American University in Cairo

AUC Knowledge Fountain

Theses and Dissertations

Student Research

Winter 1-31-2022

Final Bid Price Estimation for Negotiated Contracts: Bargaining Game Theory Approach

Amr El Mahallaoui
amrmahallawy@aucegypt.edu

Follow this and additional works at: https://fount.aucegypt.edu/etds

Part of the Construction Engineering and Management Commons

Recommended Citation

APA Citation

El Mahallaoui, A. (2022).*Final Bid Price Estimation for Negotiated Contracts: Bargaining Game Theory Approach* [Master's Thesis, the American University in Cairo]. AUC Knowledge Fountain.
https://fount.aucegypt.edu/etds/1890

MLA Citation

El Mahallaoui, Amr. *Final Bid Price Estimation for Negotiated Contracts: Bargaining Game Theory Approach*. 2022. American University in Cairo, Master's Thesis. *AUC Knowledge Fountain*.
https://fount.aucegypt.edu/etds/1890

This Master's Thesis is brought to you for free and open access by the Student Research at AUC Knowledge Fountain. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of AUC Knowledge Fountain. For more information, please contact thesisadmin@aucegypt.edu.

The American University in Cairo

School of Sciences and Engineering

FINAL BID PRICE ESTIMATION FOR NEGOTIATED CONTRACTS: BARGAINING GAME THEORY APPROACH

A Thesis Submitted to the Department of Construction Engineering

In Partial Fulfillment of the Requirements for the Degree of

Master of Science in Construction Engineering

By

Amr Nagui El Mahallaoui

Under the Supervision of

Dr. Ahmed Samer Ezeldin

Professor, Construction Engineering
Department

The American University in Cairo

Dr. Elkhayam Dorra

Adjunct Assistant Professor,
Construction Engineering
Department

The American University in Cairo

Fall 2021

DEDICATION

This work is dedicated to my family. I am grateful to my mother, Dr. Alyaa Ragaei, for her endless love, support and encouragement. I will always appreciate all she has ever done for me. I am also thankful to my father Dr. Nagui, my sister Nouray and my brother Mohamed for their endless love and support.

This thesis is also dedicated to my friends who have supported me throughout the process. I am grateful to them for always being there for me.

This work is dedicated to the memory of my friend Dalia ElShafei who has always been a source of inspiration.

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude to my advisors, Dr. Samer Ezeldin and Dr. Elkhayam Dorra, for their continuous support and guidance. Without their guidance and patience, this thesis would not have been possible.

I would like also to express my sincere appreciation and thanks to Eng. Omneya Mostafa (Dar Al Handasah) for all the things I have learnt from her and for her continuous support and motivation.

Special thanks go to Eng. Mostafa Hesham for his great support throughout the course of this research.

I would like as well to acknowledge and thank Eng. Ahmed Hegab (Dar Al Handasah) and Eng. Ayman El Hakea (CRC Dorra) for their great help and support throughout this research.

Finally, many thanks go to the experts who participated in the brainstorming session and in the survey.

ABSTRACT

The wide use of the low bid method by employers for awarding construction contracts has created an aggressively competitive environment among contractors in the construction industry. As a result, a contractor may resort to use a low mark-up percentage for his bid to increase his chances of winning, which may lead to losses and conflicts in case he is awarded the project. Additionally, a contractor who reaches the final negotiations stage for a certain project is faced by the dilemma of the minimum discount percentage he may need to offer to the employer that maximizes his chance to win the project. This research presents the framework for a decision support tool / model that uses bargaining game theory to help contractors make rational decisions regarding the discount percentage to offer to the employer during negotiations in order to establish a win-win scenario in which the employer gets the lowest possible price for his project and at the same time the contractor's profit is maximized. The developed Monte Carlo simulation based python model uses the source code of Gambit in order to determine the Nash Equilibrium of a typical negotiation process in private sector projects; where negotiations are allowed, contractors are procured through competitive bidding, and the low bid method is used for awarding the contracts. The negotiation process is depicted by a game composed of three players (two contractors and one employer), through a two stage negotiation process. Moreover, a real case study of a hospital mega project in Egypt is used to validate the developed python model. The analysis of this case study showed that using the developed model by the winning contractor could have saved him almost 299.5 M EGP of unnecessary discount offered to the employer. Additionally, another objective of this research is to determine and rank the factors that affect the level of aggression (bargaining power) of the two negotiating parties (employer/contractor) in the Egyptian market.

Table of Contents

ABSTRACT 3

List of Figures 8

List of Tables 9

List of Equations 9

Chapter 1: Introduction 10

- **1.1 Background:** 10
- **1.2 Problem Statement:** 12
- **1.3 Research Objectives:** 14
- **1.4 Research Methodology:** 15
- **1. 5 Research Organization:** 16

Chapter 2: Literature Review 18

- **2.1: Background** 18
  - 2.1.1: History of Game Theory: 18
  - 2.1.2: Nash Equilibrium: 19
  - 2.1.3: Important Definitions & Concepts: 21
  - 2.1.4: Bargaining Theory: 23
- **2.2: Game Theory Applications in the Construction Industry:** 24
  - 2.2.1: Joint Resource Management: 24
  - 2.2.2: Subcontractor Selection: 25
  - 2.2.3: ADR approaches: 25
  - 2.2.4: Construction Claims: 26
  - 2.2.5: Profit Sharing for Joint Ventures: 27
  - 2.2.6: Bid Mark-Up Estimation: 27

2.3: Bargaining Theory Applications in the Construction Industry: 29

2.3.1: Public – Private – Partnership (PPP) Projects: 30

2.4: Bargaining Theory Applications in Other Fields: 32

2.5: Research Gap: 33

Chapter 3: Research Methodology 34

3.1 Introduction: 34

3.2 Brainstorming Session: 35

3.2.1 Introduction: 35

3.2.2 Experts Selection: 36

3.2.3 Brainstorming Session Conduction: 36

3.2.4 Brainstorming Session Demography: 37

3.3 Surveying Experts: Ranking of Factors Affecting the Bargaining Power 38

3.3.1 Introduction: 38

3.3.2 Survey Architecture: 39

3.3.3 Survey Sample Selection: 40

3.3.4 Sample Size: 41

3.3.5 Survey Conduction: 42

3.3.6 Survey Execution: 43

3.3.7 Survey Demography: 43

3.4 Designing the Negotiations Game: Gambit 45

3.4.1 Introduction: 45

3.4.2 Description of the Negotiations Game: 46

3.4.2.1 Introduction: 46

3.4.2.2 Players Strategies: 47

3.4.2.3 Players Payoffs: 49

3.4.3 Modeling the Game on Gambit: 50
3.5 Developing the Decision Support Tool: Python Model 52
3.5.1 Introduction: 52
3.5.2 Model Processes: 54
3.5.2.1 Input Module: 55
3.5.2.2 Processing Module: 56
3.5.2.3 Output Module: 60
3.5.3 Python Model Advantages over Gambit: 62
Chapter 4: Results, Verification and Validation 64
4.1 Introduction: 64
4.2 Survey Results: 64
4.3 Model Verification: 67
4.3.1 Game 1 on Gambit: 68
4.3.2 Game 2 on Gambit: 69
4.3.3 Python Model Output for the same Scenario: 71
4.4 Case Study – Hospital Project in Cairo: 72
4.4.1 Introduction: 72
4.4.2 Bidding Behavior – Contractor 1: 73
4.4.3 Bidding Behavior – Contractor 2: 75
4.4.4 Project Direct Cost & Owner’s Value: 76
4.4.5 Predictions of the Python Model: 77
4.4.6 Comparing the Predictions of the Python Model to the Actual Scenario: 79
Chapter 5: Conclusion and Recommendations 82
5.1 Research Overview: 82
5.2 Research Contribution: 84

5.3 Encountered Challenges: 85
5.4 Recommendations for Future Research: 86
References: 88
Appendices 92
Appendix A: Brainstorming Session Experts: 93
Appendix B: List of Factors Affecting the Bargaining Power of the Negotiating Parties: 94
Appendix C: Sample Questionnaire: 95
Appendix D: VS Code: 105
Appendix E: Projects Data Provided by Contractors: 113

List of Figures

Figure 1: Prisoner's Dilemma Extensive Form Representation 21
Figure 2: Research Methodology 34
Figure 3: Experts Years of Experience - Brainstorming Session 37
Figure 4: Experts Company Sector - Brainstorming Session 37
Figure 5: Experts Positions - Brainstorming Session 38
Figure 6: Survey Participants - Company Sector 44
Figure 7: Survey Participants - Positions 44
Figure 8: Survey Participants - Years of Work Experience 45
Figure 9: Illustration Diagram for Games 1 & 2 47
Figure 10: Sample Game Modeled on Gambit in Extensive Form 51
Figure 11: Python Model Processes 54
Figure 12: Data Input - Json File 55
Figure 13: Data Input - Tkinter 56
Figure 14: Python Model - Bid Prices Calculation 57
Figure 15: Payoffs Calculation 57
Figure 16: Python Model - Nash Equilibrium Computation 58
Figure 17: Python Model - Determination of Winning Contractor 59
Figure 18: Python Model - Storing of Winning Strategies 60
Figure 19: Results Output - Terminal 61
Figure 20: Results Output - Tkinter 61
Figure 21: Game 1 Modeled on Gambit – Model Verification 69
Figure 22: Game 2 Modeled on Gambit – Model Verification 70
Figure 23: Python Model Predictions – Model Verification 71
Figure 24: Normal Q-Q Plot - Contractor 1 74
Figure 25: Normal Q-Q Plot - Contractor 2 76
Figure 26: Input Data - Validation Case Study 77
Figure 27: Predicted Outcomes - Validation Case Study 78

List of Tables

Table 1: Prisoner's Dilemma Strategic Form Representation 20
Table 2: Ratings and Raking of the Factors Affecting the Bargainning Power of the Negotiating Parties.. 64
Table 3: Players Data – Model Verification 68
Table 4: Extraction of Contractor 1 Projects Data 73
Table 5: Extraction of Contractor 2 Projects Data 75
Table 6: Actual Figures - Validation Case Study 80

List of Equations

Equation 1: Cochran's formula 42
Equation 2: Employer's Payoff.. 49
Equation 3: Employer's Payoff in case of Rejection of the Contractro's Offer 49
Equation 4: Owner's value @ game 2 50
Equation 5: Contractor's Payoff 50

Chapter 1: Introduction

1.1 Background:

The Construction Industry is one of the important sectors of the Egyptian economy. According to the American Chamber of Commerce in Egypt, the construction & real estate state sector accounted for almost 15% of the national GDP as per their estimate for the first quarter of fiscal year 2019/2020 (American Chamber of Commerce in Egypt, 2020). And with the high population growth in Egypt and the high urbanization rate, the construction industry will keep growing, especially that the high need for low and middle income housing is yet to be addressed. In fact, in the past few years, the Egyptian government had launched several large- scale construction projects, including the construction of new cities, roads, bridges, factories and power plants. One of the important projects is the construction of the New Administrative Capital (on a total area of 700 Kms^2), which will become the new administrative and financial capital of Egypt, housing the main government departments and ministries, as well as foreign embassies (State Information Service, 2020). Another important large-scale project launched by the government is the New Alamein city which is built across 50,000 feddans and with an estimated capacity of over 3 million residents (Arab Contractors, 2020). Thus, with the current strategy of the Egyptian government, the construction industry is expected to grow more in the coming period, especially that these new projects launched by the government attract real estate developers to the surrounding areas.

These mega projects launched by the Egyptian Government pass through several phases during their life cycles. One of the important phases of these projects is the tendering stage. In fact, in any construction project, the contractor that will construct the works is usually procured through a tendering process. In a tender, the contractor offers a price to the project owner for the supply of goods and / or services required. The owner of the project may allow anyone to submit a tender for his project (open tendering) or may send an invitation to tender to certain contractors only (selective tendering). Depending on the nature of the project, the owner selects one of the two previously mentioned techniques: if the project is not of a complex nature, the owner may select the open tendering to ensure a high competition between the contractors who will bid for the project in order to get the lowest price possible. Also in the open tendering the project owner may apply a pre-qualification process to produce a shortlist of suitable contractors who will be invited to submit tenders. On the other hand, if the nature of the project is complex, the project owner may resort to selective tendering and in this case, a pre-selected list of suitable contractors having the required experience for the project are invited to submit tenders. Whether the project owner resorts to open tendering or selective tendering, the contractors submit their tenders, which will be evaluated afterwards financially and technically by the project owner. The technical evaluation's aim is to evaluate to what extent the contractor understands the requirements of the project and whether he has the necessary resources (staff, labor, equipment and financial) to undertake the project. The financial evaluation's aim is to evaluate the financial offer submitted by the contractor and to detect items left unpriced and the arithmetic errors. It also detects whether the bid involves a front loading case by the contractor (when elements of the work taking place early on during the project are attributed higher prices than normal) or whether the bid is an unbalanced bid (when the contractor places a high price on some items and a low price on other items in a re-measured contract). Based on the technical and financial evaluation of the submitted tenders, the project owner chooses the best contractor to undertake the project.

1.2 Problem Statement:

The main challenge for a contractor when preparing the bid for a low/average complex project, is the financial part. In fact, in most cases the financial offer has a critical role in awarding the project. The project owner will not award the project to an unqualified contractor (from the technical point of view). However, as the low bid method is the most common method used for awarding construction contracts, the project owner will award the project to the contractor with the best financial offer from the technically qualified bidders (Ahmed et.al, 2016). Thus, the contractor's financial offer should be one of the lowest offers submitted to be able to win the project.

In fact, the use of the aforementioned competitive bidding technique for contractor selection has created an aggressive competitive environment among contractors in the construction industry (Ahmed et.al, 2016). Due to this competitive environment, the contractor may resort to use a low mark-up percentage for his bid (in order to increase his chances of winning the project), which may lead him to win the bid, but it may also lead him to make low profits or even losses as he may underestimate the project's true cost (Ahmed et.al, 2016). And in order to recover their low profits/losses, contractors may rely on their negotiations power while negotiating with the employer (after being awarded the project) on the price of variation orders (Dyer and Kagel, 1996) and on claims (Rooke et.al, 2010). These aforementioned strategies used by contractors to recover their losses/make profit leads certainly to conflicts with employers. Also, these strategies do not guarantee to contractors to always recover their losses.

Consequently, a contractor needs to make sure that his estimated markup will not lead to low profits nor to losses in case he is awarded the project (avoiding submitting an underestimated bid). But, this is not the only problem a contractor faces when preparing the financial offer. In fact, for negotiated contracts, there will be a negotiation stage between the project owner and the contractors with the lowest financial offers. The project owner will try during these negotiations to make the contractors lower their prices in order to get the best price possible for his project (Dzeng and Lin, 2004). Thus, the contractor knows that he cannot put his real final price in the initial financial offer because he knows that there will be a negotiation stage with the project owner and he will have to lower his price in order to get the project (specially that the project owner will be negotiating with the contractor's competitors at the same time). Therefore, the problem faced by the contractor is that he needs to know the negotiation margin to win the project and at the same time to ensure that he will not make low profits nor losses in case he is awarded the project.

Thus, in order to avoid the abovementioned conflicts and loses resulting from the competitive environment, contractors need to approach things differently and to use new techniques for their bid prices estimation. In fact, the negotiations stage between the project owner and the contractor can be seen as a game: a strategic interaction between two rational players (the project owner and the contractor). The two players in this situation want to cooperate but they do not know exactly how to cooperate. The project owner wants to give the project to the contractor at the lowest possible price and at the same time the contractor wants to take the project and decrease his original bid price minimally. During the negotiations process, each of these two players will use a strategy which (in his opinion) will lead him to the best possible outcome. The final outcome of this negotiation (the final agreed contract price) depends on the strategy choices of the two players.

Game Theory can be used to analyze and explain this strategic interaction between the project owner and the contractor, and predict its rational outcome. Game Theory can be defined as a "theory of rational decision in conflict situations". It is the "logical analysis of situations of conflict and cooperation" (Bhuiyan, 2016). Game theory was originally used in economics to understand the behaviors of "firms, markets and consumers"(Bhuiyan, 2016). It is also used in other domains such as: applied mathematics, social sciences, biology, political science and engineering. The branch of Game Theory that explains and analyzes negotiations is called Bargaining Theory. The history of Bargaining / Game Theory as well as its applications in the construction industry are addressed in details in Chapter 2.

1.3 Research Objectives:

The main objective of this research is to use Bargaining Game Theory to find a solution to an important problem faced by the contractor: determining the negotiation margin. As discussed earlier in section 1.2, the contractor needs to guarantee that the final contract price (after negotiations) will not lead him to make low profits nor losses in case he is awarded the project.

A decision support tool / model using Bargaining Game Theory will be developed to predict the strategy the contractor should follow in the negotiation stage to increase his chances of winning the project and at the same time to maximize his profit as possible.

Furthermore, all the research done using bargaining theory (in the construction field) to date, address issues related mainly to public sector projects like the PPP projects. Thus, this research aims to extend the use of bargaining theory to cover other types of projects as well.

Additionally, another objective of this research is to determine and rank the factors that affect the level of aggression (bargaining power) of the two negotiating parties (employer/contractor).

Thus, this research aims at providing contractors in the Egyptian market with a decision support tool to determine rationally their final bid prices considering that the contractors are bidding for private projects (where negotiations are allowed), procured through competitive bidding and the low bid method is used for awarding the contracts (i.e. the contractor’s price is the only factor considered by the employer in awarding the project).

1.4 Research Methodology:

In order to develop a model / decision support tool that predicts the percentage of the bid price the contractor should lower, the following approach will be adopted:

1. Determining the factors that affect the level of aggression (bargaining power) of the two negotiating parties (employer/contractor). As the Egyptian market is very dynamic and as it keeps changing very fast specially after the devaluation of the Egyptian pound in 2016 and with the current mega projects launched by the government, these factors will be determined by conducting a brainstorming session with a group of experts in the Egyptian construction field. Using this technique, an up-to-date assessment of the Egyptian market could be obtained.

2. Based on the list of factors determined from the brainstorming session with the group of experts, surveys will be distributed to experts in the field of construction to rate each of these factors. And based on the obtained ratings, the factors affecting the bargaining power of the parties will be ranked.

3. After ranking the aforementioned factors, “Gambit” software will be used to design the negotiation game between the contractors and the employer. Gambit is an open-source collection of tools for conducting computations in game theory and is highly recommended by researchers.

4. After designing the negotiation game, a python model will be developed using the source code of Gambit. This model can be used in determining the strategy the contractor should follow in the negotiation process to get out with the maximum profit possible.
5. Finally, the developed python model will be validated through a real case study.

1. 5 Research Organization:

This research is organized into five chapters as follows:

Chapter 1 – Introduction:

This chapter includes a background section explaining the importance of the construction industry in the Egyptian market followed by a problem statement section explaining the problem this research aims to solve. It includes also the research objectives as well as a summary of the research methodology.

Chapter 2 – Literature Review:

This chapter includes a background (history and important concepts) of Game Theory and its branch Bargaining Theory. It includes also a summary of the research efforts of using Game Theory in different applications in the construction industry as well as the applications of Bargaining Theory.

Chapter 3 – Research Methodology:

This chapter includes the proposed research methodology and approaches to be used to tackle the research objectives and scope.

Chapter 4 – Results, Verification and Validation:

This chapter presents the research findings as well as an analysis of the model that predicts the percentage of the bid price the contractor should lower during negotiation. It also presents the verification and validation of this model.

Chapter 5 – Conclusion and Recommendations:

This chapter includes a summary of the research and its main contributions. It includes also the research limitations and the recommendations for future research.

Chapter 2: Literature Review

2.1: Background

2.1.1: History of Game Theory:

Before discussing the applications of game theory in the construction industry, it is important to take a look at its history. The first documented discussion of game theory occurred in 1713 when Francis Waldegrave wrote a letter to Pierre-Remond de Montmort in which he provided a minimax mixed strategy equilibrium to a two-person version of the card game le Her. The solution for this game is known as "minimax" because each player tries to minimize the maximum payoff for his opponent and at the same time, he tries to minimize his own maximum loss (Hurtado, 2015). In 1871, Charles Darwin gave the first game theoretic argument in evolutionary biology in his book "The Descent of Man, and Selection in Relation to Sex". Darwin's argument was that if the births of females are less common than males, then females are expected to have more offspring. Thus parents that produce females tend to have more than the average numbers of grandchildren, thus, the female births become more common. Also, this advantage associated with producing females disappears when the 1:1 sex ratio is approached (Hurtado, 2015). In 1913, Ernst Zermelo published the first theorem of game theory which states that chess is strictly determined and that it has only "one individually rational payoff profile in pure strategies". This theorem is known as "Zermelo's Theorem". In the period 1921-1927, Emile Borel published four notes on strategic games and gave the first modern formulation for "finding the minimax solution for two-person games with three or five possible strategies". Borel claimed also that games with more possible strategies would not have minimax solutions. Despite the contribution of the previously mentioned researchers in the history of Game Theory, it did not exist by this name before John Von Neumann's work in 1928. In 1928, Von Neumann proved that "every two- person zero-sum game with finitely many pure strategies for each player is determined" and that this game has precisely "one individually rational payoff vector". For such game, an equilibrium can always be found from which "neither player should deviate unilaterally" (Walker, 1995) and (Najera, 2019).

Moreover, Von Neumann & Oskar Morgenstern published "Theory of Games and Economic Behavior" in 1944 which established Game Theory as an interdisciplinary research field and opened the door for social sciences to develop mathematical tools to describe human behaviors (Ahmed et.al, 2016). After the publication of this book, Game Theory was applied into different aspects of human life: in the 1950s and 1960s, the United States and the Soviet Union started to use it during the Cold War to take rational political decisions. In the 1970s, it was used in several applications in economy. It was also used in the biology, psychology and sociology domains and even in several domains in the construction industry (Walker, 1995) and (Najera, 2019).

2.1.2: Nash Equilibrium:

Furthermore, John Nash, a research mathematician at Princeton University made fundamental contributions to Game Theory. In 1994, Nash shared the Nobel Memorial Prize in Economic Sciences with game theorists Reinhard Selten and John Harsanyi. In 2015, he also shared the Abel Prize with Louis Nirenberg and became the only person in history to be awarded both the Nobel Memorial Prize and the Abel Prize (Kelly, 2015).

The period 1950-1953 witnessed Nash's revolutionary contribution to Game Theory. Nash (1950) and Nash (1951) proved that there exists a strategic equilibrium for non-cooperative games -Nash Equilibrium- and that this solution is applicable to N-person games and not simply to two-person games (broader solution than the one proposed by Von Neumann in 1928). The Nash Equilibrium for a game is the optimal outcome that no player would have an incentive to deviate from the strategy leading to this outcome considering his opponent's choices. The Nash Equilibrium is the solution that would leave the player better-off regardless of what his opponents do.

A very famous example to illustrate the Nash Equilibrium is the "Prisoner's Dilemma". In this game, two criminals are arrested for committing a robbery. The police do not have enough evidence to convict them, so the two criminals are separated (with no mean of communication with each other) and the police offer each of them the opportunity to cooperate and admit that the other prisoner committed the crime in order to get a reduced sentence (passing less time in prison). If both prisoners betray each other, each of them is sentenced for 8 years. If one prisoner betrays the other and the other remains silent, then the one who remained silent will be sentenced for 10 years and the other one will be set free. If both remain silent, they will be sentenced for 1 year only. The following figure shows the possible outcomes for the 2 prisoners in this game:

Table 1: Prisoner's Dilemma Strategic Form Representation

		Prisoner B	
		Confess	Remains Silent
Prisoner A	Confess	**(-8,-8)**	**(-10,0)**
	Remains Silent	**(0,-10)**	**(-1,-1)**

So, the best outcome for the prisoners is that they both remain silent to get the least sentence. But, as they are separated, each of them does not trust the other to remain silent. Moreover, by remaining silent the prisoner can end up receiving the 10 years maximum prison sentence possible, which is the worst case scenario. The Nash Equilibrium for this game is for both players to betray each other. Because if the prisoner decides to betray the other one, he will be left better-off regardless of what the other criminal decides to do. In the worst-case scenario, he will end up with 8 years in prison only (if the other criminal betrays him). However, if he decides to remain silent, he may end up with 10 years in prison (Faghih and Akhavian, 2019). Figure1 shows the extensive form representation of the Prisoner's Dilemma game:

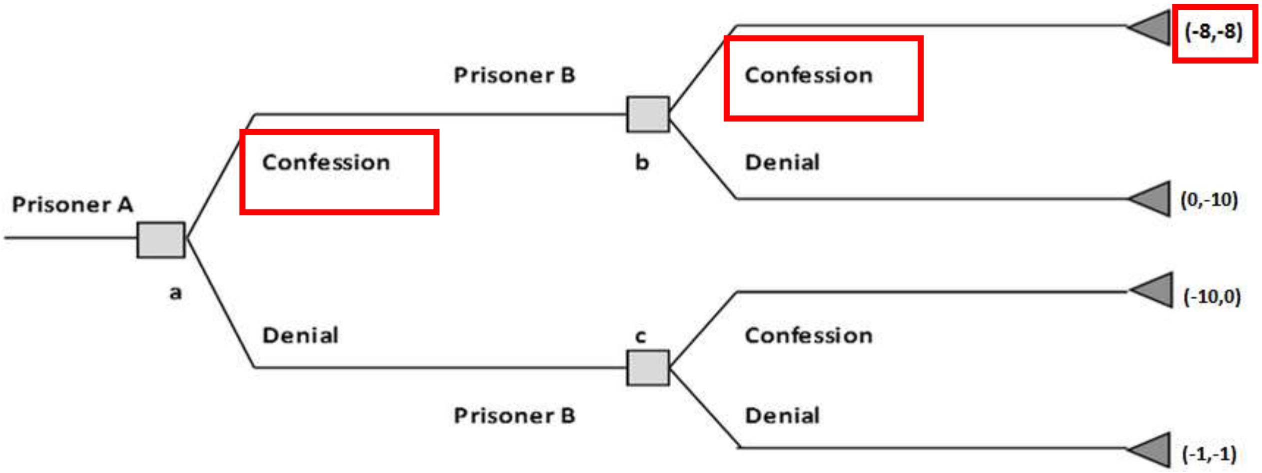


Figure 1: Prisoner's Dilemma Extensive Form Representation

Thus, determining the Nash Equilibrium for the prisoner's dilemma or any non-cooperative game, allows the player to determine the best strategy to follow to reach the best outcome regardless of his opponent's choices (given that the players are rational).

