

Programming Pedagogy in the Age of Accessible Artificial Intelligence

by

Lakshmi Preethi Kamak

A thesis submitted in partial fulfillment of the

requirements for the degree of

Master of Science

in

Computer Science

in the

Faculty of Science and Environmental Studies

of

Lakehead University, Thunder Bay

Committee in charge:

Dr. Vijay Mago (Principal Supervisor)

Dr. Piper Jackson (External Examiner)

Dr. Jinan Fiadhi (Internal Examiner)

Winter 2023

The thesis of Lakshmi Preethi Kamak, titled Programming Pedagogy in the Age of Accessible Artificial Intelligence, is approved:

Chair _____	Date _____
_____	Date _____
_____	Date _____

Lakehead University, Thunder Bay

Declaration

I certify that,

- The work contained in this thesis is original and has been done by myself and under the general supervision of my supervisor(s).
- The work reported herein has not been submitted to any other Institute for any degree or diploma.
- Whenever I have used materials (concepts, ideas, text, expressions, data, graphs, diagrams, theoretical analysis, results, etc.) from other sources, I have given due credit by citing them in the text of the thesis and giving their details in the references. Elaborate sentences used verbatim from published work have been clearly identified and quoted.
- I also affirm that no part of this thesis can be considered plagiarism to the best of my knowledge and understanding and take complete responsibility if any complaint arises.
- I am fully aware that my thesis supervisor(s) are not in a position to check for any possible instance of plagiarism within this submitted work.

Programming Pedagogy in the Age of Accessible Artificial Intelligence

Copyright 2023

by

Lakshmi Preethi Kamak

Abstract

PROGRAMMING PEDAGOGY IN THE AGE OF ACCESSIBLE ARTIFICIAL INTELLIGENCE

In recent years, new teaching opportunities have emerged as artificial intelligence has gained increasing attention in computational thinking education. However, to design effective pedagogy based on the present research landscape, the technology solution must be tailored to a learning environment through a collaboration between human-computer interaction and human-artificial intelligence interaction research. The thesis aims to enhance programming experiences and increase accessibility to programming resources for students in remote schools and post-secondary graduate settings using human-computer interaction and human-artificial intelligence interaction techniques. It addresses the limited computational thinking education resources and the potential of artificial intelligence-assisted coding in a self-learning method suitable for remote Northwestern First Nation communities in Canada. This thesis proposes methods to cater to students' learning styles in two different learning environments using human-computer interaction for kindergarten to grade 12 students and human-artificial intelligence interaction for university students. Incorporating these research principles can help novice programmers overcome cognitive overload and poor user experience and achieve an optimal user experience. The thesis begins with bibliometric analysis and provides a holistic perspective of computational thinking and artificial intelligence trending strategies. It then presents an empirical study on human-computer interaction, investigating computational thinking in remote kindergarten to grade 12 schools with blended learning environments. It also presents another empirical study on human-artificial intelligence interaction to experiment with a self-learning style for artificial intelligence coding assistants for university students using massive open online courses. The thesis findings inform the development of pedagogy, research areas and technology advancements for developing computational thinking skills through applicable artificial intelligence in a multidisciplinary research aspect that has grown exponentially since 2017. Our empirical studies demonstrate that the scaffolding intervention technique for Python integrated with block-based visual aids combines

block and text-based programming effectively and is suitable for middle school education. Likewise, we demonstrate ChatGPT's technological acceptance as a coding support tool for university education while undertaking complex subjects such as reinforcement learning. The human-computer interaction and human-artificial intelligence interaction metrics results are vital in promoting inclusive and accessible education. Moreover, this research contributes significantly to developing future coding assistants and enhancing students' programming experiences. As a result of this work, we contribute to removing geographical barriers to programming resources in Canadian Northwestern First Nations communities by developing a sustainable learning culture that adapts to fast-evolving technology.

Dedication

This is dedicated to all my family and friends who have been supportive and motivating throughout my research journey, and to my grandfather, Late Mr Raja Sekar and my grandmother, Late Mrs Rohini Devi, who have always been my biggest and proud advocates and the educators who believed in me and gave me this research platform.

Contents

Contents	ii
List of Figures	iv
List of Tables	vi
List of Abbreviations	vii
1 Introduction	1
2 The Convergence of Computational Thinking and Artificial Intelligence in Programming Pedagogy: A Bibliometric Analysis	3
2.1 Introduction	4
2.2 Related Works	5
2.3 Methodology	7
2.4 Results and Discussion	10
2.5 Conclusion	24
3 Evaluating the efficacy of Python Scaffolding to teach Computational Thinking in Remote Schools	26
3.1 Introduction	27
3.2 Background	29
3.3 Methodology	31
3.4 Results and Discussion	38
3.5 Conclusion	46
4 Assessing the Impact of AI-Assisted Coding in Self-Directed Reinforcement Learning MOOC	48
4.1 Introduction	49
4.2 Related Works	50
4.3 Methodology	52
4.4 Results and Discussion	57

4.5 Conclusion	66
5 Conclusion	68
Bibliography	70
6 Appendix	81
6.1 Research Ethics Approval	81
6.2 Student Questionnaire: HCI Study	105
6.3 Student Questionnaire: AI Context	115
6.4 Student Questionnaire: non-AI Context	121
6.5 Transcript: Participant and ChatGPT	128

List of Figures

2.1	Architecture diagram of dataset 1 ($\mathbb{D}1$) and dataset 2 ($\mathbb{D}2$) representing extracts from the literature highlighted with cross hatching	8
2.2	Top research areas from CT-AI literature	11
2.3	Annual distribution of CT-AI research publications	12
2.4	Countries contributing to CT-AI research	13
2.5	Journal and conference distribution of CT-AI research	14
2.6	Co-word network map of the keywords in abstract and title from programming pedagogy literature	16
2.7	Co-word network map of the keywords in abstract and title from CT-AI literature	17
2.8	Keywords distribution in the co-word network map from $\mathbb{D}2$	18
2.9	Author citation map in the field of programming pedagogy	22
3.1	Curriculum sample - Chapter 7: Computational Thinking with Cultural Context: Python-based block code with a visual aid to highlight the flow of execution . .	32
3.2	Comparison of the original Callysto CTt survey item sample and the translated CT survey item with visual blocks [50]	35
3.3	Graph depicting the student performance metrics in CT and programming skills	40
3.4	Student's perception of their comfort level with technology	41
3.5	Student's self evaluation of their computational thinking skills	42
3.6	Student attitudes towards computationally thinking learning interests	42
3.7	Student attitudes towards programming experience	43
3.8	Student's subjective advantages in blended learning	44
3.9	Student feedback on their blended learning course experience	44
3.10	Comparison of student's experience in blended learning and in-person classes . .	45
3.11	Spearman's correlation analysis of computational thinking related elements . . .	46
4.1	Human-Artificial Intelligence Interaction Evaluation Framework	54
4.2	General preferences of graduate students based on their coding experience . . .	58
4.3	General attitude towards potential bias in coding assistance tools	58
4.4	Reinforcement learning coding test performance	61
4.5	Explicit numerical information on the difficulty index and tasks supporting Figure 4.4	61

