
4400	4	7	782	108
4401	9	52	506	93
4402	4	86	859	13
4403	1	45	498	114
4404	8	30	391	72
4405	5	50	284	56
4406	7	41	710	103
4407	1	14	536	91
4408	10	63	718	87
4409	6	99	349	113
4410	8	71	482	33
4411	8	3	509	87
4412	5	95	263	112
4413	10	61	299	99
4414	1	97	354	12
4415	9	41	838	88
4416	3	46	399	77
4417	8	29	422	82

4418	5	54	272	99
4419	4	32	534	37
4420	3	17	344	53
4421	9	62	306	80
4422	4	91	826	49
4423	4	61	178	106
4424	3	55	209	73
4425	9	29	417	59
4426	5	94	328	30
4427	3	2	187	83
4428	1	83	366	46
4429	5	74	299	109
4430	6	81	315	56
4431	5	67	976	114
4432	4	86	218	27
4433	10	76	218	73
4434	6	13	713	77
4435	4	83	1000	103
4436	8	47	783	97
4437	3	26	767	36
4438	8	12	301	109
4439	6	65	990	116
4440	10	60	575	119
4441	6	30	433	87
4442	8	7	696	42
4443	5	100	186	76
4444	3	48	234	35
4445	10	36	950	91
4446	5	84	603	46
4447	7	54	510	118
4448	5	28	299	105
4449	3	24	909	89
4450	9	63	657	109
4451	10	46	234	104
4452	10	75	767	104
4453	9	83	163	91
4454	7	38	487	70
4455	10	50	749	99
4456	8	72	435	54
4457	6	51	631	80
4458	1	34	695	107
4459	2	38	329	115
4460	4	16	420	87
4461	6	74	515	106
4462	8	66	725	87
4463	10	24	202	112
4464	9	30	505	47

4465	1	48	699	81
4466	4	99	197	34
4467	1	60	210	31
4468	1	45	352	30
4469	3	19	797	25
4470	10	76	665	114
4471	9	89	893	63
4472	6	41	178	69
4473	5	65	190	44
4474	6	81	901	115
4475	9	86	304	44
4476	9	4	191	54
4477	4	44	263	28
4478	8	68	506	85
4479	7	33	703	59
4480	5	71	184	65
4481	2	77	195	102
4482	10	19	731	40
4483	8	74	693	38
4484	2	92	274	120
4485	5	11	383	69
4486	6	30	278	64
4487	3	27	677	108
4488	9	93	792	72
4489	10	99	744	69
4490	1	41	479	34
4491	6	72	615	12
4492	4	17	608	84
4493	10	7	331	48
4494	4	47	627	83
4495	4	1	961	50
4496	9	47	395	93
4497	6	67	808	47
4498	5	24	393	18
4499	6	89	193	69
4500	10	7	776	42
4501	3	76	656	92
4502	10	84	312	51
4503	1	93	678	57
4504	4	12	995	30
4505	9	19	533	65
4506	10	7	274	42
4507	9	44	993	101
4508	7	63	807	84
4509	9	80	679	89
4510	2	24	599	59
4511	4	62	874	44

4512	10	32	198	25
4513	3	41	609	52
4514	4	83	530	20
4515	1	44	323	31
4516	10	15	446	112
4517	5	35	677	112
4518	1	25	424	95
4519	4	30	354	79
4520	1	87	316	93
4521	9	18	462	52
4522	2	36	783	91
4523	10	67	498	65
4524	6	58	234	66
4525	10	40	345	89
4526	3	92	998	49
4527	8	32	546	47
4528	6	7	577	50
4529	3	91	387	11
4530	3	29	287	45
4531	10	43	805	90
4532	6	52	994	98
4533	1	55	338	34
4534	4	22	328	114
4535	2	91	697	89
4536	8	84	480	57
4537	7	1	806	12
4538	4	35	994	40
4539	6	51	178	60
4540	6	73	871	53
4541	4	82	950	73
4542	5	33	594	50
4543	4	23	397	70
4544	4	60	726	17
4545	10	78	357	89
4546	10	14	538	60
4547	4	13	427	79
4548	5	10	982	112
4549	4	24	467	47
4550	5	100	324	51
4551	4	96	942	46
4552	1	48	834	41
4553	9	98	468	60
4554	6	86	612	105
4555	8	38	625	49
4556	9	24	970	84
4557	8	4	859	111
4558	7	35	286	35