2.1.3: Important Definitions & Concepts:

Additionally, there are several definitions and concepts in game theory that must be explained, before discussing the applications of Game Theory. The following terms are commonly used in the study of game theory (Rasmusen, 2006):

- Game: is a formal description of a strategic situation that has a result dependent on the choices of the decision-makers / players.
- Players: are the decision-makers in a game.
- Strategy: is the set of choices a player will take given the circumstances of the game.
- Payoff: is a number that shows to what extent a player desires a certain outcome. Payoffs may represent profit, quantity, utility, or any continuous measures.
- Information Set: is the available information at a given time in the game.
- Zero-Sum Game: is a game where the sum of the payoffs to all players is zero. One player's gain = the other player's loss.
- Rationality: A player is rational if he plays in a manner that maximizes his profits (payoff). In Game Theory, it is assumed that all players are rational.
- Dominant Strategy: is the strategy which gives the player a better payoff regardless of what the other players decide.
- Cooperative and Non-Cooperative games: Cooperative games are games in which players cooperate to get more benefits (higher payoff) and distribute those benefits in a fair manner between them. On the other hand, non-cooperative games are games in which each player selects his strategy independently from all other players and tries to maximize his own benefits (payoff). There is no contribution between the players.
- Strategic and extensive form games: Strategic form is the basic type of game in non-cooperative game theory. In strategic form, the strategies of each player are listed and the outcomes that may result from the combination of the possible choices are determined. Each outcome is represented by a separate payoff for each player which measures to what extent each player will be satisfied with this outcome. The strategic form is usually

represented by a matrix which shows the players, strategies, and payoffs. Table 1 is an example of a strategic form game. On the other hand, extensive form games "game tree" is more detailed than the strategic form. It shows the order in which the players take their decisions and the information available to them at each stage. Figure 1 is an example of a strategic form game. The extensive form game can be analyzed directly or can be converted into an equivalent strategic form game (Turocy, 2001).

- Static and dynamic games: In static games, players make their decisions and take actions simultaneously without knowing the decisions of the other players. In dynamic games, players make decisions and take actions sequentially with the observation of the other players' decisions (Ahmed et.al, 2016).

2.1.4: Bargaining Theory:

Moreover, Nash (1950) and Nash (1953) founded the bargaining theory (a branch of game theory) and proved the existing of the "Nash Bargaining Solution" for cooperative games. In a bargaining situation, the two players want to cooperate but they are not sure how they should cooperate exactly; in other words, each player wants to cooperate with the other player and at the same time he wants to get the highest payoff possible. To tackle this problem and to reach the "Nash Bargaining Solution" for this game, Nash (1950) and Nash (1953) proposed the "Nash Program", which is a research agenda for studying cooperative games by transferring them to non-cooperative form. This is achieved by exploring non-cooperative procedures that yield cooperative solutions as their equilibrium outcomes" (Serrano, 2004). Nash (1953) proposed two methods to transfer cooperative games into non-cooperative games. The first is to "make the player's steps in negotiations in the cooperative game become moves in the non-cooperative model". To be able to perform that, the "negotiation process must be formalized and restricted, but in such a way that each participant is still able to utilize all the essential strength of his position”. The second method proposed by Nash (1953) is to use the axiomatic method, which is a procedure by which an entire system is developed by logical deduction from specified basic rules (axioms), which are constructed from primitive terms. According to Nash (1953), several properties (that would seem natural for the solution to have) should be stated as axioms, and the axioms would “determine the solution uniquely”.

2.2: Game Theory Applications in the Construction Industry:

In fact, the Construction industry experiences numerous instances of cooperative and non-cooperative environments and bargaining that lends itself ideally to Game Theory. Accordingly, researchers have applied various game theoretic concepts to develop models explaining and predicting outcomes for several issues in the construction industry.

2.2.1: Joint Resource Management:

Game Theory was used to create a framework for joint resource management in construction projects. Asgari et.al (2014) examined the opportunities for sub-contractors to make considerable benefits from joint resource management in large construction projects involving several sub-contractors. A case was presented in which the subcontractors agree to put some of their resources in a “joint pool” for a certain duration and to allocate those resources effectively among them. This situation is considered as a cooperative game as the sub-contractors share some of their resources among themselves to make more profits. Asgari et.al (2014) used game theory to determine a “fair and efficient allocation of the incremental benefits of cooperation among the cooperating sub-contractors”. It was concluded that using game theory methods to “design fair, efficient, and stable schemes” for allocating cooperative gains among the sub-contractors is extremely effective as it resulted in creating considerable savings for them (Asgari et.al, 2014). Thus, deciding to cooperate with the other sub-contractors and applying game theory methods is a rational decision for a sub-contractor working in a large project as it leads to the increase of profits.

2.2.2: Subcontractor Selection:

Another application for game theory in the construction industry is the examination of strategies for subcontractor selection. Unsal and Taylor (2011) developed an agent-based simulation model integrated with game theory to examine whether the contractors' strategies of not considering the subcontractor selection as a repeated game may lead to a holdup problem. It was concluded that the holdup problem can exist in project networks in case the subcontractor selection is not treated as a repeated game (Unsal and Taylor, 2011). The holdup problem is a situation where two parties can obtain more benefits by cooperating with each other but they refrain from doing so because one party fears that the other party may have a stronger bargaining power, thus it may reduce their profits. This situation can be considered as a cooperative game. Hence, using game theory to determine how the profits should be shared between the two parties will ensure that the holdup problem does not occur.

2.2.3: ADR approaches:

Furthermore, game theory models can be used as decision support tools in alternative dispute resolution (ADR) approaches (arbitration, conciliation, and mediation). For example, Faghih and Akhavian (2019) explored two game theory models (prisoner's dilemma and chicken game) and discussed the potential of using them as decision support tools in mediation cases. It was investigated how the two previously mentioned game theory models can be used to predict the conflict outcomes for the involved parties in the mediation and their impacts on the parties' interests in different scenarios. Faghih and Akhavian (2019) also presented two case studies on conflicts in the construction industry where the role of the mediator was to resolve the conflict based on the two game theory models. Finally, it was concluded that game theory is an extremely useful tool in helping the parties of a conflict examining the outcome of the different actions they may take to resolve the conflict (Faghih and Akhavian, 2019).

2.2.4: Construction Claims:

Game theory can also be used in the analysis of construction claims. Ho and Liu (2004) developed an analytical model "Claims Decision Model" using game theory as an analytical framework to study opportunistic bidding and construction claims. According to Ho and Liu (2004), opportunistic bidding is when "contractors bid low on a project and hope to recover the loss through negotiations or claims". The developed model explains how the different project parties behave during claiming situations, how the claiming situations may be related to the opportunistic bidding behavior and what situations encourage the opportunistic behavior to take place. It also allows to analyze the decisions and strategies of the different project parties and provides a method to analyze the claims in a systematic and rational way. Ho and Liu (2004) found that the Nash Equilibrium in a construction claim (in an opportunistic bidding condition) is for the parties to "negotiate and settle". The possible range of a "negotiation settlement" was also derived (Ho and Liu, 2004).

2.2.5: Profit Sharing for Joint Ventures:

Furthermore, researchers have used game theory to determine a profit-sharing scheme for joint ventures in the construction industry. With the globalization of the construction market, many construction companies decided to enter new markets and form joint ventures with the local construction firms. However, the existing conventional method of profit-sharing between the joint venture firms (based on the ratio of investment) is found to be doubtful and it doesn't guarantee that the parties will maintain a "stable cooperative relationship" as the parties make different contributions to the project. Hsueh et.al (2011) developed a contribution-based profit-sharing model for joint ventures based on game theory. The model allows a certain firm to determine whether it should participate in a joint venture for a certain project or not, which company would be the most appropriate joint venture partner and it also provides a "fair profit-sharing rule" for profits distribution among the parties of the joint venture that ensures an "acceptable profit-sharing solution to both parties" and a "stable cooperative relationship" between them. Hsueh et.al (2011) has also validated the model on a case study, and it was shown that the model ensures that all the joint venture parties will obtain "a profit more than their independent works".

2.2.6: Bid Mark-Up Estimation:

Another important area of the construction industry where researchers have applied game theory is the bid mark-up estimation. One of the most important challenges contractors face is the estimation of the mark-up percentage for their bids. A high mark-up percentage may lead to a bid price higher than the prices submitted by the other competitors, thus losing the bid. On the other hand, a low mark-up percentage may lead to winning the bid, but it may also lead to low profits or even losses as the contractor may underestimate the project's true cost. Thus, the contractors need to make sure that their estimated markup will not lead to low profits nor to losses in case they are awarded the project (avoiding an underestimated bid), and thus reducing the "winner's curse". The winner's curse occurs when the winning bidder earns negative or at least below normal profits because of submitting an underestimated bid (optimistic cost estimate) for a project (which is less than the project true construction cost) and is thus cursed by being selected to undertake the project.

Contractors need also to predict in an effective way what their competitors would select as a mark-up, to be able to win the bid. To tackle this problem, researchers have resorted to game theory. One of the tools developed using the game theory concept is the Symmetric Risk Neutral Nash Equilibrium (SRNNE) bid function, which is considered a tool for optimal strategic bidding developed by Dyer et al. (1989).

Ahmed et.al (2016) have tried to take the research undertaken by Dyer et.al (1989) to the next level and determined how learning from past bidding decisions and experiences in different construction bidding environments can help mitigating the winner's curse. Furthermore, Ahmed et.al (2016) developed simulation models for different bidding processes using the SRNEE bid function as an optimal bid function. The results of the simulation models (which were based on actual data set for real projects) were also analyzed and it was concluded that learning from past experiences give the contractors a higher chance to suffer less from the winner's curse in a multistage bidding environment compared to a single-stage bidding environment (Ahmed et.al, 2016). Moreover, Asgari et.al (2016) have proposed several solutions for the bid mark-up estimation, and optimal risk attitudes that contractors should take in competitive bidding environments were suggested based on an agent-based model of the construction bidding environment that was developed.

2.3: Bargaining Theory Applications in the Construction Industry:

Nevertheless, the use of game theory is not limited to the bid price estimation only. It can also be used during the tender negotiation phase between the employer and the contractor. In fact, the bid price submitted by a contractor in the tender phase, might not be the final contract price in the event of being awarded the project. The bidding contractor might have to undergo a negotiation phase with the client before determining his final contract award price. During the negotiation stage, the contractor is pressured by the client to lower his price, while at the same time the contractor wants to maximize his profit. Thus, the contractor needs a decision support tool which can help him deciding on the lowest bid price that will maximize his probability of winning the project while maintaining his interest of obtaining the maximum benefits possible during the negotiation phase. To tackle the dilemma of negotiations, researchers have resorted to the “Bargaining theory”. Bargaining theory is the branch of game theory dealing with the analysis of bargaining problems, in which some parties bargain over the division of certain goods. A solution to a bargaining problem means the determination of such a division. A bargaining situation can be defined as “a situation in which two players have a common interest to cooperate but have conflicting interests over exactly how to cooperate” (Muthoo, 1999). Thus, a negotiation between a client and a contractor on the final bid price for the project can be seen as a “bargaining situation”. Also, bargaining theory deals with the situations where people interact rationally with each other to get as much of the benefits as possible in the bargaining process. Thus, bargaining theory is applicable to the bid negotiations in construction projects as these are situations where employers and contractors negotiate in a rational way: the employer wants to pay the lowest cost possible for his project and the contractor wants to get the project and makes the highest possible benefits.

The majority of research done related to the use of bargaining theory (in the construction field) to date, address issues related to government /private sector projects mainly the PPP projects:

2.3.1: Public – Private – Partnership (PPP) Projects:

Shen et.al (2007) have used bargaining theory to develop a model that predicts the suitable concession period for a BOT project. The concession period is the time span in which the investor takes the revenues of operating the facility before transferring it to the government. A large concession period is beneficial to the investor but may result in a loss to the government. A short concession period will make the investor reject the BOT contract or increase the service fees in the operation of the project which will result also in a loss to the government. Thus, Shen et.al (2007) used bargaining theory to determine an equilibrium for this situation. The developed BOT bargaining concession model allows the identification of a specific concession period for a project with a BOT contract, which enables both the government and the investor to get as much benefits as possible during the bargaining process for the concession period. To determine the concession period, the model takes into consideration the bargaining behavior of the two parties engaging in the BOT contract. Shen et.al (2007) also used an example case to show that the use of the model is “effective to enable the identification of an agreeable concession span”.

Furthermore, Bayat et.al (2020) have examined the problem of concessionary items in BOT contracts in more depth. The parties involved in the BOT project were separated into three parties: the government, sponsor and lender and they have not considered the sponsor and lender as one party (which was the case in Shen et.al (2007)). Moreover, Bayat et.al (2020) model involved using bargaining theory to create a game model between the sponsor and the government to determine the value of the concession period length and equity to debt ratio simultaneously considering the lender’s requirements. This separation allows a better understanding of “the conflicting financial points of view and interests” regarding the concessionary items. The sponsor wants a long concession period and low equity level while the government wants a short concession period and a minimum equity level. On the other hand, the lender wants the highest possible equity level. Thus, determining the values of the concessionary items is extremely important for a successful implementation of a BOT project. Also, Bayat et.al (2020) validated the model on a real-world BOT project and it was found to be applicable and efficient in enabling the parties to “predict the final agreement point” and preventing the failure of the negotiation or costly renegotiations.

Moreover, researchers have used bargaining theory in risk allocation for PPP projects. In fact, proper risk allocation is mandatory for PPP projects to be successful. Li et.al (2017) used bargaining theory to develop a bargaining game model of risk allocation between the public and private sectors, the two parties of a PPP project. The model determines an equilibrium risk allocation for the two PPP parties in the case where the bargaining process was initiated in the first round by the public sector or in the case it was initiated by the private sector. The model also gives the risk ratio that each of the public and private sectors should take in a PPP project to ensure a smooth implementation of the project. Li et.al (2017) also validated the model on a real PPP project case study and the model was found to be effective and practical as it allows the two PPP parties to achieve a fair risk allocation.

Also, most of the researches related to the determination of the length of the concession period for PPP projects assume the availability of perfect information for the parties while determining the concession period, which is not reasonable. To tackle this issue, Jin et.al (2020) developed a game model (based on bargaining theory) for determining the length of the concession period in the cases of imperfect information and revenue uncertainty. The model also reflects the revenue risks borne by the two PPP parties. The model was also validated on a real highway PPP project in China and it was found that the model is effective and is able to determine the suitable concession period length that would contribute to a "win-win situation for governments and private investors". The model contributes in "reducing the chance of renegotiation" after bidding or contracting as well. Moreover, the outcome of data analysis indicated the government preferences highly influence the optimal length of the concession period in a PPP project (Jin et.al, 2020).

2.4: Bargaining Theory Applications in Other Fields:

However, the use of bargaining theory is not limited to the construction industry only. In fact, bargaining theory was applied by researchers from different disciplines to find solutions to major problems. For instance Carraro et.al (2007) applied bargaining theory in solving water management problems by proposing an approach that helps decision makers in shortening the time needed to reach an agreement with the stakeholders regarding water issues as well as helping them selecting policies that are self-enforcing and acceptable by the public. Also, according to Powell (2002), the international relations theory considers the origins and outcomes of wars as a bargaining process and thus bargaining theory could be applied to analyze a war (which is mainly a bargaining problem) and determine the solution to this problem that leads to the termination of war. Moreover, Yu et.al (2012) used bargaining theory to predict the negotiations outcomes of bilateral contracts between electricity generation companies and load-serving entities in a wholesale electric power market. Additionally, Forgo et.al (2005) modeled the negotiations for reducing the greenhouse gas accumulation in the atmosphere as extensive form game and applied bargaining theory to predict the outcomes of these negotiations.

2.5: Research Gap:

Based on the analysis of existing research conducted, there are several research gaps that currently exist. These can be summarized as follows:

1. Previous research on negotiations using bargaining theory has focused primarily on PPP projects only, with limited research being conducted on other contract types. The most recent research publications related to the application of bargaining theory in construction cover PPP/BOT projects mainly.
2. Limited research has been conducted on using bargaining theory to estimate the bid price percentage the contractor should lower during negotiation with the project owner.
3. Limited research related to the application of bargaining theory in the Egyptian construction industry was conducted.

Accordingly, this research aims to extend the use of bargaining theory to cover other types of project delivery methods. Thus, this research provides a novel approach to an innovative topic that was not addressed before in the literature and contractors may benefit from the developed model in their negotiation process.

Moreover, the main objective of this research is to use bargaining theory to develop a decision support tool / model to predict the percentage of the bid price the contractor should lower (during negotiations as explained in Chapter 1) and the strategy that the contractor should follow during negotiations, which was not addressed in the literature so far.

Chapter 3: Research Methodology

3.1 Introduction:

In this chapter, the detailed research methodology adopted to tackle the research objectives is explained. As discussed in Chapter 1, the main objective of this research is to provide contractors with a decision support tool / model that helps in determining the strategy the contractor should follow in the negotiation process in order to maximize his profit. The conducted work to tackle the aforementioned objective is demonstrated in the below figure:

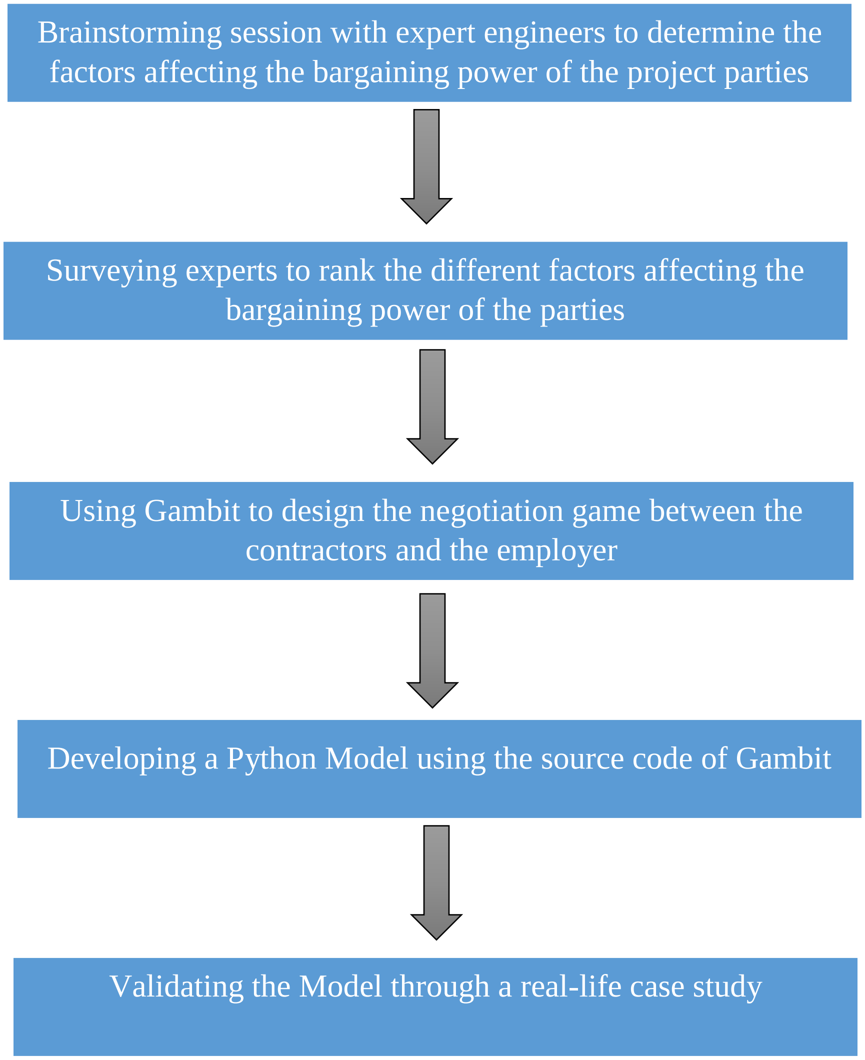


Figure 2: Research Methodology

3.2 Brainstorming Session:

3.2.1 Introduction:

The first step to determine the factors that affect the bargaining power of the negotiating parties (contractor / employer) in a tender for a certain construction project was to conduct a brainstorming session with expert engineers (experience +10 years) in medium and large size construction companies in Egypt. These experts belonged to the 3 main sectors of the construction industry: developers, consultants and contractors. It was important to ensure that the 3 parties of the construction industry were represented in this brainstorming session in order to investigate the subject matter from the 3 different perspectives, which makes the analysis of the negotiation process more accurate and makes the study more reliable.

There were no specific questions prepared to be asked to the experts who participated in the brainstorming session in order not to guide the discussion to a specific direction. Instead, the experts who belong to different sectors of the construction industry were left to ask each other the questions they find relevant to the subject. In fact, this strategy led to the determination of a number of factors that affect the bargaining power of the two parties (employer/ contractor) during their negotiations for a certain contract and thus the factors affecting the tendering strategies of the contractors.

It is worth mentioning that the factors determined by the experts during the brainstorming session were based on their experience from the actual projects they worked on in Egypt and which they find that they directly affect the tendering strategies of contractors.

3.2.2 Experts Selection:

As shown in **Appendix A**, the criteria of selecting the experts who participated in the brainstorming session and their companies are as follows:

- The experts should have a minimum of 10+ years of experience in the construction field.
- The experts should have adequate knowledge of the negotiations stage and the strategies followed by the two negotiating parties (employer / contractor) during the tender negotiation. Thus, the selected experts had to belong mainly to one of the following units in their companies:
    - Tendering Unit
    - Cost Control Unit
    - Procurement Unit
    - Contracts Unit
- The experts should be working in a large and medium size reputable companies in Egypt. In fact, the experts were carefully selected to belong to Class A contractors, consultancy houses and major developers in the Egyptian market. This selection was to ensure the conduction of a fruitful brainstorming session and to obtain reliable outcomes to be used in the coming stages of the research.

3.2.3 Brainstorming Session Conduction:

As previously explained, the brainstorming session was conducted with expert engineers having a minimum of 10+ years of experience. In order to accommodate the tight schedules of these experts and to follow the Covid-19 precautionary measures, and for easier communication, it was decided to conduct the brainstorming session online through the ZOOM platform after working hours. In fact, this allowed all of the 7 invited experts to participate in the session. The duration of the conducted session was 2 hours including the introduction of the topic.

3.2.4 Brainstorming Session Demography:

As shown in Figure 3, the years of work experience of the experts who participated in the brainstorming session vary from 10 to 30 years. 57% of the experts have 10-15 years of experience, 14% have 15-20 years of experience and 29% have 20-30 years of experience.

Figure 3: Experts Years of Experience - Brainstorming Session

Figure 4 shows that 29% of the experts work in a developer company while 42% work in a consultant company and 29% work in a contractor company.

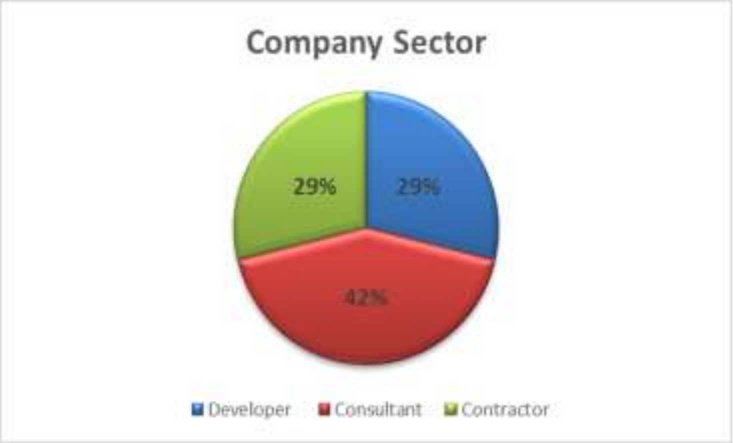


Figure 4: Experts Company Sector - Brainstorming Session

Moreover, 44% of the participants are experts in the cost control area while the remaining 56% are experts in other relevant areas (Figure 5).