4.6	Distribution of the test performance, perceived difficulty and confidence of AI and non-AI context groups	63
4.7	Perceived impact on critical thinking on using ChatGPT pre-test	64
4.8	Perceived impact on their self-direction with the respective coding assistance tool post-test	65
4.9	Perceived usefulness of the coding assistance tools	65
6.1	Computational thinking Algorithm introduction [50]	105
6.2	Computational Thinking Question 1 [50]	105
6.3	Computational Thinking Question 1 options	106
6.4	Computational Thinking Question 2 [50]	106
6.5	Computational Thinking Question 3 [50]	107
6.6	Computational Thinking Question 3 options	108
6.7	Computational Thinking Question 4 [50]	109
6.8	Computational Thinking Question 5 [50]	110
6.9	Computational Thinking Answer 5	110
6.10	Computational Thinking Question 6 [50]	111
6.11	Computational Thinking Question 7 [50]	112
6.12	Computational Thinking Question 7 options	112
6.13	Computational Thinking Question 8	113
6.14	Computational Thinking Question 8 options	114
6.15	Game tree for the Minimax algorithm test [104]	116
6.16	Game tree for the Alpha-Beta pruning algorithm test [106]	119
6.17	Game tree for the Minimax algorithm test [104]	123
6.18	Game tree for the Alpha-Beta pruning algorithm test [106]	125

List of Tables

2.1	Countries and their CT-AI research publications	13
2.2	Cluster information from Co-word knowledge map of the keyword terms from $\mathbb{D}2$	19
2.3	Top research works leading CT-AI research with citations as on 21st February, 2023	21
2.4	Top influential programming pedagogy authors with most citations	23
2.5	Top CT-AI research works by influential programming pedagogy authors	24
3.1	Coding expectations in the Ontario mathematics curriculum for grades 1 to 8 [38]	27
3.2	Descriptive text statistics of the original BL survey and translated BL survey	38
3.3	Readability test scores of the original BL survey and translated BL survey	39
4.1	GPT coding exam scores for real-time exam simulations	52
4.2	Summary of Participant Demographics	57
4.3	Reading ease of the ChatGPT prompts on RL explanation	59
4.4	The variety of Reinforcement Learning approaches implemented by participants for Question 1	60

List of Abbreviations

Abbreviation	Description
AI	Artificial Intelligence
AIoT	Artificial Intelligence of Things
BL	Blended Learning
BLQ	Blended Learning Questionnaire
CT	Computational Thinking
CTt	Computational Thinking Test
CT-AI	Convergence of Computational Thinking and Artificial intelligence
CTQ	Computational Thinking Questionnaire
D2L	Desire 2 Learn
FNMI	First Nation, Metis and/or Inuit
HCI	Human computer interaction
HAI	Human Artificial Intelligence interaction
IDT	Innovation Diffusion Theory
K12	Kindergarten through 12th grade
ML	Machine learning
MOOCs	Massive Open Online Courses
ME-CoT	Metacognitive Empowerment by Computational Thinking
NW	North Western
RL	Reinforcement Learning
STEM	Science, Technology, Engineering and Math
TAM	Technology Acceptance Model
TCPS	Tri-Council Policy Statement
TLS	Total Link Strength
UTAUT	Unified Theory of Acceptance and Use of Technology
WoS	Web of Science

Acknowledgments

I sincerely thank my supervisor, Dr Vijay Mago, for the research opportunities and platform he provided, that nurtured my talent and enabled me to grow. I am immensely thankful for the NSERC PromoScience grant and resources provided by the DaTALab and Nijii Indigenous Mentorship, which have been a significant source of support throughout my degree. Lisa Harris was an invaluable mentor to me in outreach, helping me travel to remote communities and teaching coding. I am very grateful for her always thoughtful approach. I also thank Bleau Moores, his support was instrumental in securing the approval of the Research Ethics Board. Furthermore, I thank Dr Akriti Jindal for her constant guidance in Chapter 2 of my research. My colleagues in the DaTALab, Lakehead University have also been an essential part of my research journey, and I extend my gratitude to them. Denise Baxter's support and assistance have been instrumental in organizing and coordinating student coding workshops. I appreciate the constant support and encouragement my parents, brother, and sister have provided me throughout this journey.

Chapter 1

Introduction

Throughout this degree program, I have written several articles culminating in this thesis. Computational thinking (CT) and artificial intelligence (AI) have gained traction as topics of interest in programming pedagogy research in recent years. However, choosing an appropriate programming curriculum and introducing suitable technology solutions for students depends on various factors. To establish a rewarding learning experience for the students, we can apply human-computer interaction (HCI) and human-artificial intelligence interaction (HAI) design principles in coding education while measuring their efficacy through respective metrics.

This thesis addresses different aspects of programming pedagogy in the age of fast technology and accessible AI. It aims to design effective programming pedagogy using HCI and HAI techniques based on the current research landscape. All the presented studies seek to increase accessibility to programming resources for students in remote schools and post-secondary graduate settings. It addresses the limited computational thinking education resources for students and the potential of AI-assisted learning in a self-learning method suitable for remote Northwestern First Nation communities in Canada. We hope to contribute to the growing literature on programming pedagogy as learning experiences evolve due to technology and AI. The study's findings provide effective pedagogy design interventions at various student levels to prepare students for the 21st-century workforce.