4559	5	87	417	90
4560	5	14	280	96
4561	3	8	262	17
4562	3	3	165	119
4563	2	35	548	110
4564	3	75	610	74
4565	2	81	391	36
4566	2	63	978	104
4567	3	2	372	16
4568	2	70	581	56
4569	3	99	486	81
4570	5	38	593	99
4571	9	71	380	95
4572	8	87	684	18
4573	1	68	382	57
4574	10	82	229	76
4575	7	67	270	113
4576	10	76	692	34
4577	2	97	332	42
4578	2	23	843	49
4579	8	100	956	103
4580	9	92	543	57
4581	9	27	582	88
4582	2	43	255	84
4583	6	14	974	20
4584	1	3	160	24
4585	1	32	733	14
4586	3	4	273	113
4587	2	59	289	97
4588	7	96	898	92
4589	6	66	185	55
4590	5	2	369	27
4591	7	84	256	31
4592	4	1	672	41
4593	3	37	540	114
4594	9	63	767	13
4595	8	45	822	65
4596	3	45	744	79
4597	9	47	223	13
4598	8	63	850	85
4599	4	18	746	109
4600	4	76	929	117
4601	9	12	883	33
4602	9	23	452	85
4603	10	55	213	80
4604	3	9	360	54
4605	7	89	542	65

4606	1	25	970	89
4607	5	87	560	14
4608	6	52	959	34
4609	5	41	709	68
4610	4	56	284	85
4611	1	82	507	54
4612	6	56	154	10
4613	4	73	708	70
4614	10	43	520	61
4615	2	79	603	97
4616	1	25	596	114
4617	6	71	508	63
4618	9	64	524	35
4619	7	21	829	44
4620	5	66	465	63
4621	2	57	594	31
4622	4	97	600	54
4623	7	43	514	30
4624	4	22	611	96
4625	1	73	614	22
4626	8	4	730	42
4627	2	59	819	29
4628	3	58	626	23
4629	9	93	650	48
4630	4	61	263	57
4631	3	79	233	69
4632	5	92	549	71
4633	7	36	890	71
4634	6	58	706	76
4635	2	13	339	31
4636	7	17	681	33
4637	10	99	153	70
4638	6	85	909	87
4639	8	19	402	27
4640	6	60	929	44
4641	10	57	944	112
4642	2	59	702	37
4643	9	36	569	24
4644	2	24	654	56
4645	8	15	239	103
4646	9	84	599	115
4647	3	93	752	17
4648	9	48	437	12
4649	8	81	876	26
4650	5	71	874	106
4651	10	66	527	85
4652	4	34	784	96

4653	1	18	662	95
4654	4	81	361	80
4655	8	29	714	110
4656	4	76	893	107
4657	1	71	856	16
4658	3	90	847	68
4659	10	44	928	111
4660	4	13	155	50
4661	10	81	495	61
4662	5	73	407	65
4663	10	2	565	10
4664	8	4	403	42
4665	2	35	383	73
4666	5	84	240	30
4667	5	78	914	103
4668	8	1	539	62
4669	9	56	197	106
4670	3	20	714	110
4671	5	14	555	57
4672	8	66	241	55
4673	4	4	629	119
4674	7	85	989	94
4675	10	70	430	26
4676	7	61	998	71
4677	2	21	827	29
4678	10	95	849	92
4679	1	18	565	49
4680	7	19	764	43
4681	4	69	505	56
4682	5	53	434	113
4683	4	10	564	119
4684	5	43	230	90
4685	10	100	375	35
4686	3	49	756	115
4687	2	62	298	52
4688	6	64	852	98
4689	10	89	923	44
4690	9	29	661	55
4691	9	23	183	119
4692	7	54	739	104
4693	1	95	838	32
4694	7	76	804	87
4695	2	12	846	22
4696	2	31	843	116
4697	10	49	220	27
4698	10	48	940	112
4699	6	1	740	45