Figure 5: Experts Positions - Brainstorming Session

Figures 3, 4 and 5 show that the invited experts are in established positions and have adequate years of experience to analyze the negotiations stage and the strategies followed by the two negotiating parties (employer / contractor).

Please refer to **Appendix B** for the list of factors affecting the bargaining power of the negotiating parties determined during this brainstorming session.

3.3 Surveying Experts: Ranking of Factors Affecting the Bargaining Power

3.3.1 Introduction:

The brainstorming session led to the determination of 35 factors that affect the bargaining power of the negotiating parties (contractor / employer) in a tender for a certain construction project. All of these factors are important and there was no disagreement between the experts who participated in the brainstorming session on the importance of these factors. Thus, in order to rank the importance of these factors, it was decided to conduct a survey (a questionnaire was developed) to identify the degree of significance of these factors. The survey participants were asked to rate these factors based on their experience in the projects in which they have been involved through the years. Also, the participants had to rate all the provided factors in the questionnaire.

3.3.2 Survey Architecture:

The questionnaire consists of 6 main sections as follows:

- Section 1: is an introductory section that gives a brief explanation of the research and survey objectives. This section also explains the five-point scale (with 1 being not important, 5 being extremely important) that should be used by the survey participants in rating the factors in the sections to follow.
- Section 2: consists of 5 questions regarding the survey participants' contact information, position, company name and type.
- Section 3: consists of 19 factors for the participants to rate and which are related to the contractor's initial tendering strategy.
- Section 4: consists of 8 factors for the participants to rate and which are related to the contractor's willingness to lower his original tender price during the negotiations stage.
- Section 5: consists of 4 factors for the participants to rate and which are related to the Contractor having the upper hand during the negotiations stage.
- Section 6: consists of 4 factors for the participants to rate and which are related to the Employer having the upper hand during the negotiations stage.

A sample of the questionnaire distributed to the survey participants is shown in **Appendix C**.

3.3.3 Survey Sample Selection:

The criteria of selecting the survey participants are the same criteria used in the selection of the experts participating in the brainstorming sessions (stated in section 3.2.2). Mostly, the survey participants should have adequate knowledge of the negotiations stage as well as having a minimum of 10+ years of experience in the construction field. Moreover, the survey participants should be working in reputable companies in Egypt in each of the 3 construction sectors (developers, consultants and contractors). The participants' positions range from senior engineers level to executives level in their companies.

Regarding the sampling methods, it is worth highlighting that there are several methods that can be used to select a sample from a given population. However, researchers must select the most suitable sampling methods for their research in order to reach reliable findings and conclusions. There are several sampling methods used in the sample selection of this survey. A combination of the following methods was used to select the sample of the survey participants:

- Stratified Random Sampling: which is a sampling method in which a population is divided into mutually exclusives groups and simple random samples are taken from each of these groups (Hibberts et.al, 2012). In this survey, the population is divided into 3 groups: developers, consultants and contractors and the survey participants are selected randomly from each of these 3 groups. As mentioned earlier, each member of these 3 groups should have a minimum of 10+ years of experience. The main advantage of this sampling method is that it reflects accurately the population being studied since it ensures that each group of the population is properly represented within the sample (Hibberts et.al, 2012).

- Simple Random Sampling: which is a sampling method in which every element in a population has an equal chance of being included in the sample (Hibberts et.al, 2012). In this survey, all members of the 3 groups have an equal chance to be selected to take the questionnaire. This sampling technique is the easiest technique that can be used to extract a sample from a population (Hibberts et.al, 2012).
- Referral Sampling / Snowball Sampling: which is a sampling method in which researchers ask each survey participant to recruit one or more potential participants who meet the set criteria and who might be willing to participate in the survey (Hibberts et.al, 2012). The main advantage of using this sampling technique is that it minimizes the amount of time required to obtain the required sample size (Hibberts et.al, 2012).

3.3.4 Sample Size:

Calculating the required sample size is an important step in ensuring that the conclusions drawn from the survey results are accurate and representative for the whole population. There are several formulae in the literature that are used by the researchers to determine the appropriate sample size depending on the sampling technique used. One of the well-recognized formulae for calculating representative sample size when the population is infinite and simple random sampling technique is used, is Cochran's formula and which was used in a similar research, conducted by Elbashbishy et.al (2019), to determine the sample size needed to reflect the population.

In this research, Cochran's formula is used to determine the required representative sample size. The formula is as follows:

$$n_0 = \frac{z^2 * p * q}{d^2}$$ (Cochran, 1977),

Equation 1: Cochran's formula

Where,

- $n_{0:}$ is the sample size
- z: is the selected critical value of the desired confidence level. For a 90% confidence level, z = 1.64
- p: is the estimated proportion of the population who would select the same answer. As the actual value of p is not known and in order to be conservative, maximum variability is assumed, i.e. the value of p would be 0.5
- q = 1-p = 1- 0.5= 0.5
- d: is the acceptable margin of error. In the subject case, d is taken to be equal to 0.15

By incorporating these values in Cochran's formula, the required sample size is equal to:

$$n_0 = \frac{1.64^2 * 0.5 * 0.5}{0.15^2}$$ = 30 survey participants

3.3.5 Survey Conduction:

In order to accommodate the tight schedules of the survey participants and to follow the Covid-19 precautionary measures and for easier communication, it was decided to create the questionnaire using Google Forms. Thus, sharable links were sent out to the survey participants to access the questionnaire whenever it is convenient to them.

3.3.6 Survey Execution:

After preparing an initial draft of the questionnaire, it was sent first to the research advisors for trial and ensuring its ease of execution to the required number of survey participants. After receiving their comments, the questionnaire was revised and a final draft was prepared and distributed to the participants.

As mentioned earlier, the questionnaire was distributed to participants from the 3 sectors (developers, consultants and contractors) who meet the set criteria. Some of these participants were asked to recruit one or more potential participants who meet the set criteria and who might be willing to participate in the survey. The reason for resorting to this snowball sampling technique is the difficulty of finding participants with 10+ years of experience and with positions related to the tender negotiations field. A total of 38 responses were received and after disregarding the duplicated responses, only 36 of them were valid and considered in the study (2 participants submitted mistakenly their responses twice). Thus, the required sample size of 30 participants estimated using Cochran's formula was satisfied. The results of the survey are discussed in details in the following chapter.

3.3.7 Survey Demography:

As shown in Figure 6, 10% of the experts work in a developer company while 50% work in a consultant company and 40% work in a contractor company.

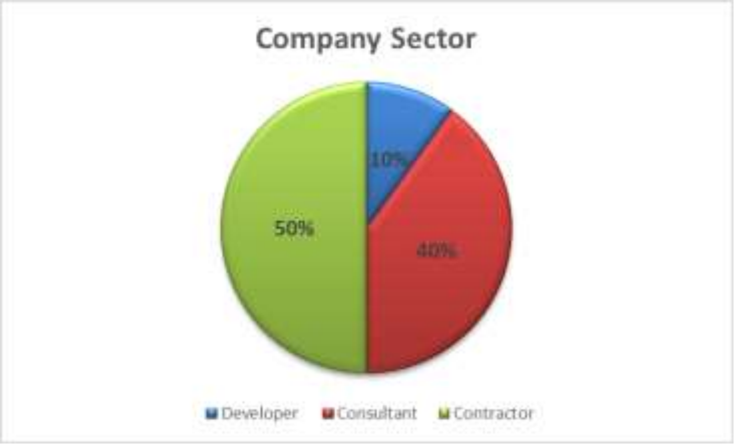


Figure 6: Survey Participants - Company Sector

Figure 7 shows that 30% of the participants are experts in the cost control area while the remaining 70% are experts in other relevant areas.

Figure 7: Survey Participants - Positions

Moreover, 35% of the participants have more than 20 years of experience, while the remaining 65% of the participants have 10-20 years of experience (Figure 8).

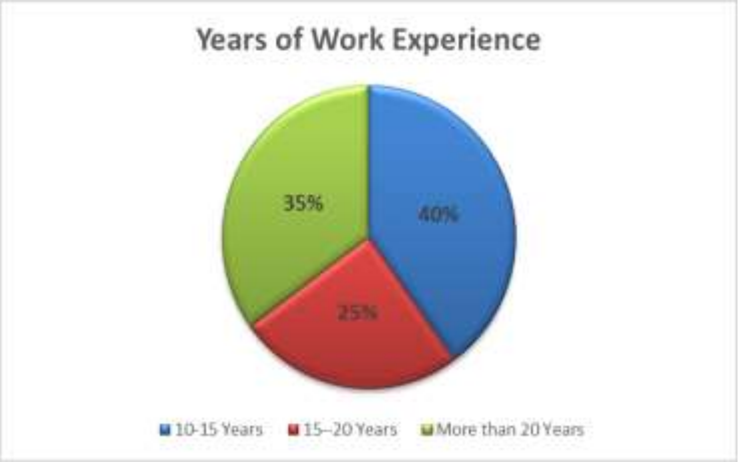


Figure 8: Survey Participants - Years of Work Experience

Figures 6, 7 and 8 show that the survey participants are in established positions and have adequate years of experience to rate the factors affecting the bargaining power of the two negotiating parties (employer / contractor).

3.4 Designing the Negotiations Game: Gambit

3.4.1 Introduction:

After receiving the responses from the survey participants and ranking the different factors affecting the bargaining power of the 2 negotiating parties (employer/contractor), a discussion was conducted with the research advisors on the scope to be considered while designing the negotiations game. Due to the complexity of the research topic and the unfamiliarity with game theory and the python programing language, it was decided to consider only the highest ranked factor from the survey in the game. Considering more than one factor will result in having a more complex Gambit tree and thus higher computational power will be required to compute the equilibrium of the games. It is worth to mention that Gambit is an open-source collection of tools for performing computation on finite, non-cooperative games in game theory. The Gambit version used in this research is Gambit 15.1.1 and which is the current stable version (McKelvey et.al, 2015).

3.4.2 Description of the Negotiations Game:

3.4.2.1 Introduction:

The game discussed in this research simulates the situation at the end stage of the negotiations where the employer has already determined the two contractors with the lowest bid prices and are technically qualified to take the project. Before starting the negotiations, the employer has his own fair price estimate of the project (usually provided by the design consultant). The employer starts the negotiations with the higher bid price contractor of the two contractors (game 1) and his value of the project at the end of this game would be the minimum between his original fair price estimate and the contractor's final price at end of game 1. Then, the employer enters into negotiations with the other contractor (game 2) after he has already known the final and best price of the first contractor. After negotiating with the second contractor, the employer accepts his offer if it is less than the final price of the first contractor (thus the employer's value of the project is lower at end of game 2 compared to game 1). Otherwise he rejects the offer and accepts the final offer of the first contractor. Thus, the game discussed in this research consists actually of 2 games: game 1 is between the employer and the contractor having the higher bid price and game 2 is between the employer and the contractor having the lower bid price. The following diagram illustrates the aforementioned games 1&2:

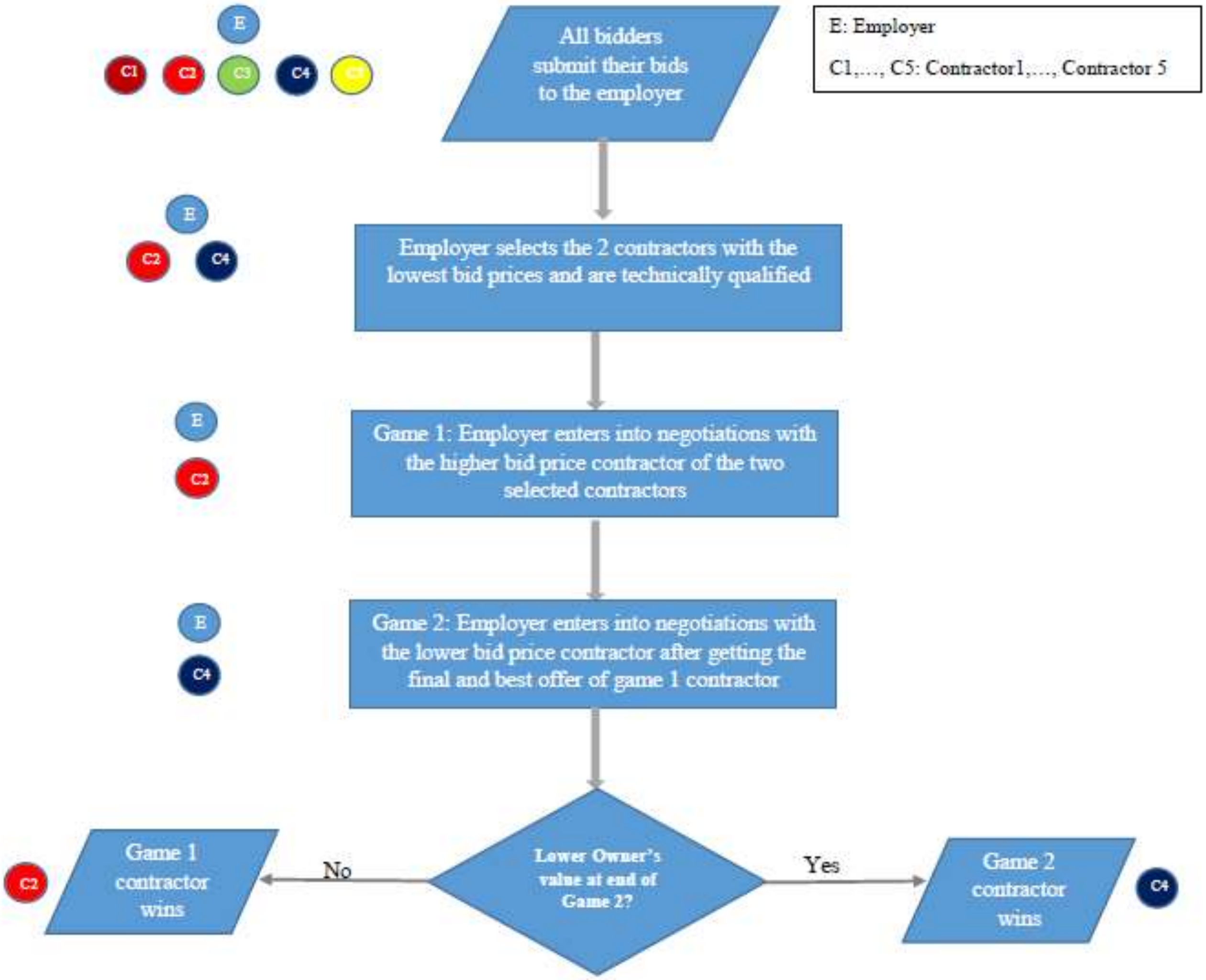


Figure 9: Illustration Diagram for Games 1 & 2

3.4.2.2 Players Strategies:

In each of these games, the 2 players (employer / contractor) have certain actions/decisions that they may take:

- Initial discount percentage: after submitting the bids, and when the contractor enters into negotiations with the employer, he starts by offering an initial discount percentage to his initial bid price. Based on discussions with the experts who participated in the brainstorming session and in the survey, it was decided that the initial discount percentage considering the highest ranked factor of the survey could range from 0% to 4%.

- Employer’s decision based on the initial discount percentage offered by the contractor: Based on the initial discount percentage offered, the employer may take one of the following decisions:
    - Accept the offer: the employer may accept the contractor’s offer directly without entering into further negotiations with him.
    - Withdraw the offer / Final Rejection: the employer may withdraw the contractor’s offer from his selection pool without giving the contractor the chance to make additional discount. The employer may resort to this choice in case the contractor’s initial discount percentage made does not show his serious willingness to take the project.
    - Reject Offer /Requesting Final Offer: the employer may also reject the contractor’s offer and asks him to submit a final and best price offer.
- Contractor’s decision based on the employer’s action taken: from the available actions for the employer, the contractor will have only to take an action in case the employer rejects his offer and asks him to submit a final offer. In this case, the contractor may take one of the following decisions:
    - Withdraw: the contractor may decide not to make additional discount and inform the employer of the withdrawal of his offer.

- Additional 1% discount: alternatively, the contractor may decide to make additional discount to increase his chances of winning the project. After discussion with the research advisors and the experts who participated in the brainstorming session and in the survey, it was decided to set the amount of this additional discount to 1%.

3.4.2.3 Players Payoffs:

3.4.2.3.1 Employer's Payoffs:

In the first game with the contractor having the higher bid price, the employer's payoff is mainly:

$$\textit{Employer's payoff} = \textit{fair price estimate (owner's value)} - \textit{bid price of the contractor}$$

Equation 2: Employer's Payoff

A positive payoff means that the employer is getting a bid price lower than his original estimate while a negative payoff means that the employer is getting a bid price higher than his estimate / budget for the project. In case the employer decides to withdraw the contractor's offer from his selection pool or he rejects the contractor's offer and the contractor elects to withdraw his offer, the employer's payoff in these cases would be as follows:

$$\textit{Employer's payoff} = \text{MIN (fair price estimate, contractor's final offer)} - \text{original bid price of the other contractor (who would participate in game 2)}$$

Equation 3: Employer's Payoff in case of Rejection of the Contractror's Offer

In the second game with the contractor having the lower bid price, the fair price estimate or the owner's value will no longer have the same value it had at the start of the first game; it would change according to the equilibrium of the first game, i.e. it would be equal to:

$$\textit{Owner's value @ game 2} = \text{Original fair price estimate} - \text{Employer's payoff @ equilibrium of game 1}$$

Equation 4: Owner's value @ game 2

Moreover, in case the employer decides to withdraw the contractor's offer or rejects the contractor's offer and the contractor elects to withdraw his offer, the employer's payoff in these cases would be equal to zero; as it is assumed that in these cases the offers received from the contractors are higher than the employer's budget and thus no contractor would be awarded the project.

3.4.2.3.2 Contractor's Payoffs:

Whether in game 1 or game 2, the contractor's payoff is mainly calculated as follows:

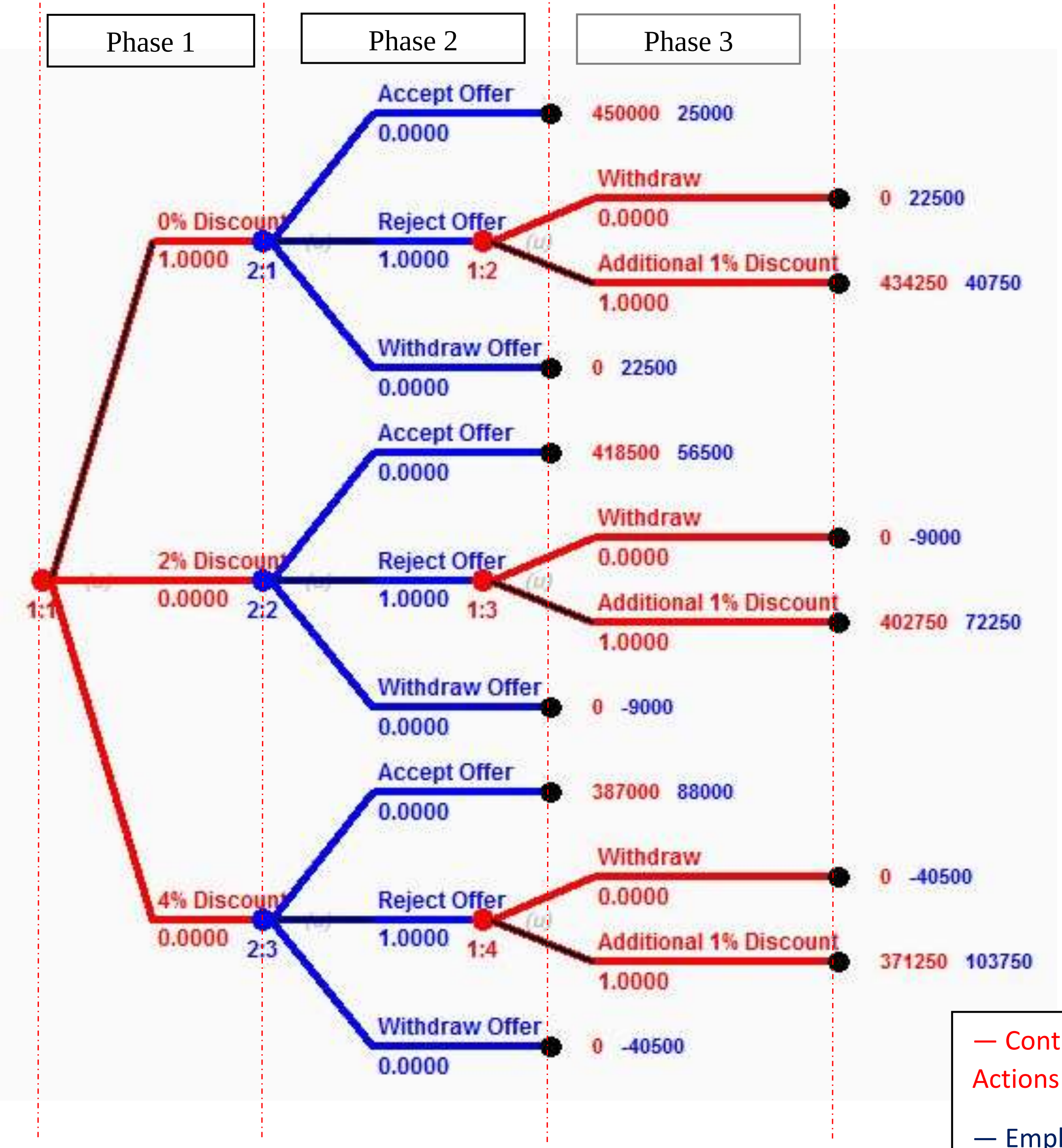
$$\textit{Contractor's payoff} = \textit{Contractor's bid price after discount} - \textit{project direct cost}$$

Equation 5: Contractor's Payoff

A positive payoff means that the contractor's submitted bid is higher than his direct cost estimated for the project while a negative payoff means that the contrary. Moreover, in case the employer decides to withdraw the contractor's offer from his selection pool or rejects the contractor's offer and the contractor elects to withdraw his offer, the contractor's payoff in these cases would be equal to zero; as it is assumed that in these cases of not being awarded the project, the impact on the contractor would be neutral (neither positive nor negative).

3.4.3 Modeling the Game on Gambit:

After defining the strategies for each player (employer / contractor) and defining the way of calculating the payoffs for each player in both game 1 and game 2, the game can now be modeled on Gambit. It would be suitable to model this game in the extensive form in order to show the order in which the players take their decisions and the information available to them at each stage of the game. Figure 10 shows a sample game modeled on Gambit in extensive form. This sample game is explained in details in the following chapter.



— Contractor's Actions / Payoffs

— Employer's Actions / Payoffs

Figure 10: Sample Game Modeled on Gambit in Extensive Form

As shown in Figure 10, the branches in phase 1 represent the actions that may be taken by the contractor in his first move; which are the possible initial discount percentages the contractor can make. The branches in phase 2 represent the actions that may be taken by the employer in his first move after knowing the decision made by the contractor in his move; the employer may accept, reject or withdraw the contractor's offer as explained earlier. The branches in phase 3 represent the actions that may be taken by the contractor in his second move and which occurs only if the employer decides in his move to reject the contractor's offer and requests him to submit a final and best offer. In this case, the contractor may withdraw his offer or make additional 1% discount as explained earlier. Furthermore, the numbers appearing next to the terminal nodes at the end of each path are the payoffs of the players. Contractor's payoffs are in red while employer's payoff are in blue.

Once the game is modeled on Gambit, the Nash Equilibrium for the game could be computed. Thus, game 1 is modeled first on Gambit and the Nash Equilibrium is computed for it. Based on this computed equilibrium, the employer's value for the project is computed and the payoffs for the players of game 2 are calculated accordingly. Game 2 is then modeled on Gambit and the Nash Equilibrium is computed for it. Based on the equilibrium of game 2, the winning contractor is identified as well as the corresponding strategies selected by the players and their payoffs.

3.5 Developing the Decision Support Tool: Python Model

3.5.1 Introduction:

Determining the payoffs of the contractors remains a challenging task. For instance, the contractor who is competing in the negotiations stage, does not know the exact bid price of his competitor, thus he will not be able to compute the exact payoffs for the game between the employer and his competitor. Knowing the average pricing factor of his competitor, the contractor may make a prediction of his competitor's bid price and calculate the payoffs, but the accuracy of this estimate will remain questionable. Moreover, modeling the game on Gambit with these payoffs will allow only to compute accurately the equilibrium of the games if these payoffs were the actual payoffs of the players, otherwise the accuracy of the computed equilibrium will remain questionable as well.

In fact, the contractor cannot predict the exact bid price of his competitor due to the interference of a random variable (which is the pricing factor chosen by his competitor). However, as the average pricing factors of the contractors and their standard deviation are known (from previous bids), a Monte Carlo simulation could be performed to estimate the bid prices of the contractors. Hence, in order to get reliable results, the games between the employer and the contractors could be simulated several times by assigning the bid price of the contractor a random value each time. Assuming a normal distribution for the pricing factor, the average and the standard deviation of the pricing factor could then be used to assign a bid price for each contractor at each iteration.

However, as the payoffs are inserted manually to Gambit and as the determination of the winning contractor and the strategy followed by him is done manually as well, hence it would be more practical to develop a python model to perform the aforementioned Monte Carlo simulation and make all the required computations automatically and efficiently. Moreover, the python model could be more user friendly and will allow the user to get instantaneous analysis of the situation.