This thesis follows the following structure: Chapter 2 provides a foundation for this thesis in terms of research direction and trending techniques. It begins with a bibliometric investi-

gation which offers insights into current research trends leading to the convergence of CT and AI in the programming pedagogy academic literature. Using performance analysis and science mapping methods, this analysis identifies emerging thematic areas based on significant author, journal and geographical contributions. Chapter 3 offers an innovative approach to teaching programming to middle school students using scaffolding techniques and computational thinking concepts in a blended learning environment for remote schools. The visual scaffolding technique with Python bridges the gap between block-based and text-based programming, enhancing the coding experience for novice learners who face distinct challenges in a remote setting. More specifically, this study lays out comprehensive curriculum and assessment materials aligned with Ontario learning outcomes. It has the mission to remove geographical barriers to building sustainability in First Nation Schools in the Northwestern communities of Canada. Using an adapted survey instrument in our pilot study, we identified computationally talented students with the prospect of developing STEM interests. Chapter 4 presents another area of HCI research, HAI. HAI research principles can facilitate artificial intelligence-assisted learning to enhance programming experiences. The chapter explores the efficacy of OpenAI's ChatGPT as an artificial intelligence coding assistant in graduate students' self-directed learning experience. In the study, students solved complex reinforcement learning tasks after completing a related Massive Open Online course. We developed an original survey instrument combining HCI frameworks and HAI metrics. This work evaluates the technology acceptance and impact of ChatGPT in academia using HAI metrics like learners' confidence and trust in the AI tool.

Chapter 2

The Convergence of Computational Thinking and Artificial Intelligence in Programming Pedagogy: A Bibliometric Analysis

All of this chapter is under review at a reputable journal as follows:

- Kamak, LP., Jindal A, Mago V. (2023). The Convergence of Computational Thinking and Artificial Intelligence in Programming Pedagogy: A Bibliometric Analysis

As part of my degree programme, I researched the application of Human-Computer Interaction and Human-Artificial Intelligence Interaction design on programming pedagogy to better understand the evolution and topics of interest in this field. This chapter gives a comprehensive overview of trends in programming pedagogies identified through a bibliometric analysis to understand the current research landscape, providing an essential foundation for this thesis's development. Based on the performance analysis and science mapping methods, the results inform the development of new opportunities in teaching using computational thinking along with applicable artificial intelligence.

Keywords : Artificial intelligence, Bibliometric analysis, Clustering analysis, Computational thinking, Educational computing, Science mapping.

2.1 Introduction

Artificial intelligence (AI) in education is transforming classroom learning globally. Interestingly, researchers are using AI to teach students programming and develop Computational Thinking (CT) skills [1]. AI and CT are both popular topics in the current educational landscape. AI can support the development of CT skills such as problem-solving, logical reasoning and creativity. There are numerous definitions of CT based on extensive research. We decided to use the definition of “computational thinking” provided by the British Broadcasting Corporation: “*Computational thinking allows us to take a complex problem, understand what the problem is and develop possible solutions. We can then present these solutions in a way that a computer, a human, or both, can understand*” [2]. The following components comprise the problem-solving approach.

- Decomposition: Breaking down a complex problem into smaller and manageable parts
- Pattern recognition: Looking for similarities within problems
- Abstraction: Ignoring irrelevant detail and focusing on the relevant information
- Algorithms: Developing a step-by-step solution to the problem
- Evaluating solutions: Ensuring that the solution satisfactorily and efficiently resolves the problem [2]

With the convergence of CT and AI in kindergarten through 12th grade (K12) education, there is an emerging trend of promoting CT skills through practical AI projects [1]. Furthermore, AI also allows students to apply their CT skills in the context of developing and programming intelligent tools and systems [3]. While many educators are still working on a range of upcoming challenges in this emerging field, from teacher training to ethical implications, a holistic perspective helps researchers gain fresh insights into this emerging field.

This analysis provides insight into the intersection of CT and AI research as programming pedagogies while identifying key players and influential publications in the field. Our study explores the “programming pedagogy” trends in K12 education from 1982 to 2023.

Although systematic bibliometrics conducted in the past studies focused mainly on AI and CT concepts, they did not fully embrace joint developments in CT and AI literacy. A thorough scientometric analysis is necessary to understand how CT and AI have evolved as programming pedagogies. In this direction, we conducted an in-depth investigation of the growing trend of CT and AI, incorporating two distinct datasets and exploring various themes. We refer to the integration of CT and AI as “CT-AI” throughout the thesis for convenience and readability. We also examined the literature regarding the intersection of CT-AI to supplement and strengthen the existing research. In this context, the following research questions have been investigated in this study:

- RQ1: How has the frequency distribution of CT-AI research publications changed over time?
- RQ2: Which nations, journals and authors are emerging in CT-AI research?
- RQ3: How have research development paths and research areas shifted over time in programming pedagogy?

The structure of the chapter is as follows: Section 2.2 discusses the various bibliometric studies carried out in the past few years in CT-AI. Section 2.3 demonstrates the data source, preprocessing techniques and tools. The main results of the bibliometric analysis, including a Performance Analysis and Science Mapping of CT-AI publications, are highlighted in Section 2.4. Performance Analysis examines the CT-AI research areas in the aspect of the disciplines, annual publications, geographical contribution, and emerging research sources. The Science Mapping section covers three types of analysis: co-word analysis, publication citation analysis, and author citation analysis. Section 2.5 concludes this study with the key findings for all the research questions.

2.2 Related Works

With the popularity of AI in education, many review studies aim to explore cutting-edge research into AI trends in educational research and how it shapes current educational

practices. In the recent year, a bibliometric study was carried out to analyze 135 articles from the *International Journal of Artificial Intelligence in Education* from 2015 to 2019 [4]. Their findings indicate that the United States, the United Kingdom, Canada, and Germany are the most prolific nations in the field of AI. Moreover, their study shows a robust collaboration network between Canada and the United States. According to their research, global collaboration can contribute to research in the field of AI by incorporating a variety of viewpoints and expertise. Following this study, AI literature research for a broader range of years from 2011 to 2021 has been conducted in 2022 [5]. They use bibliometric analysis to examine the trends in AI in education research in the Scopus database over the past decade. Their findings highlight that English is the most used language, and Dimitris Kalles is a prolific author in the AI in education field with 1474 citations overall on Google Scholar. His research includes designing intelligent tutoring systems and adaptive educational technologies, as well as researching a broad range of artificial intelligence-related topics [6, 7, 8]. In addition, their funding information analysis shows that the National Science Foundation and the National Natural Science Foundation of China are the leading funding sources for AI research. Their research trends show that AI in education research primarily focuses on applications to students, engineering education, teaching methods, e-learning, education systems, and AI-infused curriculum.

On the other hand, to evaluate the evolution of CT in the education domain, a bibliometric study explores the trends and developments from the years 2008 to 2020 [9]. The study identifies three main themes in the literature: integrating CT into STEM education, discussing definitions and experimental research on evaluating CT skills. Furthermore, a recent study published in the year 2023 aims to provide a more comprehensive and up-to-date review of the ever-growing field of CT [10]. This study analyzes 249 studies on CT using the analytical tool CiteSpace, from 2012 to 2021. Moreover, their results portray that the development of CT in K-12 education is the most frequently discussed topic. In terms of learning performance in CT knowledge and practices, the most active themes in their study are scale validation, motivation, self-efficacy, curriculum, and computational modeling. The authors emphasize the close connection between CT education and STEM disciplines, as well as the potential to develop skills such as creativity, collaboration, and problem-solving through CT education. Furthermore, they recommend that CT development be undertaken

using cutting-edge AI tools, augmented reality, and virtual reality.