4700	9	57	343	47
4701	6	30	512	41
4702	3	14	230	26
4703	8	28	845	91
4704	5	47	476	18
4705	7	78	335	40
4706	6	34	487	103
4707	2	32	571	103
4708	2	4	801	64
4709	4	40	432	76
4710	6	79	801	104
4711	1	98	806	106
4712	3	13	348	54
4713	9	41	746	91
4714	5	38	468	54
4715	3	29	763	71
4716	5	37	394	68
4717	1	26	277	25
4718	8	75	370	103
4719	1	47	329	53
4720	3	63	915	60
4721	5	16	604	58
4722	10	79	165	66
4723	10	61	513	112
4724	8	27	204	34
4725	5	64	769	109
4726	7	53	998	38
4727	5	17	951	62
4728	8	15	758	120
4729	10	33	543	38
4730	5	3	895	39
4731	8	62	663	74
4732	7	77	314	21
4733	8	25	646	26
4734	6	85	179	75
4735	6	81	171	94
4736	3	97	904	70
4737	5	65	769	47
4738	4	8	693	12
4739	5	24	969	111
4740	9	42	347	56
4741	2	2	972	90
4742	5	41	866	85
4743	8	22	458	77
4744	2	48	749	74
4745	7	67	529	71
4746	6	79	990	25

4747	4	44	443	12
4748	1	50	978	72
4749	10	82	617	77
4750	2	81	362	31
4751	3	9	939	63
4752	7	28	893	53
4753	6	45	672	14
4754	2	45	656	43
4755	7	45	793	71
4756	8	85	439	37
4757	5	79	783	58
4758	7	29	564	94
4759	3	69	674	62
4760	2	14	978	64
4761	9	72	776	103
4762	1	38	346	68
4763	9	21	177	109
4764	8	24	488	35
4765	6	56	704	47
4766	4	39	458	104
4767	8	86	751	115
4768	7	81	180	86
4769	10	93	219	98
4770	10	51	153	88
4771	7	14	245	44
4772	3	74	581	89
4773	9	96	672	89
4774	9	48	629	18
4775	7	40	976	49
4776	3	58	365	94
4777	10	89	622	49
4778	7	65	765	111
4779	3	87	970	40
4780	3	30	651	50
4781	6	67	504	43
4782	8	86	220	43
4783	10	94	483	12
4784	2	13	840	100
4785	8	77	332	88
4786	10	52	481	28
4787	8	48	243	116
4788	2	57	596	107
4789	1	48	237	35
4790	1	58	800	40
4791	8	41	579	41
4792	9	96	872	32
4793	2	29	476	113

4794	4	61	834	113
4795	1	48	817	101
4796	10	86	715	45
4797	6	65	498	40
4798	7	36	993	112
4799	6	29	474	72
4800	1	30	604	15
4801	1	14	507	42
4802	7	46	415	39
4803	10	91	769	40
4804	8	58	562	111
4805	7	24	866	35
4806	4	14	554	38
4807	4	23	336	82
4808	5	10	385	68
4809	3	7	223	84
4810	10	39	711	58
4811	10	55	654	66
4812	2	66	445	79
4813	2	21	456	54
4814	6	64	304	107
4815	2	80	633	24
4816	3	82	197	76
4817	3	7	621	24
4818	5	91	990	54
4819	10	32	978	70
4820	2	56	491	37
4821	7	32	741	32
4822	10	71	482	34
4823	1	18	301	103
4824	9	8	897	66
4825	3	25	182	71
4826	2	30	952	82
4827	2	7	217	30
4828	4	32	659	27
4829	3	20	398	108
4830	1	46	885	35
4831	3	46	547	75
4832	7	57	461	100
4833	8	66	186	64
4834	10	84	930	110
4835	1	47	153	45
4836	6	83	864	86
4837	2	35	894	61
4838	10	28	278	103
4839	7	58	682	103
4840	7	11	536	21