Additionally, it is worth highlighting that developing the python model requires the use of the Gambit's open-source library in python and which can be accessed through the following link: https://gambitproject.readthedocs.io/en/latest/pyapi.html. Also, developing the python model requires installing the Ubuntu operating system and python version 2.7 as this is the version in which the last stable version of the Gambit library was written.

3.5.2 Model Processes:

The following flow chart summarizes how the developed python model / decision support tool works:

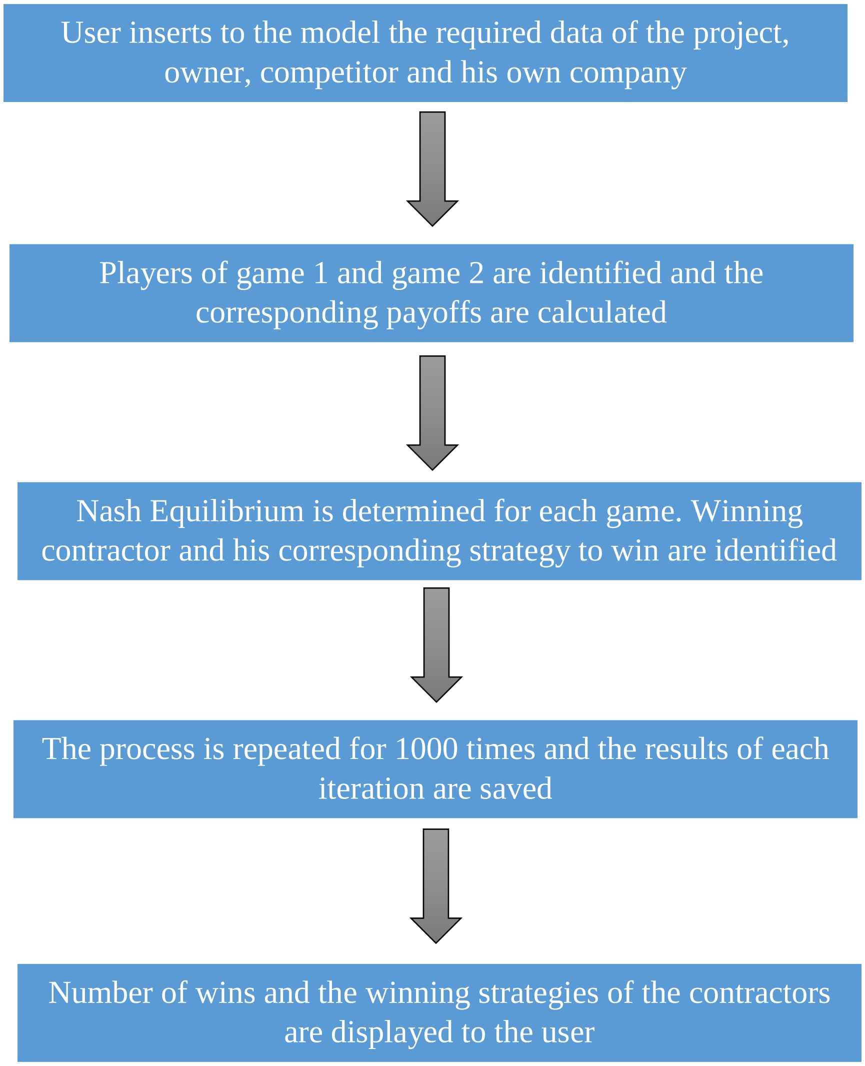


Figure 11: Python Model Processes

3.5.2.1 Input Module:

As discussed in section 3.4, the data required to compute the payoffs of the players are the employer/owner's value of the project, the cost of the project, and the bid prices of the contractors. Accordingly, the developed Python Model requires the user to insert the following:

- Project Direct Cost: which can be estimated easily by the user.
- Owner Value: which represents the fair price estimate of the project.
- Contractor 1 Average Pricing Factor & its Standard Deviation: which can be estimated from previous bids in which contractor 1 participated.
- Contractor 2 Average Pricing Factor & its Standard Deviation: which can be estimated from previous bids in which contractor 2 participated.

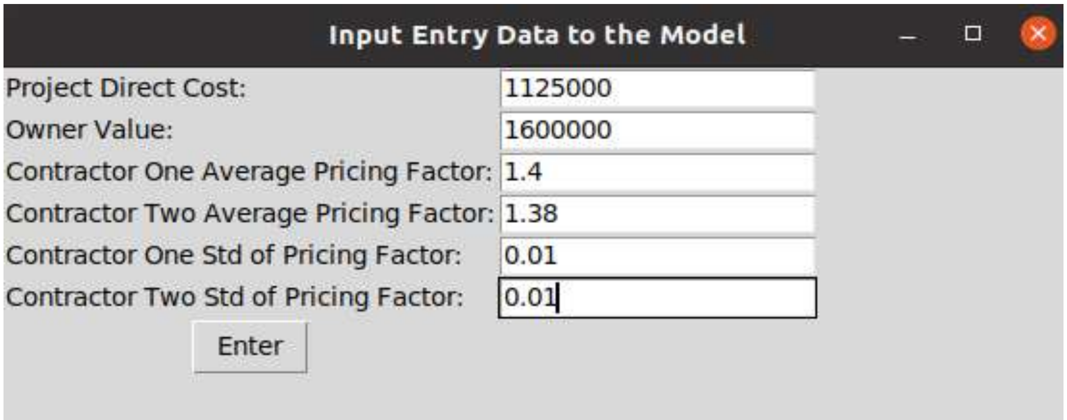
It is worth highlighting that this model / decision support tool could also be used by the employer and not by the contractor only. Also, the accuracy of the model results increases with the number of previous contractors' bids data available to the user.

There are 2 methods to insert the required data to the model by the user. The first method is to insert the data directly to a json file as shown in Figure 12. In this case, the user will have to run the model with the python command which will load and read the json file.

```
{
    "projectCost": 1125000.0,
    "averages": [
        1.4,
        1.38
    ],
    "stds": [
        0.01,
        0.01
    ],
    "ownerValue": 1600000.0
}
```

Figure 12: Data Input - Json File

The second method (which was used) is to develop a more user-friendly graphical interface with the Tkinter library. The user needs to double-click on the file icon of the graphical user interface and inserts the required data directly. The inserted data will then be fed automatically into the developed code. The following figure shows the developed graphical user interface and the required data to be inserted by the user:

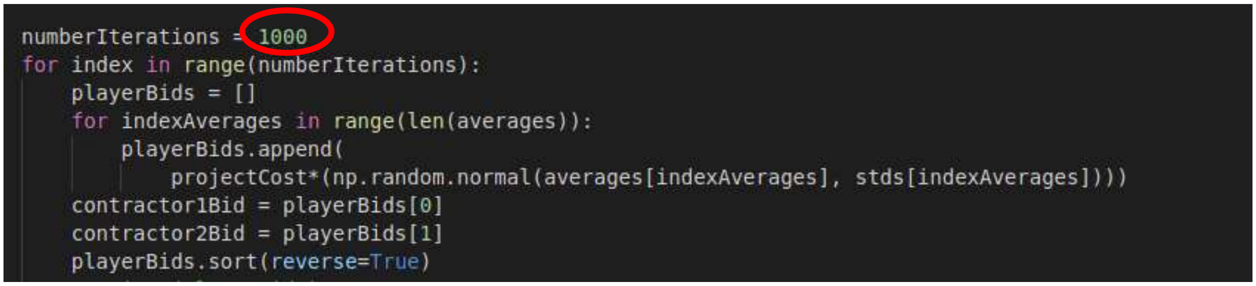


Project Direct Cost: 1125000
Owner Value: 1600000
Contractor One Average Pricing Factor: 1.4
Contractor Two Average Pricing Factor: 1.38
Contractor One Std of Pricing Factor: 0.01
Contractor Two Std of Pricing Factor: 0.01

Figure 13: Data Input - Tkinter

3.5.2.2 Processing Module:

The model begins with reading the game tree created in Gambit (developed in the previous section) and stores it as an object. Then, the model estimates the bid prices of the 2 competing contractors: for each contractor, the user inserted an average pricing factor and its standard deviation; accordingly the model generates a random value for the bid prices (assuming a normal distribution). In fact, Gates (1967) suggested that normal distribution could be used to describe the bidding strategy of contractors. Once the bid prices for the contractors are determined, the players of game 1 and game 2 are identified: as previously discussed, game 1 is between the employer and the contractor having the higher bid price and game 2 is between the employer and the contractor having the lower bid price. The following extraction of the Visual Studio (VS) code shows how the bid prices calculation and the sorting of the contractors are performed:

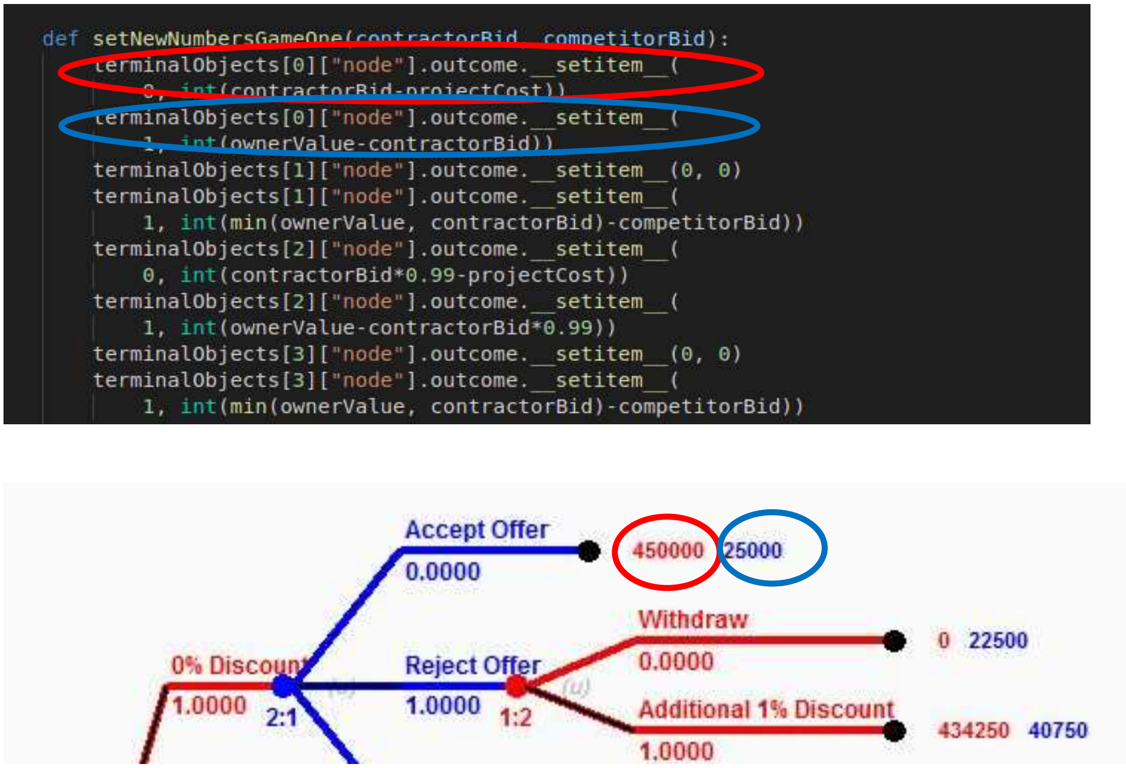


```
numberIterations = 1000
for index in range(numberIterations):
    playerBids = []
    for indexAverages in range(len(averages)):
        playerBids.append(
            projectCost*(np.random.normal(averages[indexAverages], stds[indexAverages])))
    contractor1Bid = playerBids[0]
    contractor2Bid = playerBids[1]
    playerBids.sort(reverse=True)
```

Figure 14: Python Model - Bid Prices Calculation

Please refer to **Appendix D** for the complete VS code developed.

After computing the bid prices and identifying the players of each game, the payoffs of the players can now be computed. The equations used to calculate the payoffs are the same equations discussed in section 3.4.2.3. The following extraction of the VS code shows how the payoffs are computed for game 1:



```
def setNewNumbersGameOne(contractorBid, competitorBid):
    terminalObjects[0]["node"].outcome.__setitem__(
        0, int(contractorBid-projectCost))
    terminalObjects[0]["node"].outcome.__setitem__(
        1, int(ownerValue-contractorBid))
    terminalObjects[1]["node"].outcome.__setitem__(0, 0)
    terminalObjects[1]["node"].outcome.__setitem__(
        1, int(min(ownerValue, contractorBid)-competitorBid))
    terminalObjects[2]["node"].outcome.__setitem__(
        0, int(contractorBid*0.99-projectCost))
    terminalObjects[2]["node"].outcome.__setitem__(
        1, int(ownerValue-contractorBid*0.99))
    terminalObjects[3]["node"].outcome.__setitem__(0, 0)
    terminalObjects[3]["node"].outcome.__setitem__(
        1, int(min(ownerValue, contractorBid)-competitorBid))
```

Figure 15: Payoffs Calculation

The payoffs are calculated in the same order they appear on the Gambit tree. For instance the first payoffs calculated are the contractor (red circle in the figures above) and the employer's (blue circle) payoffs at 0% initial discount and acceptance of the offer. The remaining 22 payoffs are calculated in the same manner according to their orders on the Gambit tree.

After calculating all the payoffs of the players in game 1, the Nash Equilibrium is computed for this game. Based on the payoff of the employer at the equilibrium of game 1, the payoffs of the players in game 2 are calculated (as discussed in section 3.4.2.3) and finally the Nash Equilibrium is computed for game 2. Figure 16 shows in details how the model handles this process. In fact, the model starts with computing the Nash Equilibrium for game 1, then for each of the terminal objects (the branches that have payoffs associated), it computes the probability of the path (multiplication of all probabilities on the path leading to the branch, included). Also, in case 2 branches of the tree have the same probability and for tiebreak purpose, the model computes the sum of the 2 payoffs on this branch (the branch with higher sum of payoffs is considered the winning branch). Then, the model determines the branch with highest path probability and stores it in an array (each game has 1000 arrays to store the results of the 1000 iterations).

```
def solveGambit(maxActions):
    # print (g.write())
    solvedProfile = solver.solve(g)[0]

    for terminalObject in terminalObjects:
        terminalObject["prob"] = round(solvedProfile.__getitem__(
            terminalObject["action"]) * solvedProfile.realiz_prob(terminalObject["action"].infoset), 4)
        terminalObject["sumPayoffs"] = float(terminalObject["node"].outcome.__getitem__(
            0))+float(terminalObject["node"].outcome.__getitem__(1))

    maxObject = max(terminalObjects, key=lambda terminalObject: (
        terminalObject['prob'], terminalObject['sumPayoffs']))
    maxActions.append(maxObject["action"])
    return maxObject
```

Figure 16: Python Model - Nash Equilibrium Computation

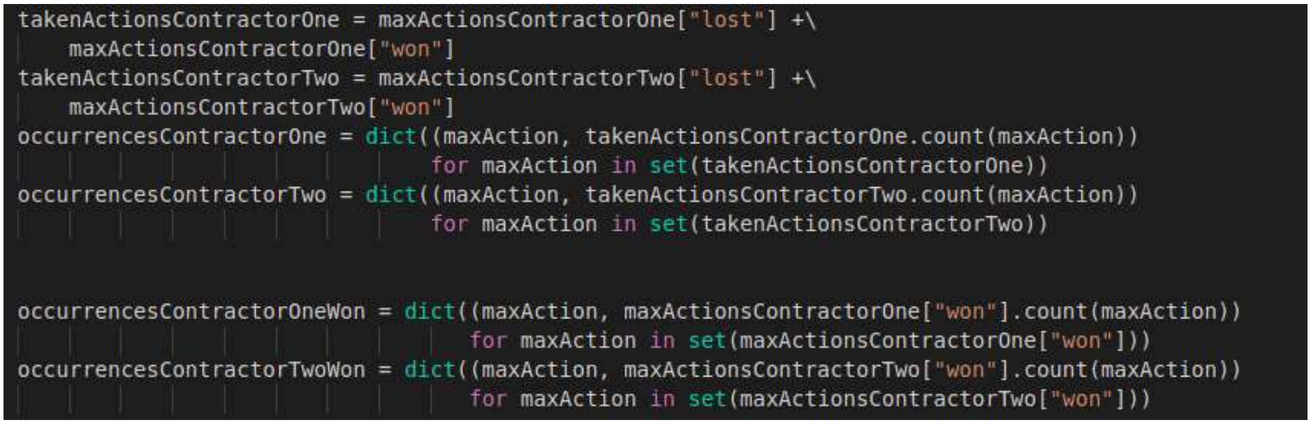
Then based on the results of game 1, the model calculates the payoffs for game 2 and associates them to the terminal objects of game 2 tree. The same process for game 1 is repeated for game 2 and the model determines the branch with highest path probability and stores it in the corresponding array.

After storing the results of game 1 and game 2 of the same iteration, the model determines the winning contractor. In case the payoff of the employer is negative in game 2, then the contractor who played in game 1 is the winning contractor and if the payoff of the employer is positive in game 2 then the contractor who played in game 2 is the winning contractor. Figure 17 shows how this determination of the winning contractor is handled by the model. For each iteration, the model stores for each contractor whether he won or lost the game and it updates the number of wins counter accordingly. Moreover, there is another counter that counts whether the winning contractor played game 1 or game 2 and it is updated after each iteration as well.

```
if contractor1Bid < contractor2Bid:
    if payoffOwnerGameTwo >= 0:
        counterContractorOne += 1
        maxActionsContractorTwo["lost"].append(maxActionsGameOne[index])
        maxActionsContractorOne["won"].append(maxActionsGameTwo[index])
    else:
        counterContractorTwo += 1
        counterGameOneWon += 1
        maxActionsContractorTwo["won"].append(maxActionsGameOne[index])
        maxActionsContractorOne["lost"].append(maxActionsGameTwo[index])
else:
    if payoffOwnerGameTwo >= 0:
        counterContractorTwo += 1
        maxActionsContractorTwo["won"].append(maxActionsGameOne[index])
        maxActionsContractorOne["lost"].append(maxActionsGameTwo[index])
    else:
        counterContractorOne += 1
        counterGameOneWon += 1
        maxActionsContractorTwo["lost"].append(maxActionsGameOne[index])
        maxActionsContractorOne["won"].append(maxActionsGameTwo[index])
```

Figure 17: Python Model - Determination of Winning Contractor

Furthermore, as shown in Figure 18, for each iteration the model stores for the winning contractor the path/strategy that led him to win and keeps counting the number of time this path occurred for this contractor.



```python
takenActionsContractorOne = maxActionsContractorOne["lost"] +\
    maxActionsContractorOne["won"]
takenActionsContractorTwo = maxActionsContractorTwo["lost"] +\
    maxActionsContractorTwo["won"]
occurrencesContractorOne = dict((maxAction, takenActionsContractorOne.count(maxAction))
                                for maxAction in set(takenActionsContractorOne))
occurrencesContractorTwo = dict((maxAction, takenActionsContractorTwo.count(maxAction))
                                for maxAction in set(takenActionsContractorTwo))


occurrencesContractorOneWon = dict((maxAction, maxActionsContractorOne["won"].count(maxAction))
                                   for maxAction in set(maxActionsContractorOne["won"]))
occurrencesContractorTwoWon = dict((maxAction, maxActionsContractorTwo["won"].count(maxAction))
                                   for maxAction in set(maxActionsContractorTwo["won"]))
```

Figure 18: Python Model - Storing of Winning Strategies

These procedures are repeated for 1000 iterations as shown in Figure 14 and the final results are finally displayed to the user. It is worth highlighting that it was decided to carry 1000 iterations based on the recommendations of several studies conducted in the construction management field using Monte Carlo simulation such as Gurmu and Ongkowijoyo (2020), Raoufi and Fayek (2020), and El Asmar et.al (2009).

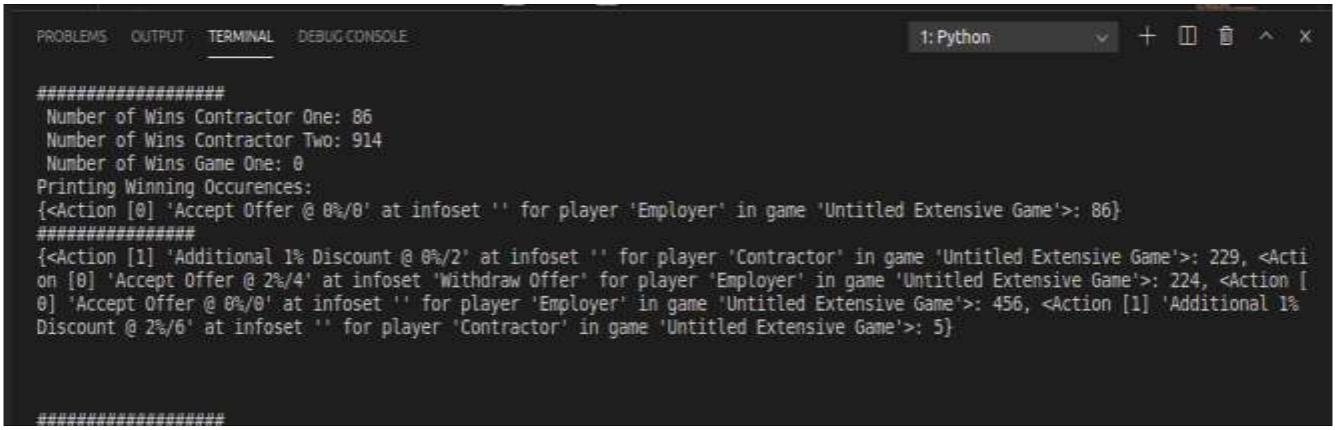
3.5.2.3 Output Module:

The developed Python Model displays to the user the following results:

- <u>Number of wins for each contractor</u>: the user can know how many games each contractor has won from the conducted 1000 iterations.
- <u>Number of wins for game 1:</u> the user can know how many times the contractor who played in game 1 (contractor with the higher bid price) won the project.

- Distribution of the winning strategies of the contractors: the user can know the distribution of the strategies that led to winning for each contractor.

There are 2 methods to display these outputs to the user. The first method is that the user can run the python file manually and the results will be displayed in the terminal as shown in Figure 19.



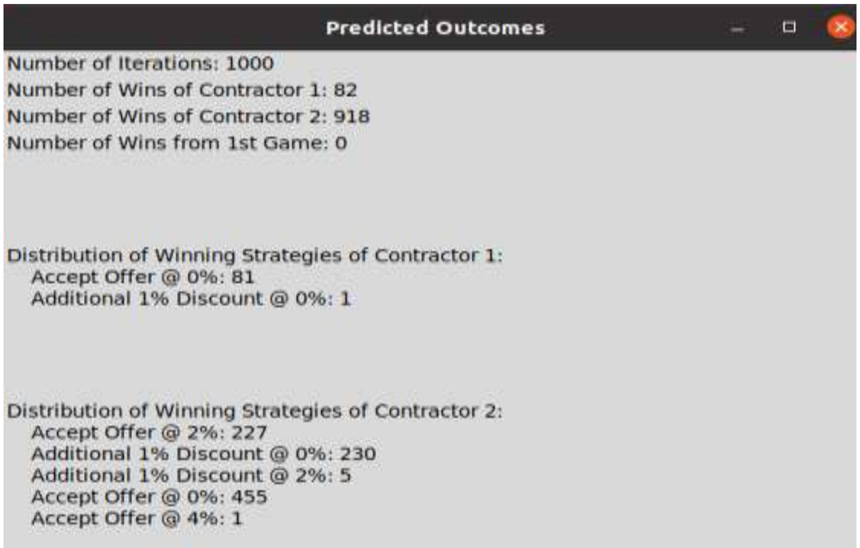
```
###################
 Number of Wins Contractor One: 86
 Number of Wins Contractor Two: 914
 Number of Wins Game One: 0
Printing Winning Occurences:
{<Action [0] 'Accept Offer @ 0%/0' at infoset '' for player 'Employer' in game 'Untitled Extensive Game'>: 86}
################
{<Action [1] 'Additional 1% Discount @ 0%/2' at infoset '' for player 'Contractor' in game 'Untitled Extensive Game'>: 229, <Acti
on [0] 'Accept Offer @ 2%/4' at infoset 'Withdraw Offer' for player 'Employer' in game 'Untitled Extensive Game'>: 224, <Action [
0] 'Accept Offer @ 0%/0' at infoset '' for player 'Employer' in game 'Untitled Extensive Game'>: 456, <Action [1] 'Additional 1%
Discount @ 2%/6' at infoset '' for player 'Contractor' in game 'Untitled Extensive Game'>: 5}



###################
```

Figure 19: Results Output - Terminal

The second method (which was used) is to develop a more user-friendly graphical interface with the Tkinter library. After the user inserts the data required and press the enter button, the below graphical user interface with the results is displayed to the user:



```
Number of Iterations: 1000
Number of Wins of Contractor 1: 82
Number of Wins of Contractor 2: 918
Number of Wins from 1st Game: 0



Distribution of Winning Strategies of Contractor 1:
   Accept Offer @ 0%: 81
   Additional 1% Discount @ 0%: 1



Distribution of Winning Strategies of Contractor 2:
   Accept Offer @ 2%: 227
   Additional 1% Discount @ 0%: 230
   Additional 1% Discount @ 2%: 5
   Accept Offer @ 0%: 455
   Accept Offer @ 4%: 1
```

Figure 20: Results Output - Tkinter

The verification and validation of the model are discussed in details in the following chapter. Moreover, a real case study used for model verification is presented in the following chapter.