Even though the studies mentioned above offer insightful information with AI or CT as the primary concepts, they only partially embrace collaborative advances in CT-AI literacy. The development and evolution of CT-AI as programming pedagogies require comprehensive bibliometric analysis. A more detailed analysis is required to understand the growing trend of AI integration into CT and to complement and strengthen existing research.

2.3 Methodology

The Web of Science (WoS) is a significant source of databases on research articles, conference proceedings, and other scientific literature from 1900 to the present [11]. The WoS Core Collection database offers researchers a substantial source of reliable journals and publications and their respective citation data. The database covers over 34,000 scholarly journals in the sciences, social sciences, arts, and humanities and is expanding day by day. Exporting and importing data from WoS to the bibliometric platform enables researchers to track their work's impact and discover upcoming research in their fields. The multi-step investigation process allowed for a comprehensive examination of programming pedagogy research. The investigation entails a study of programming pedagogy literature from 1982 to 2023, followed by an exploration of CT-AI literature from 2010 to 2023. An in-depth analysis using the WoS Core Collection's publication data enabled us to gather insight into the advancements and impact of CT-AI in education.

The study uses WoS Core Collection for its data source and exports dataset 1 ($\mathbb{D}1$) for programming pedagogies and dataset 2 ($\mathbb{D}2$) for CT-AI, as shown in Figure 2.1. The main objective of developing two separate datasets is to address the research questions in an efficient way. RQ1 will be answered based on $\mathbb{D}2$ and RQ2, RQ3 based on $\mathbb{D}1$ and $\mathbb{D}2$. $\mathbb{D}1$ focuses broadly on the education discipline and pedagogy trends, while $\mathbb{D}2$ focuses more narrowly on the multi-disciplinary analysis intersection of CT and AI research. Since CT-AI is a niche area and the overall picture is only complete when the prior CT experience of the authors from $\mathbb{D}1$ is taken into consideration. For constructing $\mathbb{D}1$, we consider popular programming pedagogies like block-based, Scratch, AI, PRIMM, Block Model, Code Tracing, and Parson's Problems for the experiment [12, 13, 14, 15, 16, 17]. The export

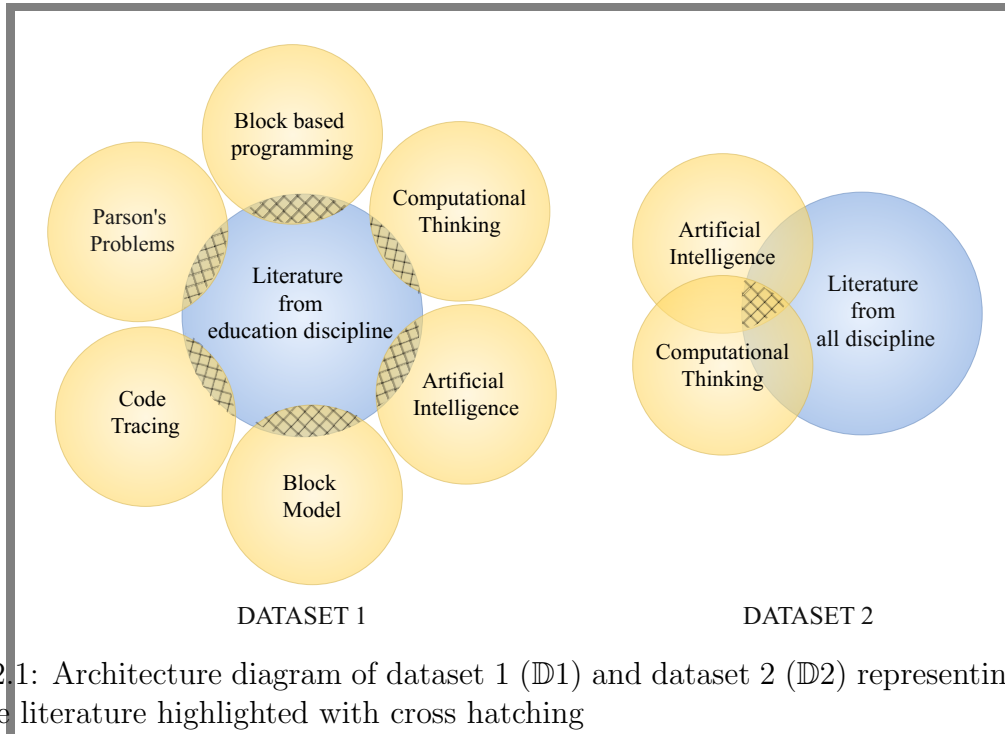


Figure 2.1: Architecture diagram of dataset 1 ($\mathbb{D}1$) and dataset 2 ($\mathbb{D}2$) representing extracts from the literature highlighted with cross hatching

process limits the WoS categories to Education, Educational Research, or Education Scientific Disciplines. Moreover, the export also filters citation topics to only Education and Educational Research. To export the data from the WoS interface we use the following query: (ALL=(“Programming”) OR ALL=(“programming”)) AND (ALL=(“Education”) OR ALL=(“education”) OR ALL=(“pedagogy”)) AND ((ALL=(“Computational thinking”) OR ALL=(“Computational Thinking”)) OR (ALL=(“Block based”) OR ALL=(“block based”) OR ALL=(“block-based”) OR ALL=(“Block-based”) OR ALL=(“Scratch”) OR ALL=(“scratch”)) OR (ALL=(“Artificial Intelligence”) OR ALL=(AI) OR ALL=(“Artificial intelligence”) OR ALL=(“artificial intelligence”)) OR (ALL=(“PRIMM”)) OR (ALL=(“Block Model”) OR ALL=(“Block model”) OR ALL=(“block model”)) OR (ALL=(“Code Tracing”) OR ALL=(“code tracing”) OR ALL=(“Code tracing”)) OR (ALL=(“Parson’s Problems”) OR ALL=(“parson’s problems”) OR ALL=(“Parson’s problems”))). The output format of the export is a series of tab-delimited files. Furthermore, we clean the data by removing rows with either missing values or outliers in the published year and WoS categories features. As a result, our dataset $\mathbb{D}1$ consists of 1062 articles. The second dataset, $\mathbb{D}2$, is extracted using the broad search term “Computational Thinking” and “Artificial

Intelligence” with no filter on categories using the search query: (ALL=(“Computational thinking”) OR ALL=(“Computational Thinking”) OR ALL=(“computational thinking”)) AND (ALL=(“Artificial Intelligence”) OR ALL=(AI) OR ALL=(“Artificial intelligence”) OR ALL=(“artificial intelligence”)). The resultant dataset contains 157 articles after cleaning missing values and outliers from the published year and excluding unrelated categories from neuroscience and fuzzy systems.