4841	6	47	362	15
4842	10	54	497	42
4843	7	91	700	33
4844	2	96	356	31
4845	6	69	696	77
4846	6	32	157	15
4847	2	71	434	89
4848	2	77	344	39
4849	6	18	160	116
4850	7	71	583	31
4851	4	79	359	115
4852	10	20	606	74
4853	8	71	216	101
4854	6	56	963	56
4855	10	89	246	108
4856	10	98	900	65
4857	5	32	567	95
4858	2	13	595	88
4859	4	69	606	19
4860	8	89	649	80
4861	3	72	483	86
4862	6	34	961	72
4863	5	78	401	103
4864	6	58	826	103
4865	9	68	495	20
4866	1	60	748	98
4867	8	84	605	38
4868	2	35	175	107
4869	9	68	657	29
4870	1	20	319	111
4871	7	63	679	17
4872	9	72	784	51
4873	8	7	503	18
4874	5	5	243	118
4875	8	8	959	21
4876	2	75	194	17
4877	2	30	307	29
4878	8	11	192	44
4879	6	77	550	62
4880	9	96	352	117
4881	4	34	252	114
4882	5	67	661	80
4883	5	93	342	25
4884	2	78	386	45
4885	8	8	539	23
4886	1	29	221	26
4887	3	50	739	113

4888	4	35	966	107
4889	2	32	736	86
4890	6	66	960	17
4891	10	17	306	110
4892	1	91	825	106
4893	8	12	539	83
4894	9	61	376	29
4895	4	32	234	110
4896	5	32	463	81
4897	5	5	252	100
4898	6	15	223	120
4899	1	48	428	30
4900	8	48	398	63
4901	2	21	609	77
4902	7	65	367	108
4903	1	8	614	32
4904	10	93	484	107
4905	7	35	245	49
4906	6	48	522	109
4907	6	57	777	34
4908	9	67	433	54
4909	3	5	628	106
4910	2	4	854	76
4911	2	39	715	106
4912	7	46	379	74
4913	8	72	634	99
4914	5	87	530	93
4915	8	5	677	45
4916	9	53	790	11
4917	9	98	746	75
4918	8	23	512	14
4919	7	56	327	39
4920	6	14	571	12
4921	2	6	778	58
4922	8	15	294	73
4923	5	80	537	48
4924	4	92	296	19
4925	9	18	909	48
4926	8	63	861	84
4927	7	21	167	25
4928	6	33	926	51
4929	6	11	961	60
4930	10	62	159	61
4931	6	19	821	59
4932	8	41	697	81
4933	6	78	917	74
4934	5	88	993	81

4935	8	28	693	112
4936	5	85	931	103
4937	1	63	165	70
4938	3	74	658	51
4939	6	12	696	108
4940	4	39	512	67
4941	3	55	158	11
4942	10	36	910	38
4943	8	86	585	28
4944	5	9	823	72
4945	1	13	325	48
4946	6	33	795	61
4947	10	81	611	12
4948	7	79	890	54
4949	9	68	432	45
4950	1	2	718	27
4951	4	29	875	116
4952	6	45	277	115
4953	3	77	914	29
4954	2	99	697	10
4955	3	65	312	47
4956	1	20	530	82
4957	3	86	796	101
4958	7	7	773	22
4959	8	42	240	86
4960	2	64	607	102
4961	3	74	856	95
4962	1	24	963	15
4963	10	56	779	87
4964	7	12	904	84
4965	1	78	602	54
4966	8	99	241	58
4967	10	73	191	91
4968	2	1	258	23
4969	10	75	976	44
4970	5	53	425	79
4971	8	90	752	86
4972	5	96	532	107
4973	7	23	501	57
4974	1	74	858	11
4975	8	54	585	52
4976	4	24	620	19
4977	10	26	177	49
4978	7	86	811	73
4979	5	61	587	11
4980	7	73	423	51
4981	10	28	596	109

4982	6	75	597	116
4983	4	57	769	105
4984	4	67	889	51
4985	9	54	647	20
4986	9	7	469	13
4987	6	27	691	22
4988	5	80	262	27
4989	8	82	809	87
4990	10	15	196	110
4991	10	51	412	62
4992	6	61	687	69
4993	6	28	857	78
4994	10	72	941	70
4995	7	8	491	107
4996	10	72	661	14
4997	4	27	944	115
4998	1	94	433	111
4999	9	51	856	103
5000	2	86	237	32