3.5.3 Python Model Advantages over Gambit:

The developed decision support tool / python model provides the users with several advantages compared to the Gambit interface, such as:

- Friendly Graphical User Interface: the user will only deal with a GUI to insert the required data related to the players and presses the enter button simply to get the simulation results.
- No Need to Draw the Gambit Tree: As the user deals only with the GUI of the python model, he will not be required to draw the Gambit tree at all (as it is a part of the developed code), which will save time for the user and make the process easier to him.
- Payoffs are Calculated Automatically: The user will not need to calculate manually the payoffs of the players. The model calculates the payoffs of the players for Game 1 and Game 2 automatically (the user is no longer required to use the Equilibrium of Game 1 to determine the payoffs of the players in Game 2).
- Conducting Large Simulations Effectively: Conducting 1 trial only is not sufficient for the user to make a decision regarding his negotiation strategy: the strategy that leads the contractor to win most of the times is not necessarily the sane strategy obtained from a single trial conducted on Gambit. The developed python model provides the user with the results of the 1000 simulations and tells the user the number of wins he scored based on his bidding behavior and gives him a distribution of the negotiation strategies that lead him to these wins. The model gives the user these important results promptly and which

represents a great advantage over the Gambit interface where the user has to conduct each trial manually on it.

- <u>Helping the User in Making Huge Savings:</u> Thus, this python model is an effective decision support tool that processes the data inserted by the user and gives him a summary of 1000 simulations without any intervention required from him. The model provides to the user his probability of winning based on his bidding behavior and the strategies that led to these wins. In fact, these results will help the user make rational decisions in the negotiations; a matter that can saves him a huge amount of money that may be left on the table as shown in the real case study discussed in the following chapter.

Chapter 4: Results, Verification and Validation

4.1 Introduction:

This chapter begins by presenting the survey results and the ranking of the different factors affecting the bargaining power of the 2 players (employer and contractor). Then, the verification of the developed python model is presented. Afterwards, the importance of the model outputs / predictions is discussed. Finally, a real case study is presented to validate the model and to highlight the value added of the developed tool.

4.2 Survey Results:

As discussed in chapter 3, the survey participants were asked to rate the factors affecting the bargaining power of the 2 negotiating parties (employer and contractor) based on their experience in the projects in which they have been involved through the years. The below table is a summary of the collected ratings and it shows the overall ranking of the factors as well.

Table 2: Ratings and Raking of the Factors Affecting the Bargainning Power of the Negotiating Parties

Ranking	**Factor**	Participants Votes					**Average Score**
		1	2	3	4	5	
1	Contractor's need for Taking Projects	0	0	0	14	22	4.61
2	When the Project is of a Special Type (Limited Number of Specialized Contractors to Choose From)	0	0	4	16	16	4.33
3	Availability of Clearer Information Regarding the Technical Documents (Specifications, Material Selection, etc.)	0	0	5	15	16	4.31

4	Transferring some of the Risks to the Employer (Materials Price Fluctuation, etc.)	0	2	4	12	18	4.28
5	Size of the Project compared to the Contractor's Company size	0	2	5	13	16	4.19
6	Contractor Willingness to Work with a Specific Developer or Taking a Specific Project	0	0	9	14	13	4.11
7	Financial Position of the Developer	0	5	6	7	18	4.06
8	Incompleteness of the Design Documents	0	2	5	18	11	4.06
9	Market and Economy Condition	1	2	5	15	13	4.03
10	Employer Offering a Shorter Payment Period and / or Payment for Materials On Site	0	4	5	14	13	4.00
11	Contractor's need for the project as a "reference"	3	2	4	11	16	3.97
12	Size of the Contractor's Company	0	0	9	20	7	3.94
13	The Contractor Having more Time to Understand the Project Compared to the Short Period Available for the Preparation of Tender	0	0	11	16	9	3.94
14	Reputation of the Developer	0	2	12	11	11	3.86
15	Risk Allocation in the Contract	0	4	9	12	11	3.83
16	There are other Packages to be Tendered in the Future by the Employer	0	2	11	14	9	3.83
17	Size VS Duration of the Project	0	4	9	14	9	3.78
18	Material Availability in the Market	2	4	9	7	14	3.75

19	The Project is a Public Sector Project and the Employer is a Governmental Entity	0	8	7	7	14	3.75
20	Probability for Frequent Changes in Rules & Regulations	2	2	11	11	10	3.69
21	The Employer Invites Small Contractors to Bid for the Project	0	4	12	11	9	3.69
22	Competitors Participating in the Tender	0	2	11	20	3	3.67
23	Contractor Willingness to Keep Control on an Area Dominated by them	0	2	13	16	5	3.67
24	Location of the Project	2	4	9	12	9	3.61
25	Tight Project Duration and Employer's Unwillingness to Increase the Project Duration	0	0	18	15	3	3.58
26	The Final Number of Competitors in the Negotiations Stage	0	4	16	9	7	3.53
27	Employer Willingness to Change the Specifications to Get Lower Prices	3	5	5	18	5	3.47
28	Contract Type: Lump-Sum or Re-measured	4	5	11	5	11	3.39
29	The Contractor Does Not Submit the Tender Bond	4	8	8	3	13	3.36
30	The Submitted Tender Prices from All Contractors Are Higher than the Employer's Price Estimate of the Project	3	4	13	11	5	3.31
31	Project Delivery Method	3	3	18	5	7	3.28

32	Tender Scoring Criteria (% for Technical Offer VS % for Financial Offer)	2	8	13	5	8	3.25
33	Opportunity to enter in a JV	2	9	17	4	4	2.97
34	Reputation of the Engineer	3	6	23	2	2	2.83
35	Possible Support from Overseas Branches	3	11	15	4	3	2.81

As mentioned in section 3.4.1, it was decided to consider only the highest ranked factor from the survey “contractor's need for taking projects” in the game design. Based on discussions with the experts who participated in the brainstorming session and in the survey, it was decided that the initial discount percentage offered by the contractor, considering an average level for the aforementioned factor, could range from 0% to 4%.

4.3 Model Verification:

In order to verify that the developed python model is working properly, it was decided to create a fictional game between the employer and the 2 competing contractors. All data related to the project cost, owner’s value and contractors’ bidding behavior were assumed. After stating all the needed assumptions, the players’ payoffs were calculated and the games were modeled first on Gambit. Then, the same data was inserted to the developed python model and the outputs of the model were compared to the Gambit results.

4.3.1 Game 1 on Gambit:

As mentioned earlier, game 1 is between the employer and the contractor having the higher bid price. The assumed data of the employer and competing contractors before the start of game 1 is as follows:

Table 3: Players Data – Model Verification

Project Direct Cost	1,125,000
Employer / Owner's Value	1,600,000
Contractor 1 Avg. Pricing Factor*	1.4
Contractor 1 Pricing Factor Std.**	0.01
Contractor 2 Avg. Pricing Factor	1.38
Contractor 2 Pricing Factor Std.	0.01

*Pricing Factor = Bid Price / Project Direct Cost ** Std.: Standard Deviation

After calculating the payoffs for the players of Game 1 (employer and the contractor with the higher bid price) using the equations discussed in chapter 3, Game 1 is modeled on Gambit and its Nash Equilibrium is computed as shown in Figure 21. The main output of this game is the payoff of the employer at equilibrium, which is equal to 40,750 (indicated with a red circle on the figure). Also, the Nash Equilibrium for Game 1 is the path "0% initial discount by Contractor – Rejection of offer by employer - 1% additional discount by the Contractor" as indicated in blue arrows on the figure.

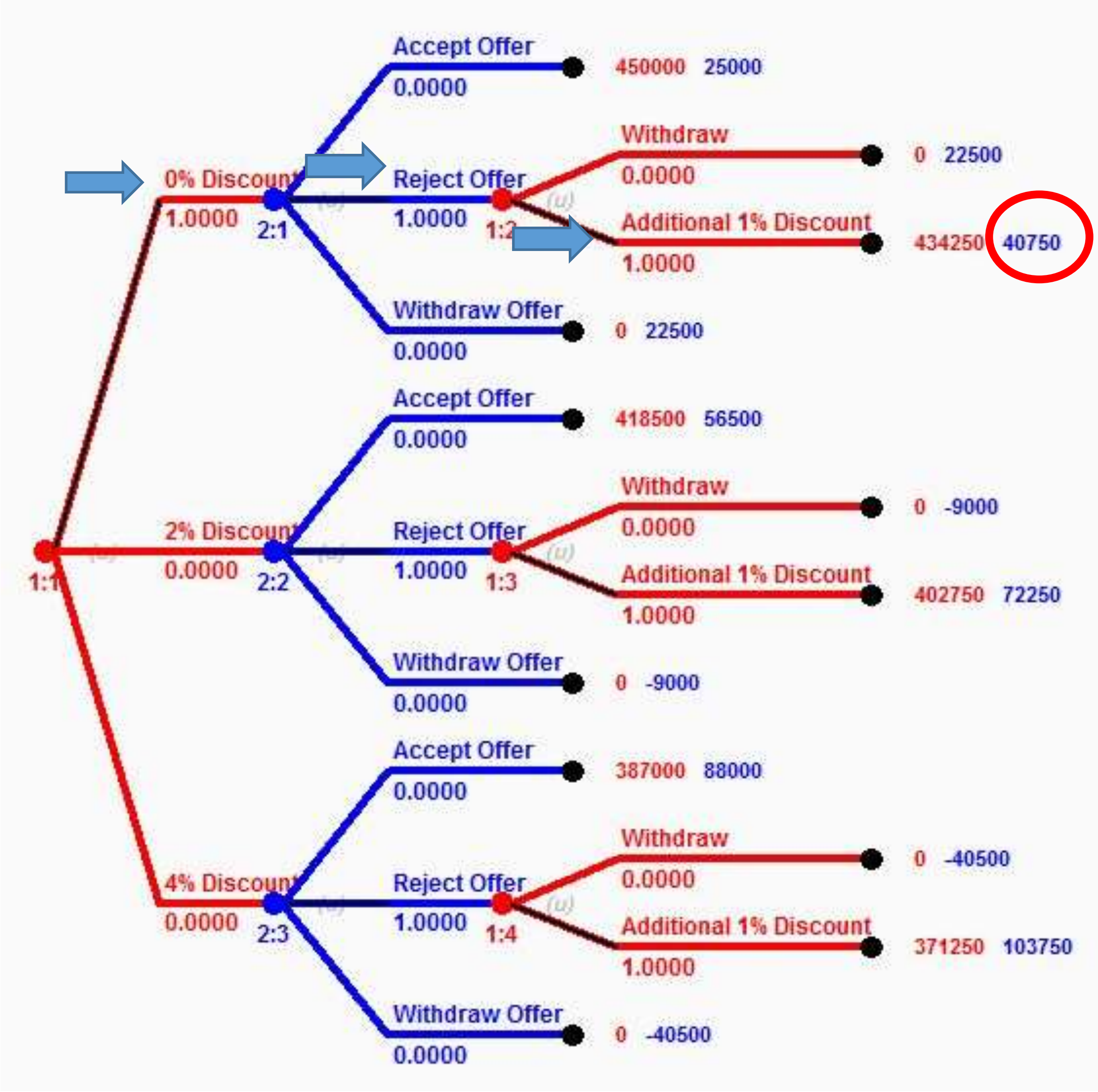


Figure 21: Game 1 Modeled on Gambit – Model Verification

4.3.2 Game 2 on Gambit:

The owner's value is the only value that changes based on the Nash Equilibrium of Game 1. As shown in Figure 21, the employer made a positive payoff of 40,750 after negotiating with the contractor having the higher bid price. This means that the owner's value for the project at start of Game 2 will be equal to the 1,600,000 – 40,750 = 1,559,250. The value for all other variables will remain the same. After calculating the payoffs for the players of Game 2 (employer and the contractor with the lower bid price) using the equations discussed in Chapter 3, Game 2 is modeled on Gambit and its Nash Equilibrium is computed as shown in Figure 22.

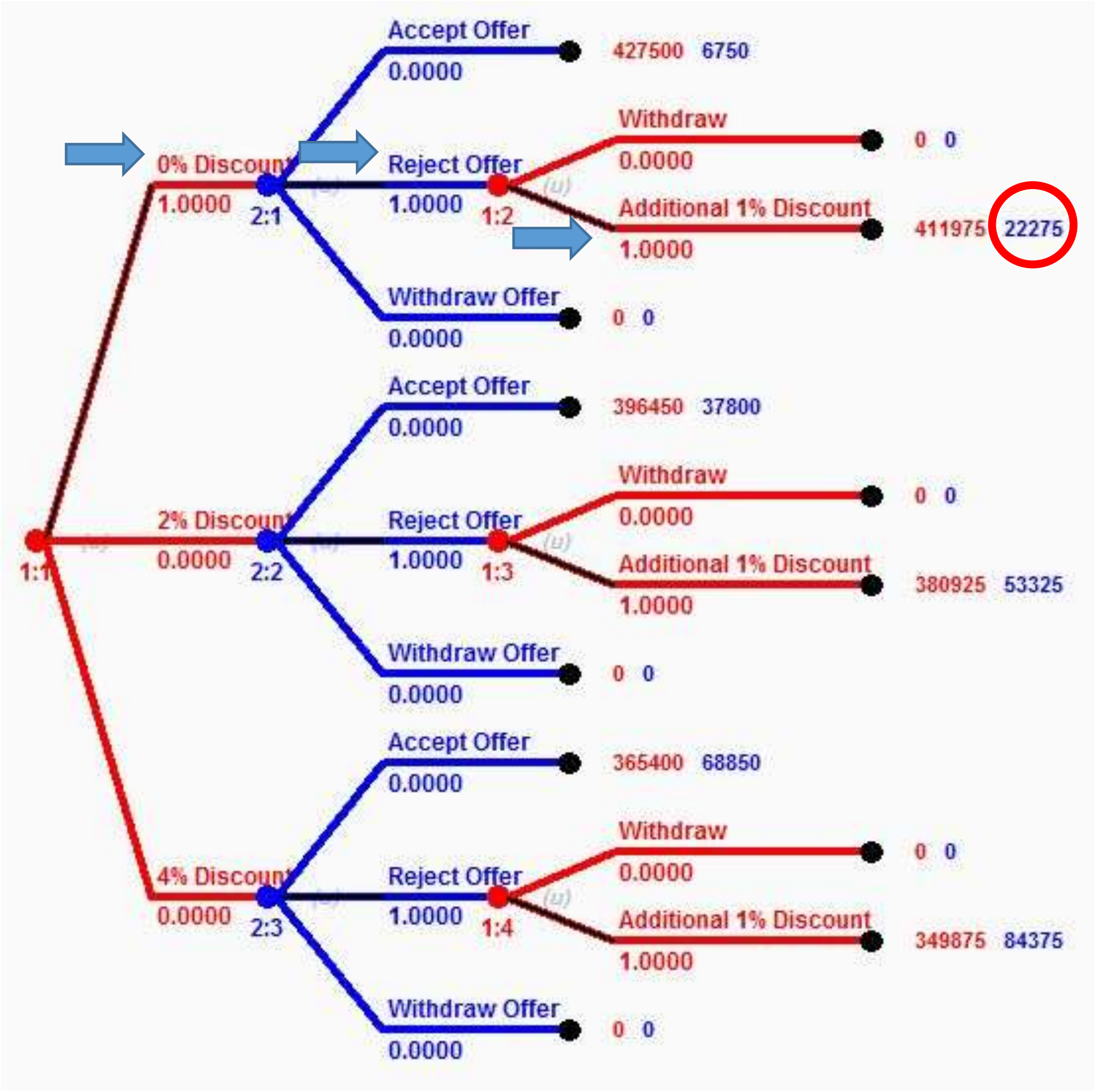
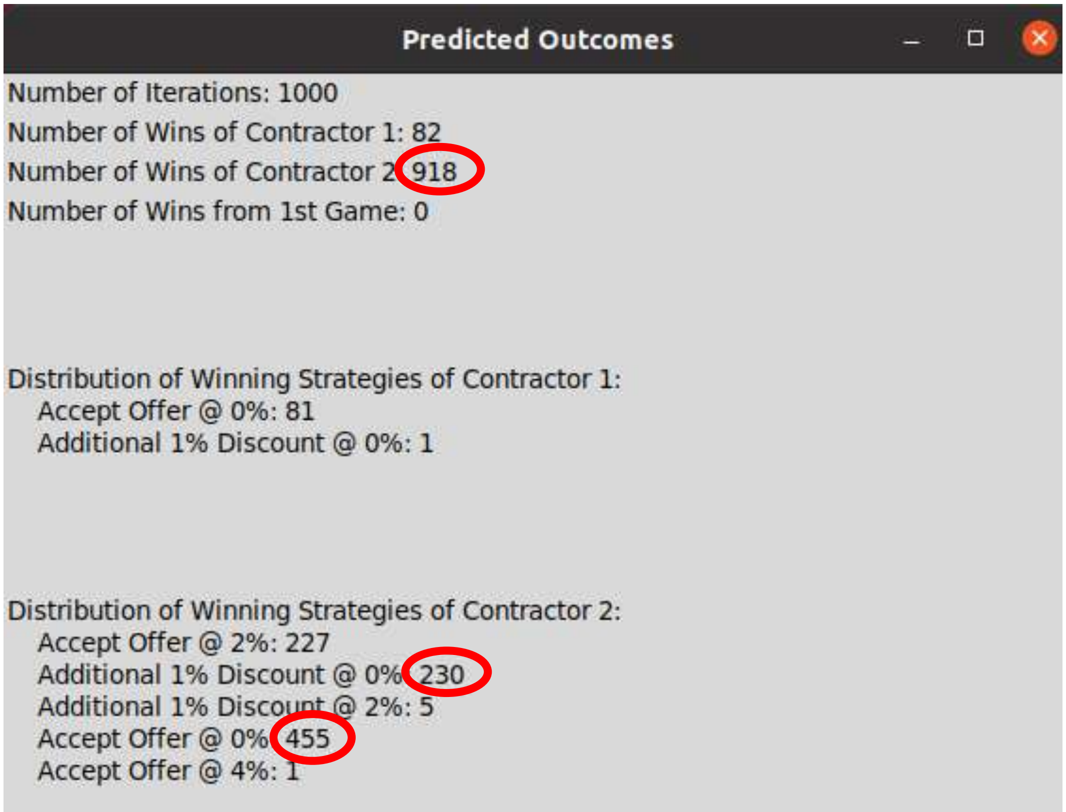


Figure 22: Game 2 Modeled on Gambit – Model Verification

The positive payoff (22,275) of the employer at the Equilibrium of Game 2 means that the employer succeeded in reaching a lower price for his project (lower owner’s value) after negotiating with the second contractor. This positive payoff means that the contractor who played Game 1 with the employer has lost the project as the employer was able to secure a less bid price from Game 2. Thus, the main output of Game 2 is the payoff of the employer at equilibrium, and which determines the wining contractor. Also, the Nash Equilibrium for Game 2 is the path “0% initial discount by Contractor – Rejection of offer by employer - 1% additional discount by the Contractor” as indicated in blue arrows on the figure. In this case, the wining contractor is the one who played Game 2 (contractor 2) and this path is thus the strategy that led the contractor to win the bid and it represents the strategies that the employer and the winning contractor have no incentive to deviate from (assuming that both of them are rational players).

4.3.3 Python Model Output for the same Scenario:

The following figure shows the outputs of the python model after inserting to it the same data of Table 3:



```
Number of Iterations: 1000
Number of Wins of Contractor 1: 82
Number of Wins of Contractor 2: 918
Number of Wins from 1st Game: 0



Distribution of Winning Strategies of Contractor 1:
   Accept Offer @ 0%: 81
   Additional 1% Discount @ 0%: 1



Distribution of Winning Strategies of Contractor 2:
   Accept Offer @ 2%: 227
   Additional 1% Discount @ 0%: 230
   Additional 1% Discount @ 2%: 5
   Accept Offer @ 0%: 455
   Accept Offer @ 4%: 1
```

Figure 23: Python Model Predictions – Model Verification

The results of the model are in line with the results obtained from the manual trial on Gambit: out of the 1000 trials, the contractor with the lower bid price (contractor 2) won 918 times and which matches the results obtained from Gambit. However, the python model which simulates the negotiations situations 1000 times showed that the path “0% initial discount by Contractor – Rejection of offer by employer - 1% additional discount by the Contractor” will not always be the path that will lead contractor 2 to win the bid. In fact, contractor 2 won 455 out of the 918 times without offering any discounts to the employer while he won 230 out of the 918 times only with the strategy obtained from the single trial conducted on Gambit. Moreover, the obtained results showed that the contractor who played Game 1 with the employer has never won the bid in the conducted 1000 simulations.

Moreover, the verification example shows that conducting 1 trial only on Gambit is not sufficient for the user to make a decision regarding his negotiation strategy: it was shown that the strategy that led the contractor to win almost 50% of the times is different than the strategy obtained from the single trial conducted on Gambit.

4.4 Case Study – Hospital Project in Cairo:

4.4.1 Introduction:

The developed python model is validated through a real case study of a hospital project in Cairo. Gathering the required data to simulate the final negotiations stage of this project was challenging, as it required obtaining confidential data from the 2 competing contractors who reached the final negotiations stage as well as obtaining confidential data regarding this hospital project itself. The two competing contractors are first category general contracting companies in Egypt and will be referred to them as Contractor 1 and Contractor 2 in this section. It is worth highlighting that the gathered information regarding the bidding behavior of these contractors were obtained from the contractors themselves, thus the data inserted to the model may have a higher level of accuracy than the data that will be inserted by the user as he may not be fully aware of his competitor's bidding behavior (especially if he has limited data regarding the past projects in which his competitor participated in). Also, it is worth mentioning that the bidding for this hospital project was a closed bid; thus no contractor would know the bid price offered by his competitor.

4.4.2 Bidding Behavior – Contractor 1:

The following table is an extraction of the data provided by Contractor 1 for some of the projects he tendered for in the period 2015 – 2020 (Please refer to **Appendix E** for the whole data provided by Contractor 1):

Table 4: Extraction of Contractor 1 Projects Data

#	Project Type	Direct Cost (EGP)	Bid Price (EGP)	Project Awarded (Y/N)	Pricing Factor
1	Sports Club	15,942,029	22,000,000	Y	1.38
2	Residential Buildings	1,566,666,667	2,115,000,000	Y	1.35
3	Commercial Mall	384,615,385	550,000,000	N	1.43
..	………….	……………..	………….	…….	……….
20	Residential Buildings	518,461,538	741,400,000	Y	1.43

Contractor 1 provided the data of 20 out of 40 projects he tendered for during the abovementioned time period (i.e. 50% of his projects were reported). The pricing factor for each of the 20 projects was computed and a normal Q-Q plot (Figure 24) of the computed pricing factors was plotted using R software to check the normality of the received data. The Q-Q plot shows that almost all data points are falling on the red line and thus the distribution of the pricing factor of Contractor 1 could be considered as normally distributed.

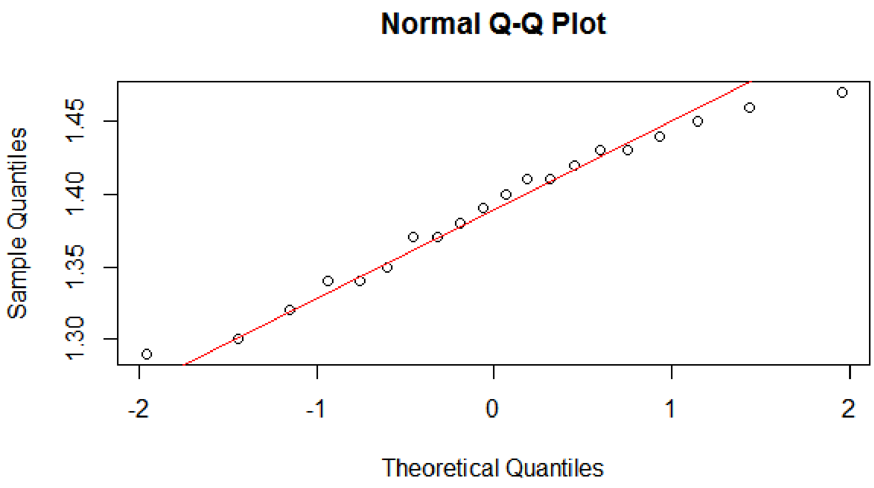


Figure 24: Normal Q-Q Plot - Contractor 1

Accordingly, the computed average pricing factor and standard deviation of the pricing factor for Contractor 1 are as follows:

Pricing Factor Average =	***1.389***
Pricing Factor Std =	***0.0527***

4.4.3 Bidding Behavior – Contractor 2:

The following table is an extraction of the data provided by Contractor 2 for some of the projects he tendered for in the period 2015 – 2021 (Please refer to **Appendix E** for the whole data provided by Contractor 2):

Table 5: Extraction of Contractor 2 Projects Data

#	Project Type	Direct Cost (EGP)	Bid Price (EGP)	Project Awarded (Y/N)	Pricing Factor
1	Residential Buildings	897,000,000	1,246,830,000	Y	1.39
2	Residential Buildings	423,000,000	626,040,000	Y	1.48
3	Industrial Building	733,000,000	989,550,000	Y	1.35
..	………….	……………..	………….	…….	……….
21	Industrial Building	288,000,000	397,440,000	N	1.38

Contractor 2 provided the data of 21 out of 178 projects he tendered for during the abovementioned time period (i.e. 12% of his projects were reported). The pricing factor for each of the 21 projects was computed and a normal Q-Q plot (Figure 25) of the computed pricing factors was plotted using R software to check the normality of the received data. The Q-Q plot shows that the majority of the data points are falling on the red line and thus the distribution of the pricing factor of Contractor 2 could be considered as normally distributed.