The datasets are then subjected to a performance analysis, which included metrics related to publications, interdisciplinary analysis, geo-contribution, and journal leadership in CT-AI [18]. This analysis is performed using quantitative methods to determine the CT-AI field’s growth trend and activity. We apply these methods to the annual distribution of the publications and identify the initial and rapid development phases. Based on publication metrics, in the performance analysis approach, the geographical analysis illustrates the active regions in the field, and the source analysis identifies emerging sources of publications. Likewise, in the science mapping approaches in our analysis using VOSViewer, the co-word analysis identifies clusters, and author citation analysis identifies the most prolific authors [19]. VOSViewer is a bibliometric network visualization software tool for mapping and exploring the relationships between scientific publications, authors, and journals in a particular field. Through VOSViewer, researchers can analyze citation networks, which provide insight into the structure of scientific communities and the relationships between different research areas. VOSViewer can import and analyze large and complex datasets from the WoS database. This software helps to identify patterns and trends by mapping patterns, trends, and key players in the field. Some of the VOSViewer terminologies frequently used in the following thesis are: **Nodes** represent publications, authors, or journals in the network, while **links** represent the connections between the nodes, indicating citation or authorship relationships. The **total link strength** (TLS) refers to the sum of all the links between different nodes in a network graph. It measures the relationship between nodes, with higher link strength indicating a stronger relationship. A **cluster** is a collection of tightly connected nodes within a network. **Mapping** is the process of visualizing the nodes and edges of a network in 2D or 3D space.

2.4 Results and Discussion

This section elaborates how the research in CT-AI evolved during the last fourteen years, based on quantitative performance analysis and science mapping techniques. For a comprehensive study to analyze the state of research on AI integration to CT, we use various bibliometric techniques such as knowledge network visualization to analyze keyword co-occurrence, author citation and performance analysis for journal contribution, and geographical contribution in the CT-AI literacy field [10]. To the best of our knowledge, this is the first thorough bibliometric analysis of CT-AI literacy. Future research may benefit significantly from this study's crucial support and direction.

Performance analysis

Interdisciplinary analysis of CT-AI publications

The interdisciplinary analysis examines the growth of research in various fields using WoS category data to assess CT-AI activity [20]. The analysis is interdisciplinary, as it covers multiple disciplines, such as computer science, education, robotics, and other relevant fields, that contribute to the growth and development of CT-AI. We can observe the distribution of CT-AI research across different disciplines and identify emerging areas of interest. It can be observed in Figure 2.2 that the main focus of the research is on the field of Education, as it accounts for almost half of all research publications in this field. Furthermore, Computer Science Artificial Intelligence, Computer Science Interdisciplinary Applications and Computer Science Theory Methods are significant research areas, with 33.75%, 26.26%, and 22.5% of the contributions, respectively. Other related fields, such as Education Scientific Disciplines, Computer Science Information Systems, Computer Science Software Engineering, Electrical Electronic Engineering, Robotics and Computer Science Cybernetics, are present at 10.63%, 9.38%, 7.5%, 7.5%, 5.63%, 4.38% respectively.

2.4. RESULTS AND DISCUSSION

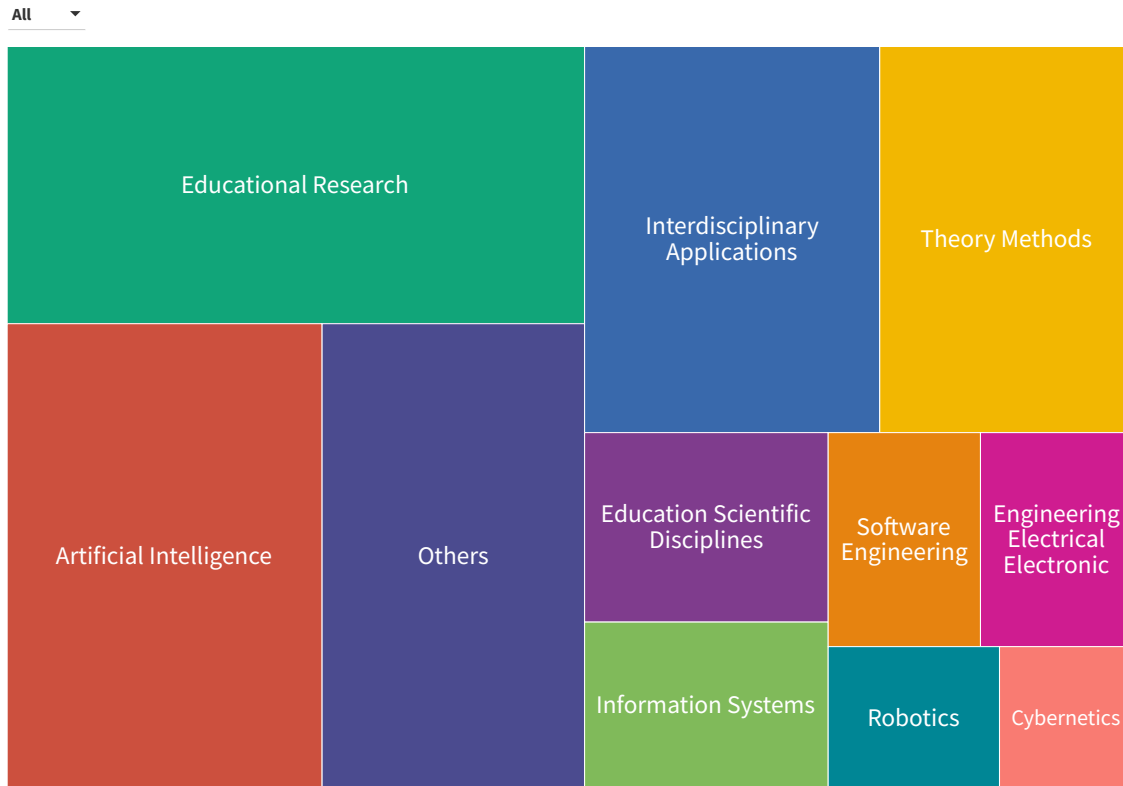


Figure 2.2: Top research areas from CT-AI literature

Annual distribution of CT-AI publications

The yearly distribution of CT-AI publications enables a comprehensive understanding of the growth of the field. Figure 2.3 illustrates the distribution of publications from 2009 to 2023 and the gradual increase in CT-AI publications. It can be observed that there are two main phases in the development of CT research: the initial exploration phase (2009–2016) and the rapid development phase (2017–2023). In the early stages of exploration, there are only 16 publications, with an average of 2 to 3 publications per year. In contrast, the volume of CT-AI correlated research surged exponentially in the rapid development stage, with 136 articles representing an average of 19 articles per year, accounting for 89.47 % of the complete data set. In total, 152 articles have been published on CT-AI, which have received 785 citations.