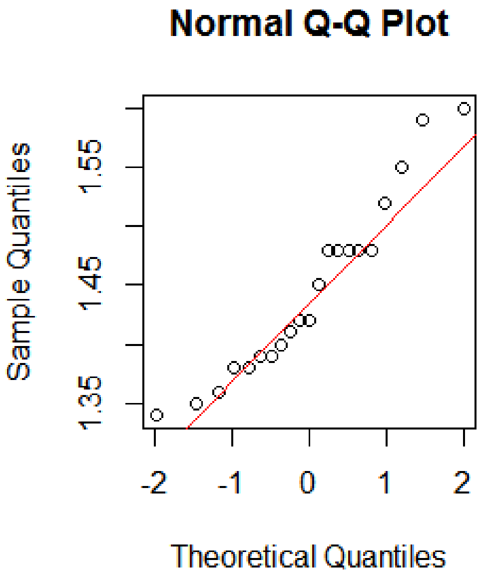


Figure 25: Normal Q-Q Plot - Contractor 2

Accordingly, the computed average pricing factor and standard deviation of the pricing factor for Contractor 2 are as follows:

Pricing Factor Average =	***1.445***
Pricing Factor Std =	***0.0755***

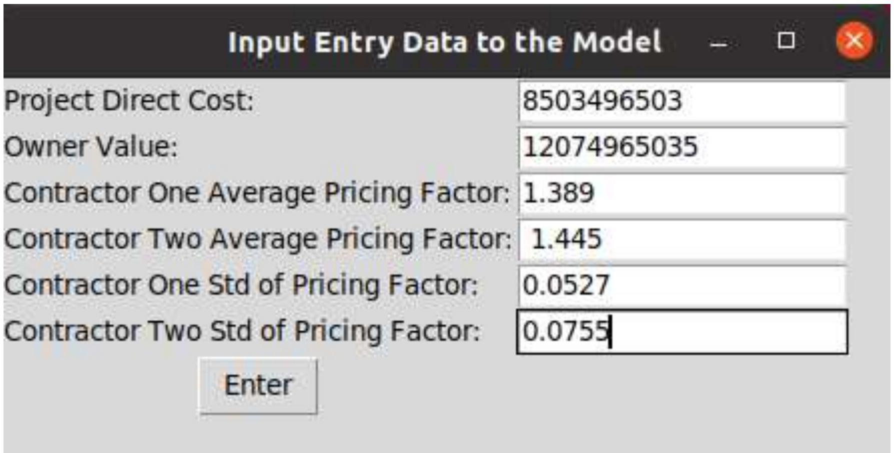
4.4.4 Project Direct Cost & Owner's Value:

The direct cost for the hospital project reported by Contractor 1 is **8,503,496,503 EGP.** Contractor 2 did not report the direct cost of the project considering this information as confidential. Thus, the value reported by Contractor 1 is the value to be inserted to the python model. In fact, as both contractors are Class A contractors, it can be assumed that they have the same technical knowledge, available information, and contacts of suppliers. Accordingly, both contractors are expected to have the same direct cost for the project.

Also, as it was difficult to obtain the Owner's Value for the project / fair price estimate from the employer or the consultant, and after discussions with certain senior cost engineers, it was decided to assume a pricing factor for this project of 1.42 and to multiply it by the direct cost reported by Contractor 1. Accordingly, the assumed Owner's Value for the hospital project is **12,074,965,035 EGP.**

4.4.5 Predictions of the Python Model:

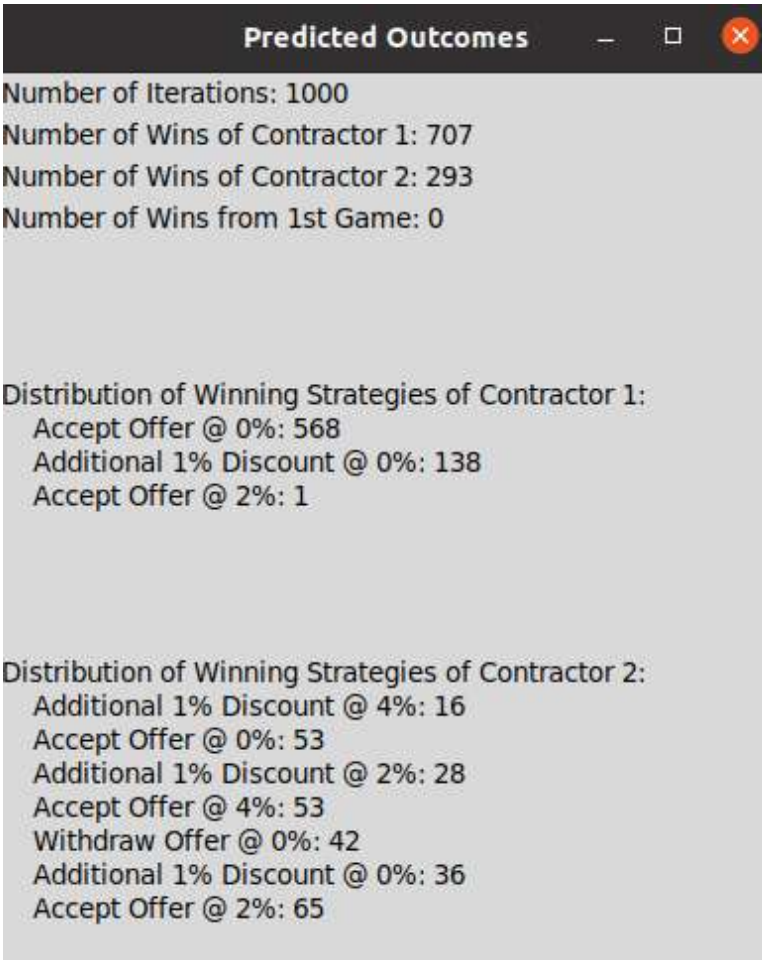
Based on the data collected from the 2 competing contractors, the required data was inserted to the developed python model as shown in Figure 26.



Project Direct Cost: 8503496503
Owner Value: 12074965035
Contractor One Average Pricing Factor: 1.389
Contractor Two Average Pricing Factor: 1.445
Contractor One Std of Pricing Factor: 0.0527
Contractor Two Std of Pricing Factor: 0.0755

Figure 26: Input Data - Validation Case Study

Figure 27 shows the simulation results for the hospital project negotiations stage based on the inserted project data and the bidding behavior of the 2 contractors.



```
Number of Iterations: 1000
Number of Wins of Contractor 1: 707
Number of Wins of Contractor 2: 293
Number of Wins from 1st Game: 0



Distribution of Winning Strategies of Contractor 1:
   Accept Offer @ 0%: 568
   Additional 1% Discount @ 0%: 138
   Accept Offer @ 2%: 1



Distribution of Winning Strategies of Contractor 2:
   Additional 1% Discount @ 4%: 16
   Accept Offer @ 0%: 53
   Additional 1% Discount @ 2%: 28
   Accept Offer @ 4%: 53
   Withdraw Offer @ 0%: 42
   Additional 1% Discount @ 0%: 36
   Accept Offer @ 2%: 65
```

Figure 27: Predicted Outcomes - Validation Case Study

The findings of the model can be summarized as follows:

- Contractor 1 has higher probability of wining: The model predicted that according to the bidding behavior of the contractors, Contractor 1 has a 70.7% chance of winning the project while his competitor Contractor 2 has only a 29.3% chance of winning.
- Contractor having the lower initial bid price always wins: The model predicted that the contractor having the higher initial bid price (who plays Game 1 with the employer) will have no chance of winning the project since the number of wins obtained from Game 1 is equal to 0. Accordingly, Contractor 2 who has a higher average pricing factor compared

to Contractor 1, has to ensure that his original offer is lower than the offer of his competitor in order to have a chance to win the project.

- Contractor 1 can make a maximum discount of 1% and wins the project: The model predicted that 706 out of the 707 wins made by Contractor 1 required him to offer the employer a maximum discount of 1%. Thus, as a rational player, Contractor 1 is expected to offer around 1% discount to the employer to win this project.
- Contractor 2 can make a discount of 2% and wins: The model predicted that 61% of the times Contractor 2 won, he was required to offer 0%-2% additional discount to the employer. This is reasonable because Contractor 2 will not win the project unless he offers a lower initial bid price than Contractor 1, and knowing that his competitor will make a maximum discount of 1%, Contractor 2 shall not offer more than 2% discount to the employer to decrease the possibility of leaving money on the table in case he wins the project.
- Contractor 1 has always the upper hand: As Contractor 1 has a lower average pricing factor compared to Contractor 2, he has the upper hand during the negotiations as he knows that he has a very high chance of winning the project and by offering a minimal additional discount to the employer. This may pressure Contractor 2 to offer unnecessary additional discount percentage to win the project in case he does not analyze the negotiations situation rationally.

4.4.6 Comparing the Predictions of the Python Model to the Actual Scenario:

The following table shows the actual initial bid prices offered by Contractor 1 and Contractor 2 as well as their final bid prices after negotiations. These figures were obtained from the contractors themselves.

Table 6: Actual Figures - Validation Case Study

Contractor	Initial Bid Price (EGP)	Final Bid Price after Negotiations (EGP)	% Discount Offered	Awarded (Y/N)
Contractor 1	12,330,069,930	12,160,000,000	1.38%	N
Contractor 2	12,093,332,587	11,552,000,000	4.48%	Y

The comparison between the predictions of the model and the actual scenario can be summarized as follows:

- Contractor having the lower initial bid price won the project: As predicted by the model, Contractor 2 having the higher pricing factor average, had to offer an initial bid price lower than his competitor in order to win the project. That's why Contractor 2 used a pricing factor of 1.42 in this project (lower than his average of 1.445) to increase his chances of winning this project.
- Contractor 1 knows that he has the upper hand: As Contractor 1 has a lower average pricing factor compared to Contractor 2, he understands that he has the upper hand during the negotiations, that's why he used a pricing factor of 1.45 (higher than his average of 1.389 and almost equal to his competitor's average).
- Contractor 1 offered around 1% discount only: As predicted by the model, Contractor 1 being a rational player, knows that he has to offer a maximum of 1% discount to the employer. In order to guarantee winning the project, Contractor 1 offered an additional 0.38% discount to the employer.
- Contractor 2 offered a discount higher than the 2%: Contractor 2 offered a discount of 4.48% to the employer, which is higher than the 2% suggested by the python model. As a result of this irrationality from Contractor 2's side (due to the pressure from Contractor

1), he won the project as predicted by the model (as he offered a lower initial bid) but he left 2.28% of money on the table (which is equivalent to 299,465,935 EGP). In case Contractor 2 used the developed model/ decision support tool, he would have saved the 2.28% of unnecessary discount offered to the employer.

Based on the above, it can be concluded that the predictions of the developed python model are in line with the actual outcomes of the negotiations stage for the hospital project. Thus, the developed model can be considered as a reliable decision support tool for contractors to use during negotiations stages.

Chapter 5: Conclusion and Recommendations

5.1 Research Overview:

The project owner usually enters into a negotiations stage with the contractors with the lowest financial offers before selecting the winning contractor. The project owner will try during these negotiations to pressure the contractors to lower their prices in order to get the best price possible for his project. Consequently, the contractor who reaches this negotiations stage is faced by the dilemma of the minimum discount percentage he may need to offer to the employer that maximizes his chance to win the project.

In fact, the negotiation stage between the project owner and the contractors can be seen as a game and the theory that best analyzes and explains this game and predicts its rational outcome is called "Bargaining Game Theory". Unfortunately, all the research done using Bargaining Game Theory (in the construction field) to date, address issues related mainly to PPP and BOT projects only with limited to no research has been conducted on other contract types. Also, limited to no research related to the application of bargaining theory in the Egyptian construction industry was conducted.

The main objective of this research is to develop a decision support tool / model that can be used by contractors who bid for private projects, where negotiations are allowed, contractors are procured through competitive bidding, and the low bid method is used for awarding the contracts. This research presents the framework for a decision support tool / model that uses bargaining game theory to help contractors make rational decisions regarding the discount percentage to offer to the employer during negotiations in order to establish a win-win scenario in which the employer gets the lowest possible price for his project and at the same time the contractor’s profit is maximized.

In phase one of this research, a literature review is conducted to explore the history and important concepts of Game Theory and its branch Bargaining Theory. This part of the research also includes a summary of the applications of Game Theory in the construction industry as well as the applications of Bargaining Game Theory specifically. In phase two, a brainstorming session is conducted with expert engineers in the Egyptian market to determine the factors that affect the bargaining power of the negotiating parties (employer/contractor) in a tender for a certain construction project. The brainstorming session led to the determination of 35 factors that were ranked based on their significance by surveying industry professionals.

In phase three, the parameters of the game representing the situation at the end stage of the negotiations where the employer is negotiating with the two contractors with the lowest bid prices are defined and the game is modeled on Gambit considering the highest ranked factor obtained from the survey results. Lastly in phase four, a python model is developed to simulate the games between the employer and the contractors several times and make all the required computations automatically and efficiently. The python model can be used as a decision support tool by contractors / employers as it is user friendly and allows the user to get instantaneous analysis of the negotiations situation.

The developed python model provides the users with the following:

- A friendly graphical user interface to deal with, with no need to use the Gambit software nor to calculate the payoffs of the players manually.

- An effective decision support tool that processes the data inserted by the user and gives him an instantaneous analysis of the negotiations situation. It provides the user with his probability of winning and the strategies that led to these wins. It also provides him with the same information for his competitor.
- Based on the analysis provided by the model, the user can make a rational decision regarding the percentage of additional discount to offer to the employer in a way to increase his chances of winning the bid and at the same time to minimize the amount of unnecessary discount offered to the employer in case he is awarded the project.

5.2 Research Contribution:

This research made several contributions to the field of bid price estimation for negotiated contracts through its adopted methodology and attained results. These contributions are:

1. Extending the use of Bargaining Game Theory in the construction field, to address issues related to contract types other than the PPP contracts.
2. Using Bargaining Game Theory to analyze the negotiations stage for a hospital mega project in Egypt and encouraging researchers in Egypt to conduct more research in this promising field.
3. Determining the factors that affect the bargaining power of the negotiating parties (employer/contractor) in a tender for a certain construction project and ranking them based on their significance by surveying Egyptian industry professionals. Using the brainstorming technique, the identified factors represent an up-to-date assessment of the Egyptian market. The identified list of factors could be used as a reference by researchers and could be re-evaluated with time to know how the Egyptian market changes.

Moreover, the identified list of factors could be used as a template by researchers in other countries to evaluate the construction market situation in their countries.

4. Developing an effective decision support tool that helps contractors make rational decisions regarding the percentage of additional discount to offer to the employer during negotiations in a way to increase their chances of winning and maximize their profits.
5. The analysis of the hospital project case study showed that using the developed model by the winning contractor could have saved him 299,465,935 EGP (equivalent to 2.28% of his bid price) that were left on the table.
6. The developed model has overcome certain drawbacks of the Gambit software by providing a friendly graphical user interface to the users with no need to calculate the payoffs of the players manually and with conducting 1000 simulations of the negotiations situation and providing the summarized results instantaneously.
7. The developed python code is attached in the appendices for researchers to use freely and develop it to be more accurate and precise.

5.3 Encountered Challenges:

Several challenges were encountered during the different phases of this research. The main challenges were as follows:

1. The Gambit documentation available online is not trivial and the Gambit-python package has to be built from the software source code itself as it is not pip-installable. Also, the used source code to build the Gambit library contained several bugs and additional effort was necessary to work around these bugs.

2. The latest stable version of the Gambit library was written for an old version of python (version 2.7) which requires installing specific prerequisites packages compatible with this old version of python.
3. Building the Gambit-python package on Windows failed several times and after conducting a research on the internet, it was found that Gambit does not support building the python extension on Windows, which was never mentioned in the documentation. In order to overcome this problem, using the Ubuntu operating system was required.
4. The Gambit development team has stopped working on the software since January 2019, which represented a great risk of reaching a dead-end during the development of the python model.
5. Obtaining data regarding the bidding behavior of contractors to validate the model was a great challenge as contractors considered the requested type of data to be confidential.

5.4 Recommendations for Future Research:

The following recommendations are to be considered in future research:

1. The developed model was based on the highest ranked factor affecting the bargaining power of the negotiating parties (employer/contractor) only. Future research may consider integrating more factors in the model to make its predictions more accurate and precise.
2. The developed model considers only 1 level for the chosen factor affecting the bargaining power of the negotiating parties. The possible discount percentages to be offered by the contractors are based on an “average” level of the chosen factor. Future research may consider adjusting these discount percentages based on other levels for the chosen factor.

3. Future research can build on the developed python code in this research in order to avoid some of the encountered challenges related to building the Gambit-python package.
4. For validation purposes, it is recommended to collect data from contractors who competed against each other in several projects in the negotiations stage.
5. The developed model considered 3 phases only for the negotiation game between the employer and the contractor. Future research may consider dividing the 3rd phase of the game into 3-4 phases. However, this will result in having a more complex Gambit tree and high computational power will be required to compute the equilibrium of the games.
6. Future research may extend the use of the developed model to other project types as well as to consider other types of bidding strategies.

References:

[1]- Ahmed, Muaz O., et al. "Construction bidding and the winner's curse: Game theory approach." *Journal of Construction Engineering and Management* 142.2 (2016): 04015076.

[2]- American Chamber of Commerce in Egypt. *Covid-19's Impact: An Overview of Sectors at Risk*. April 2020. Retrieved October 30, 2021, from https://www.amcham.org.eg/information-resources/covid-19-response/covid-19-impact-an-overview-of-sectors-at-risk.

[3]- Arab Contractors. *New El Alamein City: The Arab Contractors*. Web. 29 Aug. 2020. <www.arabcont.com/English/project-553>.

[4]- Asgari, Sadegh, Abbas Afshar, and Kaveh Madani. "Cooperative game theoretic framework for joint resource management in construction." *Journal of Construction Engineering and Management* 140.3 (2014): 04013066.

[5]- Asgari, Sadegh, Amr Kandil, and Ibrahim Odeh. "Optimal risk attitude for construction contractors in competitive bidding environments." *Construction Research Congress 2016*. 2016.

[6]- Bayat, Morteza, Mostafa Khanzadi, and Farnad Nasirzadeh. "Bargaining Game Model to Determine Concessionary Items in Build-Operate-Transfer Contracts." *Journal of Construction Engineering and Management* 146.2 (2020): 04019109.

[7]- Bhuiyan, Bellal. "An overview of game theory and some applications." *Philosophy and Progress* (2016): 111-128.

[8]- Carraro, Carlo, Carmen Marchiori, and Alessandra Sgobbi. "Negotiating on water: insights from non-cooperative bargaining theory." *Environment and Development Economics* 12.2 (2007): 329-349.

[9]- Cochran, William Gemmell. Sampling Techniques: 3d Ed. Wiley, 1977.

[10]- Dyer, Douglas, John H. Kagel, and Dan Levin. "A comparison of naive and experienced bidders in common value offer auctions: A laboratory analysis." *The Economic Journal* 99.394 (1989): 108-115.

[11]- Dyer, Douglas, and John H. Kagel. "13 Bidding in Common Value Auctions: How the Commercial Construction Industry Corrects for the Winner’s Curse." Common Value Auctions and the Winner's Curse. *Princeton University Press*, 2009. 349-369.

[12]- Dzeng, Ren-Jye, and Yu-Chun Lin. "Intelligent agents for supporting construction procurement negotiation." *Expert Systems with Applications* 27.1 (2004): 107-119.

[13]- El Asmar, Mounir, Awad S. Hanna, and Chul-Ki Chang. "Monte Carlo simulation approach to support alliance team selection." *Journal of Construction Engineering and Management* 135.10 (2009): 1087-1095.

[14]- Faghih, Aryo, and Reza Akhavian. "A Game-Theory Approach to Construction Dispute Resolution through Mediation." *Journal of Legal Affairs and Dispute Resolution in Engineering and Construction* 11.4 (2019): 05019004.

[15]- Forgó, Ferenc, János Fülöp, and Mária Prill. "Game theoretic models for climate change negotiations." *European Journal of Operational Research* 160.1 (2005): 252-267.

[16]- Gurmu, Argaw Tarekegn, and Citra S. Ongkowijoyo. "Stochastic-based model for setting formwork-productivity baseline." *Journal of Construction Engineering and Management* 146.9 (2020): 04020099.

[17]- Hibberts, Mary, R. Burke Johnson, and Kenneth Hudson. "Common survey sampling techniques." *Handbook of survey methodology for the social sciences*. Springer, New York, NY, 2012. 53-74.

[18]- Ho, S. Ping, and Liang Y. Liu. "Analytical model for analyzing construction claims and opportunistic bidding." *Journal of construction engineering and management* 130.1 (2004): 94-104.

[19]- Hsueh, Sung-Lin, and Min-Ren Yan. "Contribution-based profit-sharing scheme for joint ventures." *Technological and Economic Development of Economy* 17.3 (2011): 445-458.

[20]- Hurtado, Carlos. *Introduction to Game Theory*. University of Illinois at Urbana-Champaign, 2015, www.econ.uiuc.edu/~hrtdmrt2/Teaching/GT_2015_19/L1.pdf.

[21]- Jin, Hongyu, et al. "Imperfect Information Bargaining Model for Determining Concession Period of PPPs under Revenue Uncertainty." *Journal of Legal Affairs and Dispute Resolution in Engineering and Construction* 12.2 (2020): 04520012.

[22]- Kelly, Morgan. “A 'Long Awaited Recognition': Nash Receives Abel Prize for Revered Work in Mathematics.” Princeton University, The Trustees of Princeton University, 26 Mar. 2015, https://www.princeton.edu/news/2015/03/26/long-awaited-recognition-nash-receives-abel-prize-revered-work-mathematics.

[23]- Li, Yan, Xinyu Wang, and Yahui Wang. "Using bargaining game theory for risk allocation of public-private partnership projects: Insights from different alternating offer sequences of participants." *Journal of Construction Engineering and Management* 143.3 (2017): 04016102.

[24]- McKelvey, Richard D., McLennan, Andrew M., and Turocy, Theodore L. (2015). Gambit: Software Tools for Game Theory, Version 15.1.1. http://www.gambit-project.org.

[25]- Muthoo, Abhinay. *Bargaining theory with applications*. Cambridge University Press, 1999: 10-89

[26]- Najera, Jesus. “Game Theory-History & Overview.” *Medium*, Towards Data Science, 12 Sept. 2019, towardsdatascience.com/game-theory-history-overview-5475e527cb82.

[27]- Nash, John F. "Equilibrium points in n-person games." *Proceedings of the national academy of sciences* 36.1 (1950): 48-49.

[28]- Nash Jr, John F. "The bargaining problem." *Econometrica: Journal of the econometric society* (1950): 155-162.

[29]- Nash, John. "Non-cooperative games." *Annals of mathematics* (1951): 286-295.

[30]- Nash, John. "Two-person cooperative games." *Econometrica: Journal of the Econometric Society* (1953): 128-140.

[31]- Powell, Robert. "Bargaining theory and international conflict." *Annual Review of Political Science* 5.1 (2002): 1-30.

[32]- Raoufi, Mohammad, and Aminah Robinson Fayek. "Fuzzy Monte Carlo agent-based simulation of construction crew performance." *Journal of Construction Engineering and Management* 146.5 (2020): 04020041.

[33]- Rasmusen, Eric. Games and Information: An Introduction to Game Theory. 4th ed., *Wiley-Blackwell*, 2006.

[34]- Rooke, John, David Seymour, and Richard Fellows. "The claims culture: a taxonomy of attitudes in the industry." *Construction Management & Economics* 21.2 (2003): 167-174.

[35]- Serrano, Roberto. "Fifty years of the Nash program." (1953): 2004-20.

[36]- Shen, L. Y., et al. "Using bargaining-game theory for negotiating concession period for BOT-type contract." *Journal of construction engineering and management* 133.5 (2007): 385-392.

[37]- State Information Service. "New Administrative Capital." *SIS*. Web. 29 Aug. 2020. <https://www.sis.gov.eg/section/352/5238?lang=en-us>.

[38]- Turocy, Theodore L. "Texas a&m university." *Bernhard von Stengel, London School of Economics "Game Theory" CDAM Research Report (October 2001)* (2001).

[39]- Unsal, Hakan I., and John E. Taylor. "Modeling interfirm dependency: Game theoretic simulation to examine the holdup problem in project networks." *Journal of Construction Engineering and Management* 137.4 (2011): 284-293.

[40]- Walker, Paul. "An Outline of The History Of Game Theory." *History of Game Theory*, 1 Apr. 1995, <www.euler.fd.cvut.cz/predmety/teorie_her/histf.html>.

[41]- Yu, Nanpeng, Leigh Tesfatsion, and Chen-Ching Liu. "Financial bilateral contract negotiation in wholesale electricity markets using Nash bargaining theory." *IEEE Transactions on Power Systems* 27.1 (2011): 251-267.

[42]- Gates, Marvin. "Bidding strategies and probabilities." *Journal of the Construction Division* 93.1 (1967): 75-110.

[43]- Elbashbishy TS, Hosny OA, Waly AF, Dorra EM. "Addressing cost overruns through risk path modeling." CSCE Annual Conference (2019).

Appendices

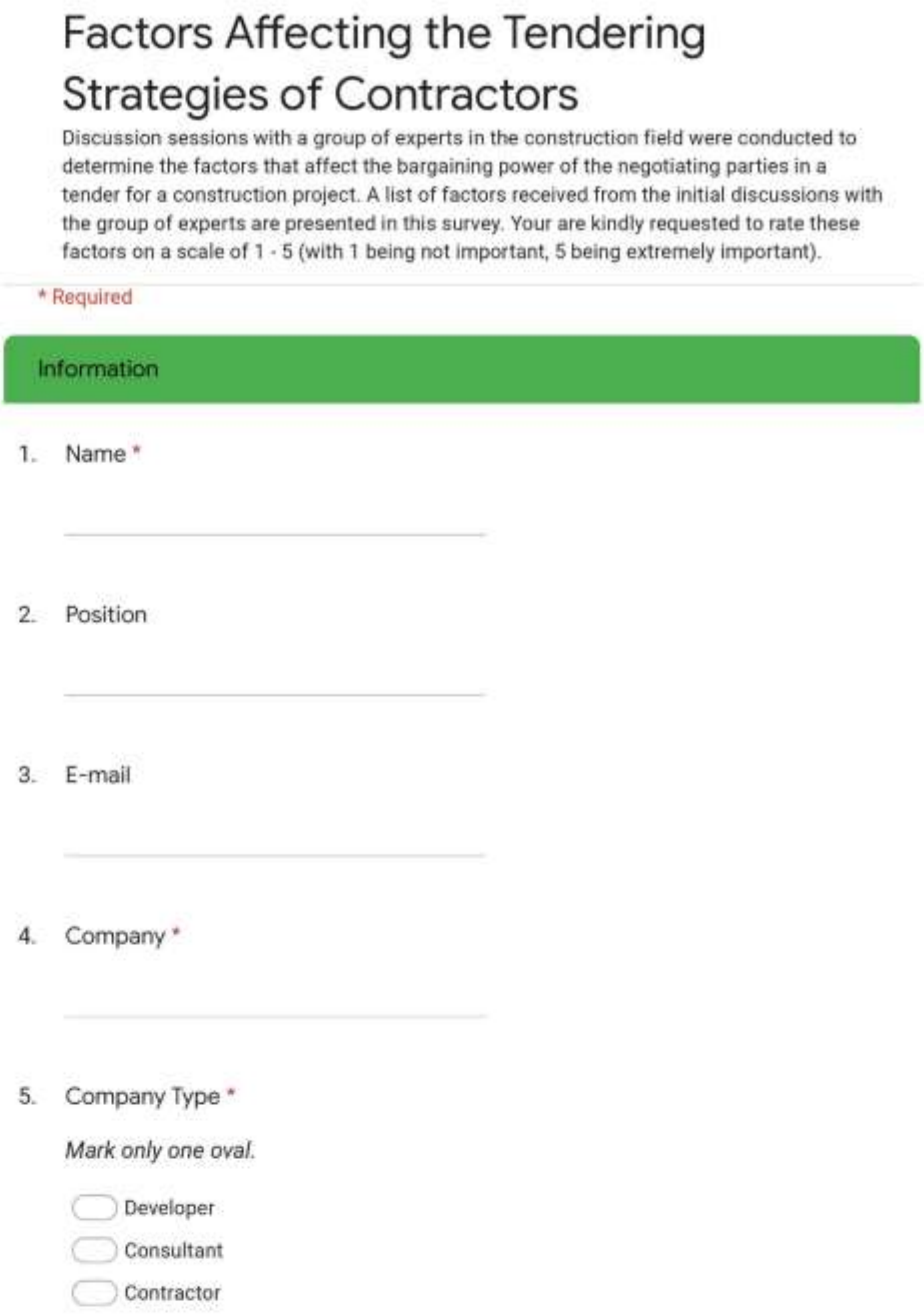
Appendix A: Brainstorming Session Experts:

#	Position	Years of Experience	Company Sector
1	Procurement Director	27	Developer
2	Project Manager	30	Consultant
3	Systems Developments Manager	15	Contractor
4	Senior Cost Engineer	12	Consultant
5	Lead Cost Control Engineer	11	Contractor
6	Senior Cost Control Engineer	10	Developer
7	Contracts Team Leader	10	Consultant

Appendix B: List of Factors Affecting the Bargaining Power of the Negotiating Parties:

#	Factor
1	Reputation of the Developer
2	Possible Support from Overseas Branches
3	Material Availability in the Market
4	Incompleteness of the Design Documents
5	Tender Scoring Criteria (% for Technical Offer VS % for Financial Offer)
6	Contractor Willingness to Work with a Specific Developer or Taking a Specific Project
7	The Employer Invites Small Contractors to Bid for the Project
8	Employer Offering a Shorter Payment Period and / or Payment for Materials On Site
9	Market and Economy Condition
10	Size of the Project compared to the Contractor's Company size
11	Contractor's need for Taking Projects
12	The Contractor Does Not Submit the Tender Bond
13	The Project is a Public Sector Project and the Employer is a Governmental Entity
14	The Submitted Tender Prices from All Contractors Are Higher than the Employer's Price Estimate of the Project
15	Competitors Participating in the Tender
16	The Final Number of Competitors in the Negotiations Stage
17	Tight Project Duration and Employer's Unwillingness to Increase the Project Duration
18	Location of the Project
19	Contractor's need for the project as a "reference"
20	Contractor Willingness to Keep Control on an Area Dominated by them
21	Contract Type: Lump-Sum or Re-measured
22	Reputation of the Engineer
23	Size VS Duration of the Project
24	There are other Packages to be Tendered in the Future by the Employer
25	Probability for Frequent Changes in Rules & Regulations
26	Opportunity to enter in a JV
27	The Contractor Having more Time to Understand the Project Compared to the Short Period Available for the Preparation of Tender
28	Risk Allocation in the Contract
29	When the Project is of a Special Type (Limited Number of Specialized Contractors to Choose From)
30	Availability of Clearer Information Regarding the Technical Documents (Specifications, Material Selection, etc.)
31	Size of the Contractor's Company
32	Financial Position of the Developer
33	Project Delivery Method
34	Transferring some of the Risks to the Employer (Materials Price Fluctuation, etc.)
35	Employer Willingness to Change the Specifications to Get Lower Prices

Appendix C: Sample Questionnaire:



Factors Affecting the Tendering Strategies of Contractors

Discussion sessions with a group of experts in the construction field were conducted to determine the factors that affect the bargaining power of the negotiating parties in a tender for a construction project. A list of factors received from the initial discussions with the group of experts are presented in this survey. Your are kindly requested to rate these factors on a scale of 1 - 5 (with 1 being not important, 5 being extremely important).

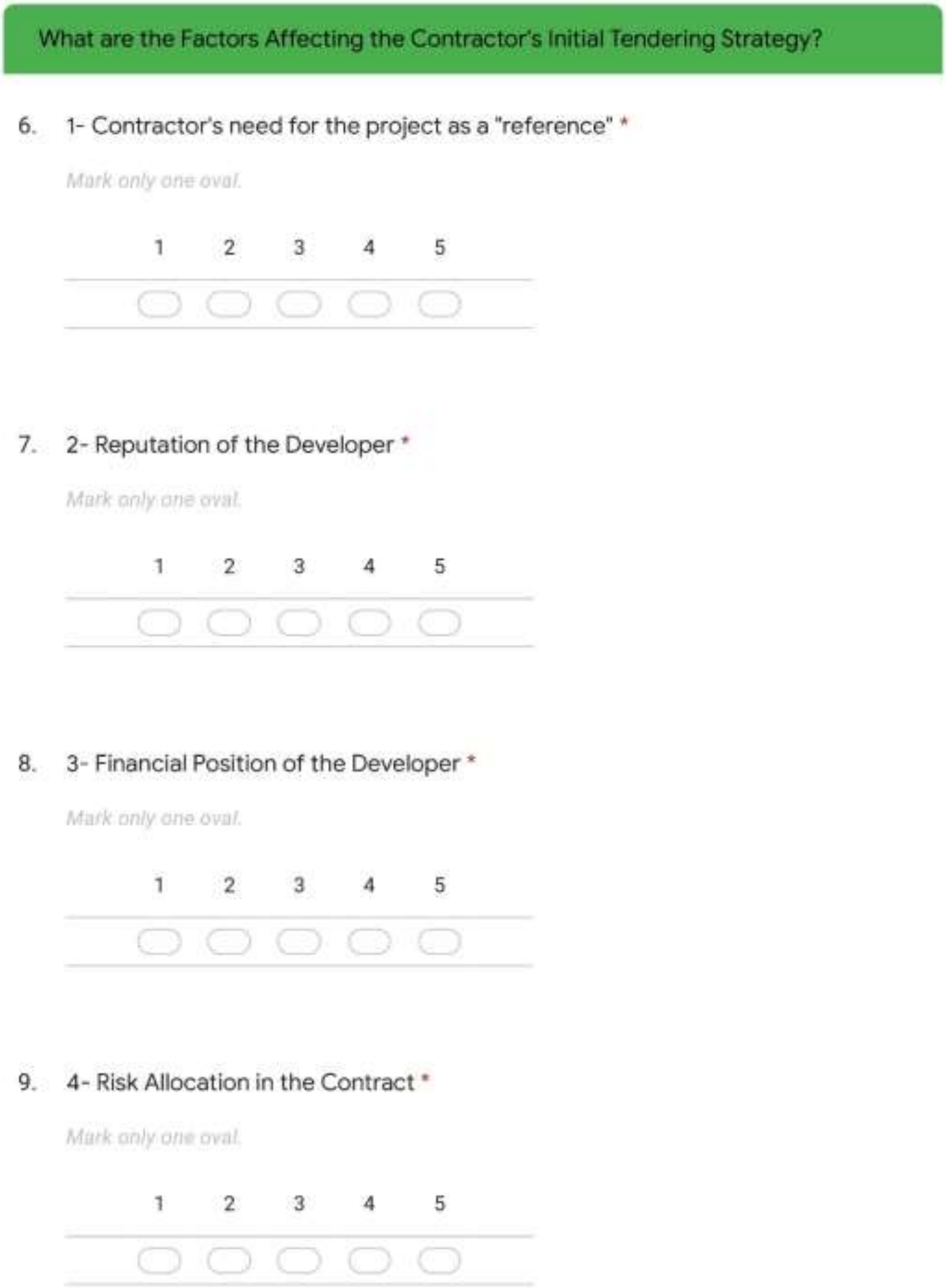
* Required

Information

1. Name *
2. Position
3. E-mail
4. Company *
5. Company Type *

Mark only one oval.

○ Developer
○ Consultant
○ Contractor



What are the Factors Affecting the Contractor's Initial Tendering Strategy?

6. 1- Contractor's need for the project as a "reference" *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

7. 2- Reputation of the Developer *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

8. 3- Financial Position of the Developer *

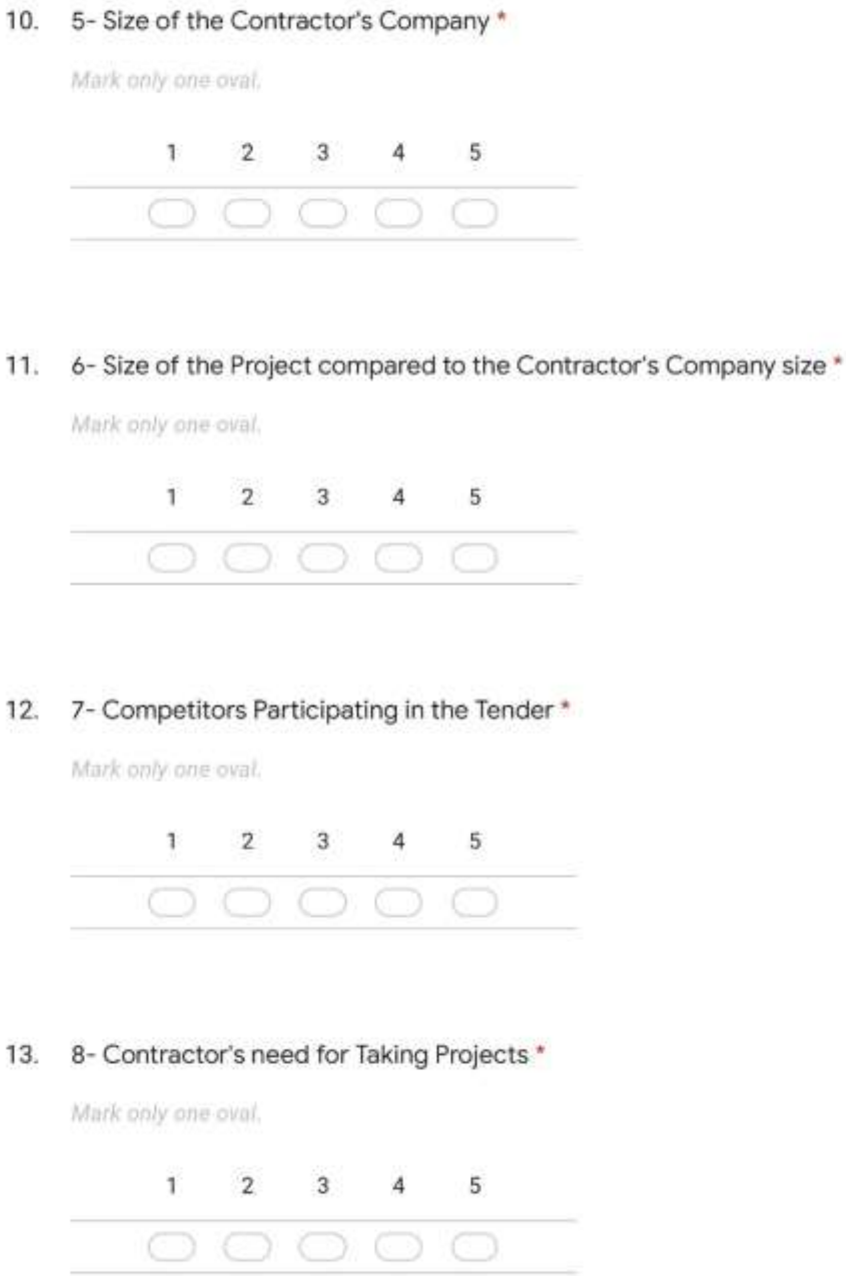
Mark only one oval.

1	2	3	4	5
○	○	○	○	○

9. 4- Risk Allocation in the Contract *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○



10. 5- Size of the Contractor's Company *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

11. 6- Size of the Project compared to the Contractor's Company size *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

12. 7- Competitors Participating in the Tender *

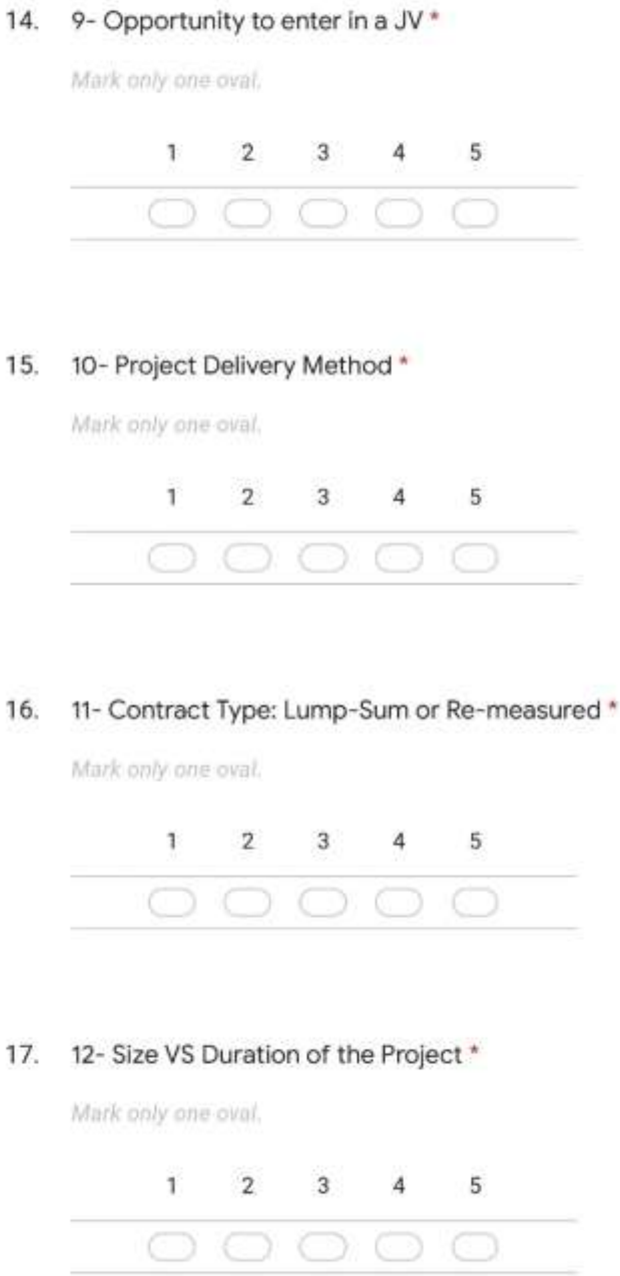
Mark only one oval.

1	2	3	4	5
○	○	○	○	○

13. 8- Contractor's need for Taking Projects *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○



14. 9- Opportunity to enter in a JV *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

15. 10- Project Delivery Method *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

16. 11- Contract Type: Lump-Sum or Re-measured *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

17. 12- Size VS Duration of the Project *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

18. 13- Reputation of the Engineer *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

19. 14- Market and Economy Condition *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

20. 15- Probability for Frequent Changes in Rules & Regulations *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

21. 16- Material Availability in the Market *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

22. 17- Tender Scoring Criteria (% for Technical Offer VS % for Financial Offer) *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

23. 18- Possible Support from Overseas Branches *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

24. 19- Location of the Project *

Mark only one oval.

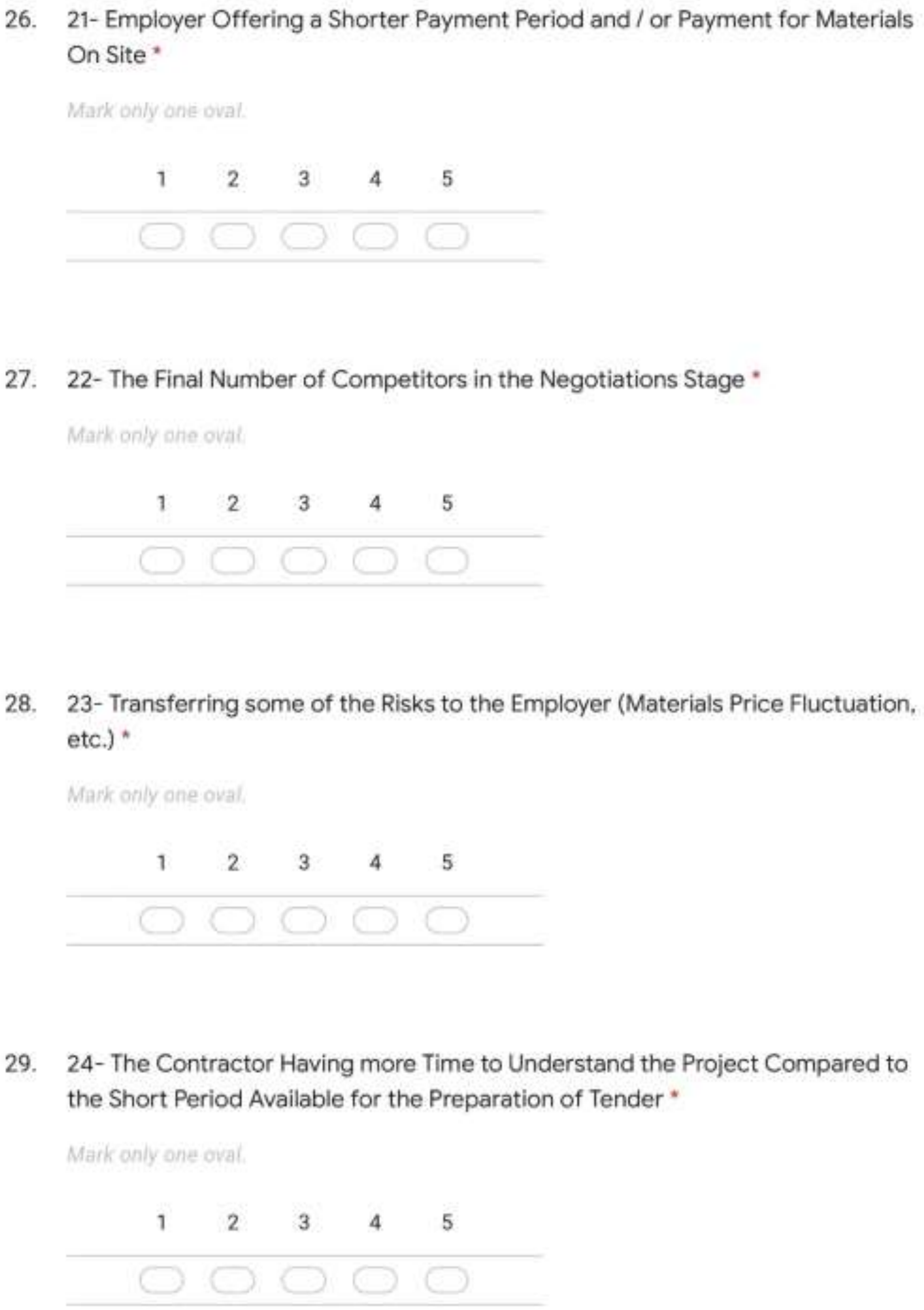
1	2	3	4	5
○	○	○	○	○

What are the Factors Affecting the Contractor's Willingness to Lower His Original Tender Price During the Negotiation Stage?

25. 20- Availability of Clearer Information Regarding the Technical Documents (Specifications, Material Selection, etc.). *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○



26. 21- Employer Offering a Shorter Payment Period and / or Payment for Materials On Site *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

27. 22- The Final Number of Competitors in the Negotiations Stage *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

28. 23- Transferring some of the Risks to the Employer (Materials Price Fluctuation, etc.) *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

29. 24- The Contractor Having more Time to Understand the Project Compared to the Short Period Available for the Preparation of Tender *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

30. 25- Employer Willingness to Change the Specifications to Get Lower Prices *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

31. 26- Contractor Willingness to Keep Control on an Area Dominated by them *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

32. 27- Contractor Willingness to Work with a Specific Developer or Taking a Specific Project *

Mark only one oval.