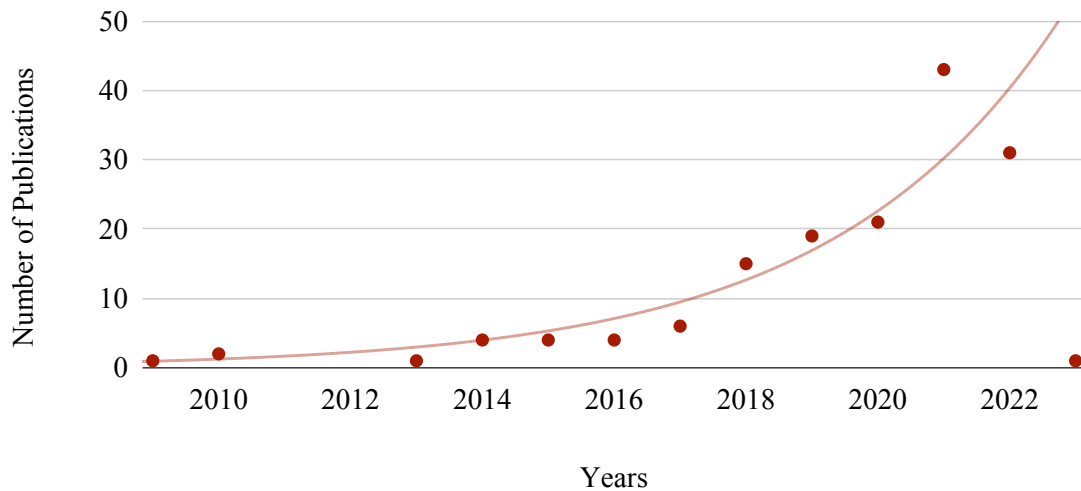


Figure 2.3: Annual distribution of CT-AI research publications

Geographic Analysis of CT-AI publications

The geographic analysis studies the distribution patterns of the production and dissemination of scholarly literature based on the geographical locations of the published resources [18]. The findings of the geographic analysis provides insights into the distribution of research resources, the development of research capacities, and the opportunities for international collaboration. Figure 2.4 along with Table 2.1 illustrates the nations and the number of contributions they have made to CT-AI. The United States of America, with 57 publications, has the most contributions, followed by Germany, Switzerland, Taiwan, and the United Kingdom, with 30, 14, 13, and 10 articles, respectively. Most of the countries listed are in Europe and North America, with a few in Asia. This suggests that CT-AI research is primarily conducted in developed countries with substantial technological infrastructure and funding for research. Emerging contributors picking up the trend in these research fields include France, Hong Kong, Canada, Ukraine, South Africa, Romania, Lithuania, and Croatia. These results support policymakers, funding agencies, and research institutions in developing strategies to support scientific research.

Table 2.1: Countries and their CT-AI research publications

Country	Publications	Country	Publications
United States	57	Turkey	2
Germany	30	Japan	2
Switzerland	14	Hong Kong	2
Taiwan	13	France	2