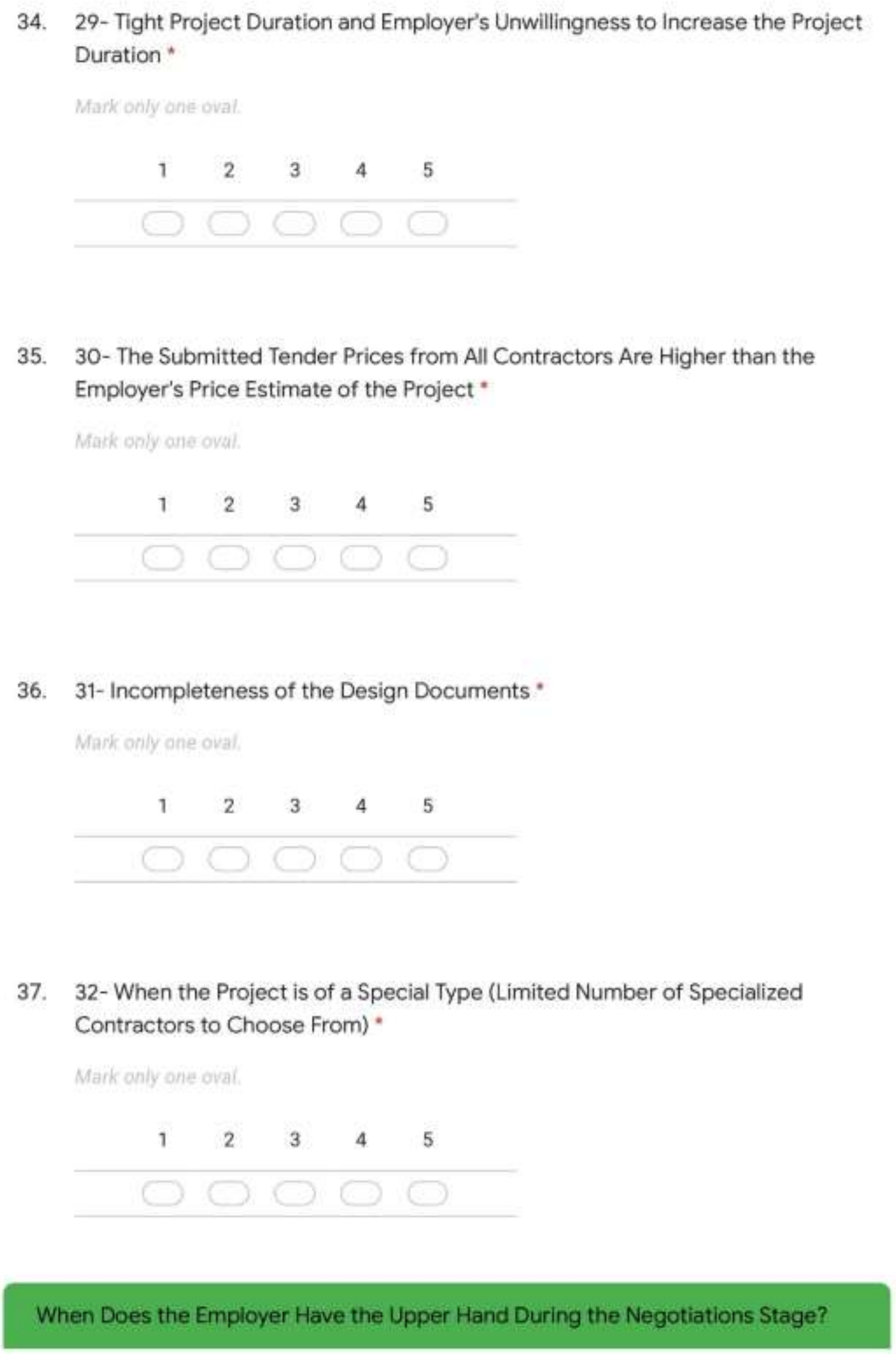
1	2	3	4	5
○	○	○	○	○

33. 28- Contractor Willingness to Work with a Specific Developer or Taking a Specific Project *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

When Does the Contractor Have the Upper Hand During the Negotiations Stage?



34. 29- Tight Project Duration and Employer's Unwillingness to Increase the Project Duration *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

35. 30- The Submitted Tender Prices from All Contractors Are Higher than the Employer's Price Estimate of the Project *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

36. 31- Incompleteness of the Design Documents *

Mark only one oval.

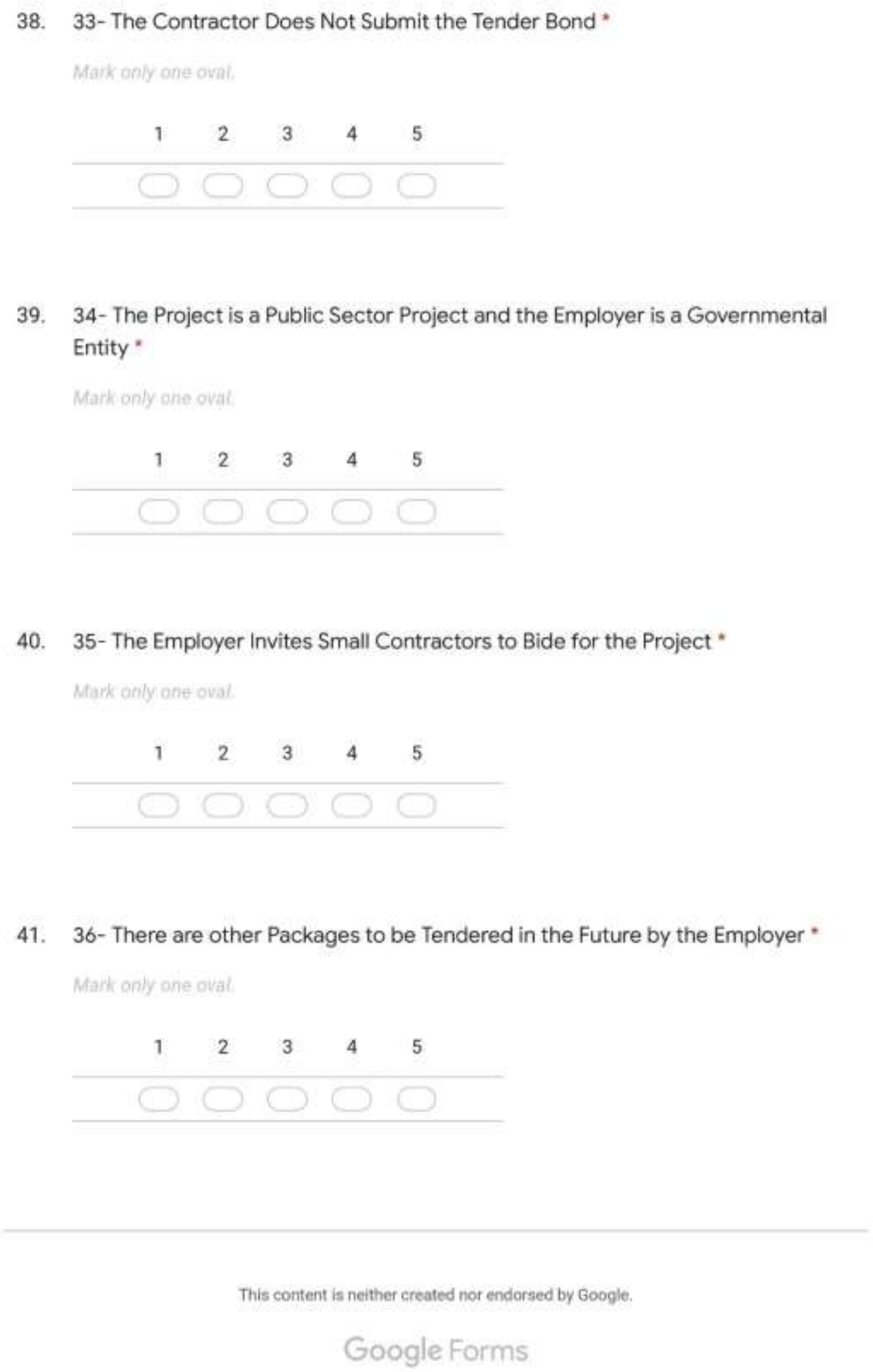
1	2	3	4	5
○	○	○	○	○

37. 32- When the Project is of a Special Type (Limited Number of Specialized Contractors to Choose From) *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

When Does the Employer Have the Upper Hand During the Negotiations Stage?



38. 33- The Contractor Does Not Submit the Tender Bond *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

39. 34- The Project is a Public Sector Project and the Employer is a Governmental Entity *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

40. 35- The Employer Invites Small Contractors to Bide for the Project *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

41. 36- There are other Packages to be Tendered in the Future by the Employer *

Mark only one oval.

1	2	3	4	5
○	○	○	○	○

This content is neither created nor endorsed by Google.

Google Forms

Appendix D: VS Code:

1- Actual Code:

```
#!/usr/bin/python
import Tkinter
import gambit
import numpy as np
import decimal
import json
from Tkinter import Tk, Label


def formatDict(actionDict):
    items = actionDict.items()
    acc = ""
    for item in items:
        acc = acc+"\n    "+item[0].label.split("/")[0]+":
"+str(item[1])
    return acc


def solveGambit(maxActions):
    solvedProfile = solver.solve(g)[0]

    for terminalObject in terminalObjects:
        terminalObject["prob"] = round(solvedProfile.__getitem__(
            terminalObject["action"]) *
solvedProfile.realiz_prob(terminalObject["action"].infoset), 4)
        terminalObject["sumPayoffs"] =
float(terminalObject["node"].outcome.__getitem__(
            0))+float(terminalObject["node"].outcome.__getitem__(1))

    maxObject = max(terminalObjects, key=lambda terminalObject: (
        terminalObject['prob'], terminalObject['sumPayoffs']))
    maxActions.append(maxObject["action"])
    return maxObject


def setNewNumbersGameOne(contractorBid, competitorBid):
    terminalObjects[0]["node"].outcome.__setitem__(
        0, int(contractorBid-projectCost))
    terminalObjects[0]["node"].outcome.__setitem__(
        1, int(ownerValue-contractorBid))
    terminalObjects[1]["node"].outcome.__setitem__(0, 0)
    terminalObjects[1]["node"].outcome.__setitem__(
        1, int(min(ownerValue, contractorBid)-competitorBid))
    terminalObjects[2]["node"].outcome.__setitem__(
        0, int(contractorBid*0.99-projectCost))
    terminalObjects[2]["node"].outcome.__setitem__(
```

```
        1, int(ownerValue-contractorBid*0.99))
    terminalObjects[3]["node"].outcome.__setitem__(0, 0)
    terminalObjects[3]["node"].outcome.__setitem__(
        1, int(min(ownerValue, contractorBid)-competitorBid))
    terminalObjects[4]["node"].outcome.__setitem__(
        0, int((contractorBid*0.98)-projectCost))
    terminalObjects[4]["node"].outcome.__setitem__(
        1, int(ownerValue-contractorBid*0.98))
    terminalObjects[5]["node"].outcome.__setitem__(0, 0)
    terminalObjects[5]["node"].outcome.__setitem__(
        1, int(min(ownerValue, (contractorBid*0.98))-competitorBid))
    terminalObjects[6]["node"].outcome.__setitem__(
        0, int((contractorBid*0.97)-projectCost))
    terminalObjects[6]["node"].outcome.__setitem__(
        1, int(ownerValue-(contractorBid)*0.97))
    terminalObjects[7]["node"].outcome.__setitem__(0, 0)
    terminalObjects[7]["node"].outcome.__setitem__(
        1, int(min(ownerValue, contractorBid*0.98)-competitorBid))
    terminalObjects[8]["node"].outcome.__setitem__(
        0, int((contractorBid*0.96)-projectCost))
    terminalObjects[8]["node"].outcome.__setitem__(
        1, int(ownerValue-(contractorBid*0.96)))
    terminalObjects[9]["node"].outcome.__setitem__(0, 0)
    terminalObjects[9]["node"].outcome.__setitem__(
        1, int(min(ownerValue, (contractorBid*0.96))-competitorBid))
    terminalObjects[10]["node"].outcome.__setitem__(
        0, int(contractorBid*0.95-projectCost))
    terminalObjects[10]["node"].outcome.__setitem__(
        1, int(ownerValue-(contractorBid)*0.95))
    terminalObjects[11]["node"].outcome.__setitem__(0, 0)
    terminalObjects[11]["node"].outcome.__setitem__(
        1, int(min(ownerValue, contractorBid*0.96)-competitorBid))


def setNewNumbersGameTwo(contractorBid, ownerValueGameTwo):
    terminalObjects[0]["node"].outcome.__setitem__(
        0, int(contractorBid-projectCost))
    terminalObjects[0]["node"].outcome.__setitem__(
        1, int(ownerValueGameTwo-contractorBid))
    terminalObjects[1]["node"].outcome.__setitem__(
        0, 0)
    terminalObjects[1]["node"].outcome.__setitem__(
        1, 0)
    terminalObjects[2]["node"].outcome.__setitem__(
        0, int((contractorBid*0.99)-projectCost))
    terminalObjects[2]["node"].outcome.__setitem__(
        1, int(ownerValueGameTwo-(contractorBid)*0.99))
    terminalObjects[3]["node"].outcome.__setitem__(
        0, 0)
    terminalObjects[3]["node"].outcome.__setitem__(
        1, 0)
    terminalObjects[4]["node"].outcome.__setitem__(
```

```
        0, int((contractorBid*0.98)-projectCost))
    terminalObjects[4]["node"].outcome.__setitem__(
        1, int(ownerValueGameTwo-(contractorBid*0.98)))
    terminalObjects[5]["node"].outcome.__setitem__(
        0, 0)
    terminalObjects[5]["node"].outcome.__setitem__(
        1, 0)
    terminalObjects[6]["node"].outcome.__setitem__(
        0, int((contractorBid*0.97)-projectCost))
    terminalObjects[6]["node"].outcome.__setitem__(
        1, int(ownerValueGameTwo-(contractorBid*0.97)))
    terminalObjects[7]["node"].outcome.__setitem__(
        0, 0)
    terminalObjects[7]["node"].outcome.__setitem__(
        1, 0)
    terminalObjects[8]["node"].outcome.__setitem__(
        0, int((contractorBid*0.96)-projectCost))
    terminalObjects[8]["node"].outcome.__setitem__(
        1, int(ownerValueGameTwo-(contractorBid*0.96)))
    terminalObjects[9]["node"].outcome.__setitem__(
        0, 0)
    terminalObjects[9]["node"].outcome.__setitem__(
        1, 0)
    terminalObjects[10]["node"].outcome.__setitem__(
        0, int((contractorBid*0.95)-projectCost))
    terminalObjects[10]["node"].outcome.__setitem__(
        1, int(ownerValueGameTwo-(contractorBid*0.95)))
    terminalObjects[11]["node"].outcome.__setitem__(
        0, 0)
    terminalObjects[11]["node"].outcome.__setitem__(
        1, 0)


inputNumbers = {}
with open('inputNumbers.json') as json_file:
    inputNumbers = json.load(json_file)
ownerValue = inputNumbers['ownerValue']
projectCost = inputNumbers['projectCost']
averages = inputNumbers['averages']
stds = inputNumbers['stds']
inputNumbers = None

g = gambit.Game.read_game("Trialtifa .gbt")

def generateNodesArray(subroot, acc):
    children = subroot.children
    acc.append(subroot)
    for childNode in children:
        generateNodesArray(childNode, acc)

players = g.players
actions = g.actions
```

```
infosets = g.infosets
rootNode = g.root
nodes = []
solver = gambit.nash.ExternalEnumPureSolver()
generateNodesArray(rootNode, nodes)

terminalNodes = filter(lambda node: node.is_terminal, nodes)
nonTerminalNodes = filter(lambda node: not node.is_terminal, nodes)

terminalActions = filter(lambda action: not any(
    action.precedes(nonTerminalNode) for nonTerminalNode in
nonTerminalNodes), actions)
terminalActions = sorted(terminalActions, key=lambda terminalAction:
int(
    terminalAction.label.split('/')[1]))
terminalObjects = []
counterContractorOne = 0
counterContractorTwo = 0
counterGameOneWon = 0
for index in range(len(terminalActions)):
    terminalObject = {}
    terminalObject["action"] = terminalActions[index]
    terminalObject["node"] = next(
        terminalNode for terminalNode in terminalNodes if
terminalActions[index].precedes(terminalNode))
    terminalObjects.append(terminalObject)

terminalActions = None
terminalNodes = None
maxActionsGameOne = []
maxActionsGameTwo = []
maxActionsContractorOne = {"won": [], "lost": []}
maxActionsContractorTwo = {"won": [], "lost": []}

numberIterations = 1000
for index in range(numberIterations):
    playerBids = []
    for indexAverages in range(len(averages)):
        playerBids.append(
            projectCost*(np.random.normal(averages[indexAverages],
stds[indexAverages])))
    contractor1Bid = playerBids[0]
    contractor2Bid = playerBids[1]
    playerBids.sort(reverse=True)

    setNewNumbersGameOne(playerBids[0], playerBids[1])
    maxObjectGameOne = solveGambit(maxActionsGameOne)

    setNewNumbersGameTwo(
        playerBids[1], ownerValue-
maxObjectGameOne["node"].outcome.__getitem__(1))
    maxObjectGameTwo = solveGambit(maxActionsGameTwo)
```

```
    payoffOwnerGameOne =
maxObjectGameOne["node"].outcome.__getitem__(1)
    payoffOwnerGameTwo =
maxObjectGameTwo["node"].outcome.__getitem__(1)

    if contractor1Bid < contractor2Bid:
        if payoffOwnerGameTwo >= 0:
            counterContractorOne += 1

maxActionsContractorTwo["lost"].append(maxActionsGameOne[index])

maxActionsContractorOne["won"].append(maxActionsGameTwo[index])
        else:
            counterContractorTwo += 1
            counterGameOneWon += 1

maxActionsContractorTwo["won"].append(maxActionsGameOne[index])

maxActionsContractorOne["lost"].append(maxActionsGameTwo[index])
    else:
        if payoffOwnerGameTwo >= 0:
            counterContractorTwo += 1

maxActionsContractorTwo["won"].append(maxActionsGameOne[index])

maxActionsContractorOne["lost"].append(maxActionsGameTwo[index])
        else:
            counterContractorOne += 1
            counterGameOneWon += 1

maxActionsContractorTwo["lost"].append(maxActionsGameOne[index])

maxActionsContractorOne["won"].append(maxActionsGameTwo[index])

takenActionsContractorOne = maxActionsContractorOne["lost"] +\
    maxActionsContractorOne["won"]
takenActionsContractorTwo = maxActionsContractorTwo["lost"] +\
    maxActionsContractorTwo["won"]
occurrencesContractorOne = dict((maxAction,
takenActionsContractorOne.count(maxAction))
                                for maxAction in
set(takenActionsContractorOne))
occurrencesContractorTwo = dict((maxAction,
takenActionsContractorTwo.count(maxAction))
                                for maxAction in
set(takenActionsContractorTwo))


occurrencesContractorOneWon = dict((maxAction,
maxActionsContractorOne["won"].count(maxAction))
                                   for maxAction in
set(maxActionsContractorOne["won"]))
```

```
occurrencesContractorTwoWon = dict((maxAction,
maxActionsContractorTwo["won"].count(maxAction))
                                   for maxAction in
set(maxActionsContractorTwo["won"]))

occurrencesGameOne = dict((maxAction,
maxActionsGameOne.count(maxAction))
                          for maxAction in set(maxActionsGameOne))
occurrencesGameTwo = dict((maxAction,
maxActionsGameTwo.count(maxAction))
                          for maxAction in set(maxActionsGameTwo))

print ("Printing Occurences of Game:")
print (occurrencesGameOne)
print ("#######################")
print (occurrencesGameTwo)
print ("\n\n\n##################")
print ("Printing Occurences of Contractor:")
print(occurrencesContractorOne)
print ("###############")
print(occurrencesContractorTwo)
print ("\n\n\n##################")
print (" Number of Wins Contractor One: " + str(counterContractorOne))
print (" Number of Wins Contractor Two: " + str(counterContractorTwo))
print(" Number of Wins Game One: " + str(counterGameOneWon))
print ("Printing Winning Occurences:")
print(occurrencesContractorOneWon)
print ("###############")
print(occurrencesContractorTwoWon)
print ("\n\n\n##################")

root = Tk()
root.title("Predicted Outcomes")

labelNumberIterations = Label(
    root, text="Number of Iterations: "+str(numberIterations))
labelNumberIterations.grid(row=0, sticky="w")


labelContractorOneWins = Label(
    root, text="Number of Wins of Contractor 1:
"+str(counterContractorOne))
labelContractorOneWins.grid(row=1, sticky="w")

labelContractorTwoWins = Label(
    root, text="Number of Wins of Contractor 2:
"+str(counterContractorTwo))
labelContractorTwoWins.grid(row=2, sticky="w")

labelNumberofWinsGameOne = Label(
    root, text="Number of Wins from 1st Game:
"+str(counterGameOneWon)+"\n\n\n\n")
```

```
labelNumberofWinsGameOne.grid(row=3, sticky="w")

labelWinningStrategiesContractorOne = Label(
    root, justify="left", text="Distribution of Winning Strategies of 
Contractor 1:"+formatDict(occurrencesContractorOneWon)+"\n\n\n\n")
labelWinningStrategiesContractorOne.grid(row=4, sticky="w")

labelWinningStrategiesContractorTwo = Label(
    root, justify="left", text="Distribution of Winning Strategies of 
Contractor 2:"+formatDict(occurrencesContractorTwoWon))
labelWinningStrategiesContractorTwo.grid(row=5, sticky="w")

root.mainloop()
```

2- Input Graphical User Interface Code:

```
#!/usr/bin/python
import json
from Tkinter import Tk, Label, Entry, Button
root = Tk()
root.title("Input Entry Data to the Model")


def fetchInputs():
    inputData = {
        "projectCost": float(entryCost.get()),
        "ownerValue": float(entryOwnerValue.get()),
        "averages": [
            float(entryContractorOneAverage.get()),
            float(entryContractorTwoAverage.get())
        ],
        "stds": [
            float(entryStdContractorOne.get()),
            float(entryStdContractorTwo.get())
        ]
    }

    with open('inputNumbers.json', 'w') as fp:
        json.dump(inputData, fp,  indent=4)

    root.destroy()
    import trial


labelCost = Label(root, text="Project Direct Cost:")
entryCost = Entry(root)
labelCost.grid(row=0, sticky="w")
entryCost.grid(row=0, column=1)
```

```
labelOwnerValue = Label(root, text="Owner Value:")
entryOwnerValue = Entry(root)
labelOwnerValue.grid(row=1, sticky="w")
entryOwnerValue.grid(row=1, column=1)

labelContractorOneAverage = Label(
    root, text="Contractor One Average Pricing Factor:")
entryContractorOneAverage = Entry(root)
labelContractorOneAverage.grid(row=2, sticky="w")
entryContractorOneAverage.grid(row=2, column=1)

labelContractorTwoAverage = Label(
    root, text="Contractor Two Average Pricing Factor:")
entryContractorTwoAverage = Entry(root)
labelContractorTwoAverage.grid(row=3, sticky="w")
entryContractorTwoAverage.grid(row=3, column=1)

labelStdContractorOne = Label(
    root, text="Contractor One Std of Pricing Factor:")
entryStdContractorOne = Entry(root)
labelStdContractorOne.grid(row=4, sticky="w")
entryStdContractorOne.grid(row=4, column=1)

labelStdContractorTwo = Label(
    root, text="Contractor Two Std of Pricing Factor:")
entryStdContractorTwo = Entry(root)
labelStdContractorTwo.grid(row=5, sticky="w")
entryStdContractorTwo.grid(row=5, column=1)

fetchButton = Button(root, text="Enter", command=fetchInputs)
fetchButton.grid(row=6)

root.mainloop()
```

Appendix E: Projects Data Provided by Contractors:

1- Contractor 1 Data:

#	Project Type	Direct Cost (EGP)	Bid Price (EGP)	Project Awarded (Y/N)	Pricing Factor
1	Sports Club	15,942,029	22,000,000	Y	1.38
2	Residential Buildings	1,566,666,667	2,115,000,000	Y	1.35
3	Commercial Mall	384,615,385	550,000,000	N	1.43
4	Residential Villas	1,267,223,379	1,698,079,328	Y	1.34
5	Residential Villas	243,661,972	346,000,000	Y	1.42
6	Residential Villas	512,765,957	723,000,000	Y	1.41
7	Educational Buildings	370,503,597	515,000,000	Y	1.39
8	Commercial Mall	49,189,166	64,929,699	Y	1.32
9	Residential Buildings	816,666,667	1,176,000,000	Y	1.44
10	Lake and Recreation Area	254,311,462	348,406,703	Y	1.37
11	Residential Buildings	558,571,429	782,000,000	N	1.4
12	Infrastructure and Fences	271,563,706	353,032,817	Y	1.3
13	Cultural Centre	47,757,022	61,606,559	Y	1.29
14	Residential Villas	416,739,810	558,431,346	Y	1.34
15	Pharmaceutical Plant	69,863,014	102,000,000	Y	1.46
16	Multipurpose Tower	1,677,340,356	2,465,690,324	Y	1.47
17	Hospital	8,503,496,503	12,330,069,930	N	1.45
18	Residential Villas	300,990,281	412,356,685	N	1.37
19	Entertainment Area	486,716,312	686,270,000	Y	1.41
20	Residential Buildings	518,461,538	741,400,000	Y	1.43

2- Contractor 2 Data:

#	Project Type	Direct Cost (EGP)	Bid Price (EGP)	Project Awarded (Y/N)	Pricing Factor
1	Residential Buildings	897,000,000	1,246,830,000	Y	1.39
2	Residential Buildings	423,000,000	626,040,000	Y	1.48
3	Industrial Building	733,000,000	989,550,000	Y	1.35
4	Passenger Bridges	95,000,000	133,000,000	Y	1.4
5	Bridges	465,000,000	660,300,000	Y	1.42
6	Infrastucture	388,000,000	574,240,000	Y	1.48
7	Infrastructure	388,000,000	574,240,000	N	1.48
8	Infrastructure	268,000,000	377,880,000	Y	1.41
9	Residential Buildings	659,000,000	955,550,000	N	1.45
10	Residential Buildings	1,169,500,000	1,777,640,000	N	1.52
11	Power Plant	522,000,000	809,100,000	Y	1.55
12	Power Plant	417,000,000	663,030,000	N	1.59
13	Educational Buildings	606,000,000	836,280,000	Y	1.38
14	Educational Buildings	407,000,000	577,940,000	Y	1.42
15	Hotel	319,000,000	472,120,000	Y	1.48
16	Residential Buildings	595,000,000	827,050,000	N	1.39
17	Office Building	614,000,000	835,040,000	Y	1.36
18	Infrastructure	809,000,000	1,084,060,000	Y	1.34
19	Power Plant	350,699,554	561,119,286	N	1.6
20	Power Plant	416,000,000	615,680,000	Y	1.48
21	Industrial Building	288,000,000	397,440,000	N	1.38