BIBLIO - Google Sheets



BIBLIO



File Edit View Insert Format Data Tools Extensions Help

100% | \$ % .0 .00 123 | Defaul... | - 10 + | B I A

J10

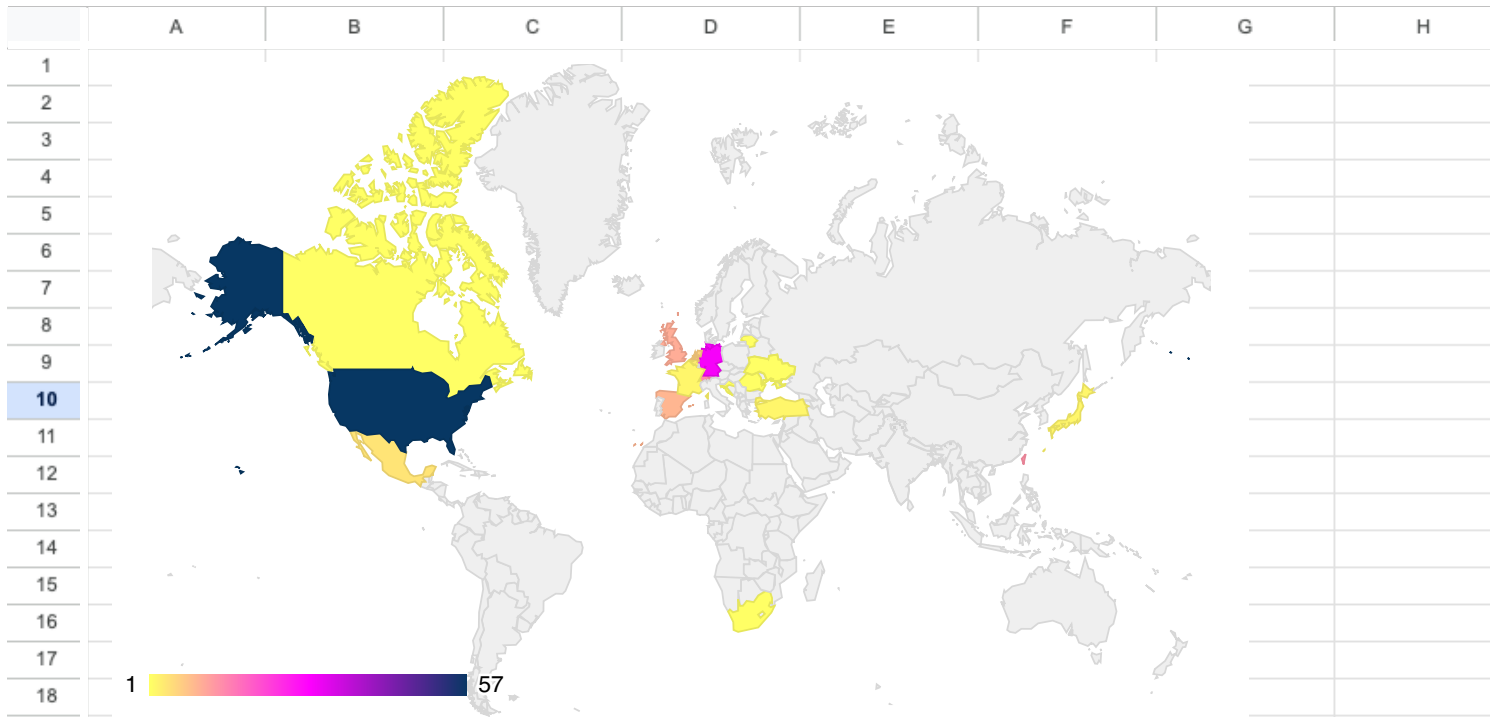


Figure 2.4: Countries contributing to CT-AI research

Influential publication sources for CT-AI research

Identification of influential journals and conferences in a field is aided by the distribution analysis of its publication sources using citation metrics [18]. Figure 2.5 displays the source

distribution of the CT-AI correlated publications. This analysis is conducted by examining the minimum number of publications and maximum citation from dataset $\mathbb{D}2$'s journals with an h-index greater than 25. The results indicate that *Robotics and Autonomous Systems* has the highest number of citations, 84, despite having only one publication. It appears that *IEEE Access* and *Artificial Intelligence in Education* journals are two of the highly cited sources in CT-AI field, with number of citations 53 and 48 respectively. Other notable journals/conferences include *Interactive Learning Environments*, *Proceedings of the ACM on Programming Languages*, *Hawaii International Conference on System Sciences*, *British Journal of Educational Technology*, *IEEE Transactions on Information Theory* and *Frontiers in Robotics and AI* with citations ranging between 11 to 22. *Kunstliche Intelligenz* and *Frontiers in Psychology* are emerging as new CT-AI research resources with 7 and 3 citations, respectively.

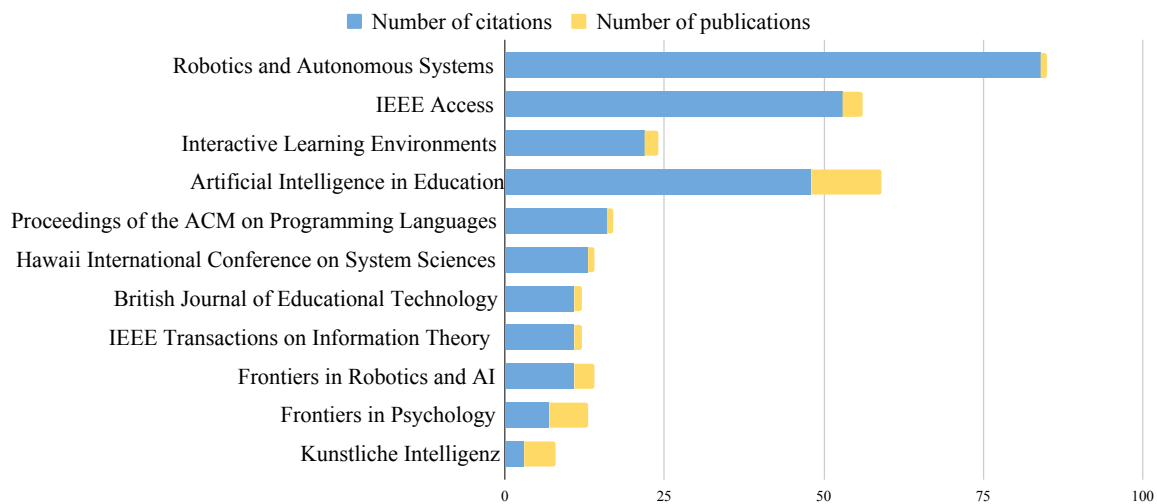


Figure 2.5: Journal and conference distribution of CT-AI research

Science mapping

Co-Word analysis

We examine the most prevalent themes using the keywords from the abstract and title of the literature from datasets $\mathbb{D}1$ and $\mathbb{D}2$ in this section by using VOS viewer's knowledge

mapping for co-word analysis [19]. We import the $\mathbb{D}1$ WoS database files into VoSViewer to perform thematic analysis by creating a keyword co-occurrence map using bibliographic data. Based on the abstract, title, and a minimum co-occurrence of five keywords in documents, we filtered out 186 keywords from a total of 2394. VOSviewer calculates the total strength of the co-occurrence links for each of the 186 filtered keywords with other keywords. For the final mapping, we chose the top 150 keywords based on their total link strength. We further filter the keywords by removing common words such as education, programming, skill, computer science, learn, and languages while mostly retaining keywords related to pedagogies, tools, and assessment techniques. Figure 2.6 shows the cluster map using the final keyword set, presenting the global picture of various programming pedagogies. We observe that CT has been the most dominant programming pedagogy since its introduction, as it has the largest node size relative to the 532 occurrences of its keyword and 1920 TLS. Considering the scattering of other nodes and the TLS of CT nodes, all research topics of CT are active as a whole. Likewise, in terms of programming tools, Scratch is the strongest node with 176 occurrences and 666 TLS.

As indicated previously, our primary focus is artificial intelligence. The red cluster highlights the main idea of this work as it contains the keyword- artificial intelligence. The cluster grouping offers a broad idea with keywords such as scratch, python, active learning, game-based learning, visual programming, gamification, digital literacy, and computational literacy. The cluster's strong link with CT, k-12, educational robotics, block-based programming, project-based learning and collaborative learning nodes indicate that AI is quite active in the programming pedagogy field. To fully evaluate the relationship between CT-AI as a theme research field and understand the cluster thoroughly, we investigate dataset $\mathbb{D}2$. We use the $\mathbb{D}2$ WoS database files to input VoSViewer and explore another keyword co-occurrence map using the bibliographic data. We identify 32 keywords from 652 total keywords using abstract text and title text, with a criterion of at least four keywords co-occurring in documents. We calculate the TLS with additional keywords for each of the 32 keywords. However, due to the limited number of keywords, we do not perform TLS-based filtering. Nevertheless, we remove keywords unrelated to pedagogy, such as digital curation. Finally, the cluster map with 30 keywords, as shown in Figure 2.7, displays the distribution of the conclusive CT-AI keyword set. The size of each node in the map represents the fre-

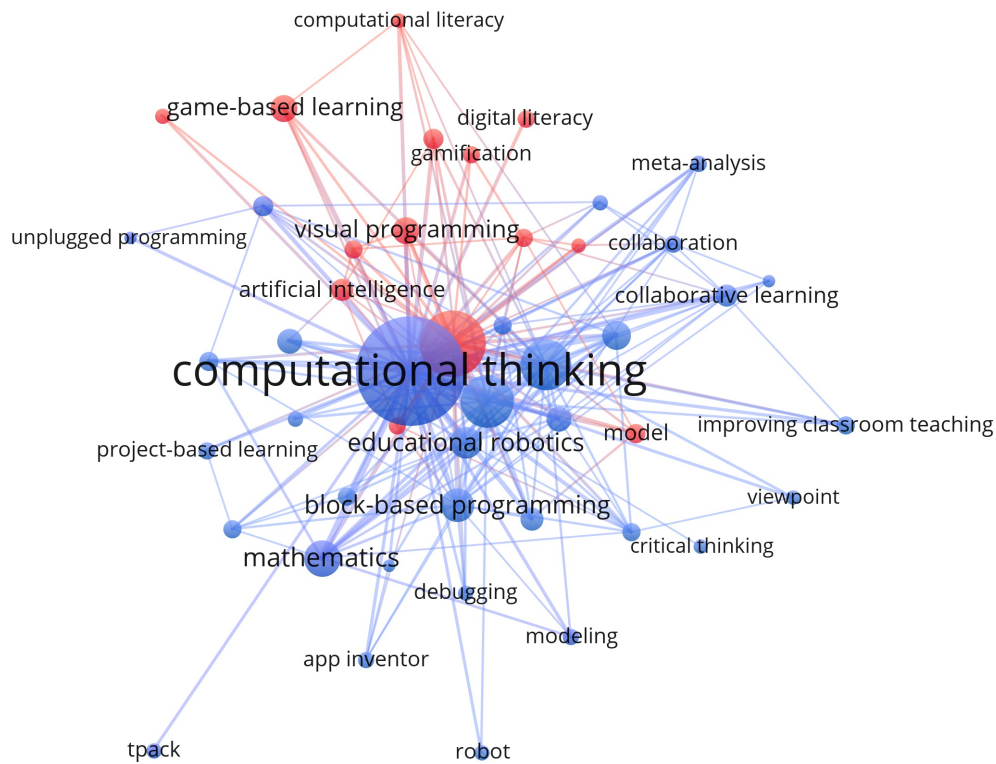


Figure 2.6: Co-word network map of the keywords in abstract and title from programming pedagogy literature

quency of keywords, and the links between the nodes indicate the co-occurrence of keywords in the same documents. Moreover, the number of co-occurrences of keywords increases as the distance between them decreases. The proximity of the CT-AI nodes to k-12, programming and education nodes confirms our suggestions that CT-AI is highly active in the education research domain. While robotics, machine learning(ML), learning analytics and mathematics education nodes are co-occurring less frequently yet, have significant TLS to AI nodes, this suggests they are emerging themes.

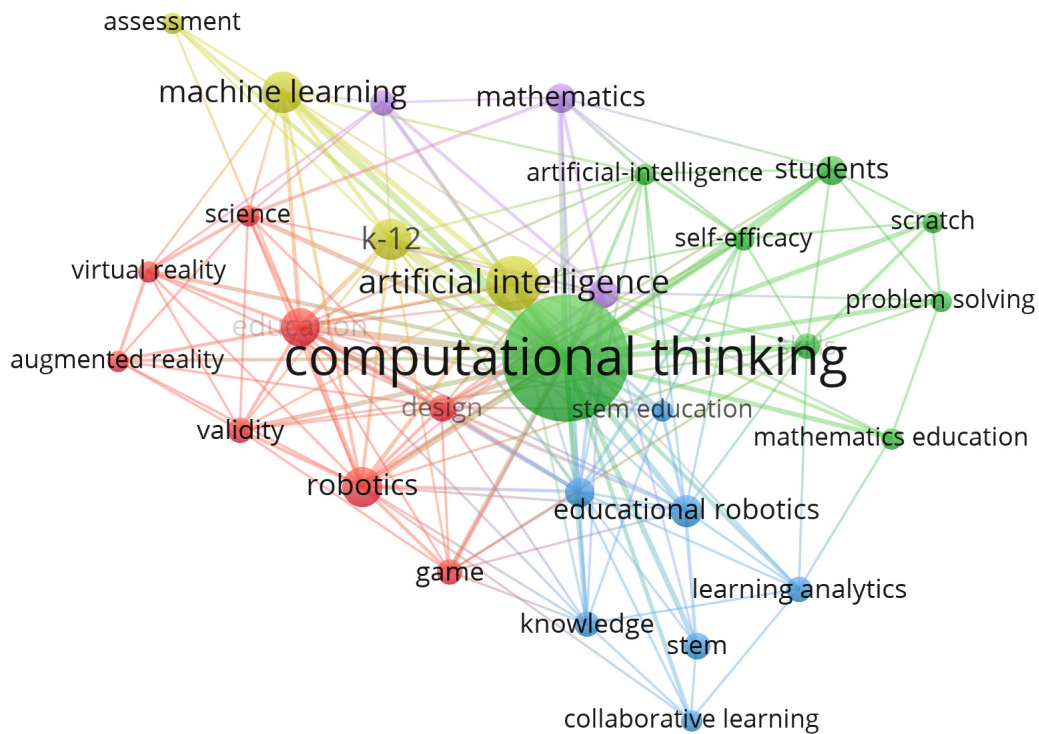


Figure 2.7: Co-word network map of the keywords in abstract and title from CT-AI literature

Additionally, the occurrences and TLS of the keywords in Figure 2.8 enable the identification of potential emergent themes. It is evident from the previous analysis that both CT and AI concepts have a strong correlation because of the high frequency of these terms. Hence we can ignore CT, AI keywords or similar variations, as they do not contribute relevant information to this detailed analysis and offsets the comparison scale. The keyword “k-12” occurs most frequently, with 13 occurrences and a TLS of 34. The keywords “machine learning” and “robotics” appear 12 to 13 times with high TLS, indicating that they are prominent topics strongly linked to the field. With a frequency of almost 8 to 11 and relatively high TLS, the keywords “education” and “educational robotics” suggest they are likely to be significant themes in the area. With 4 to 7 occurrences and a moderate TLS, the keywords “mathematics” and “mathematics education” may be less relevant in the field. They are followed by “augmented reality”, “virtual reality”, and “scratch”, with four occurrences and relatively low TLS, suggesting they are still in the experimental phase.

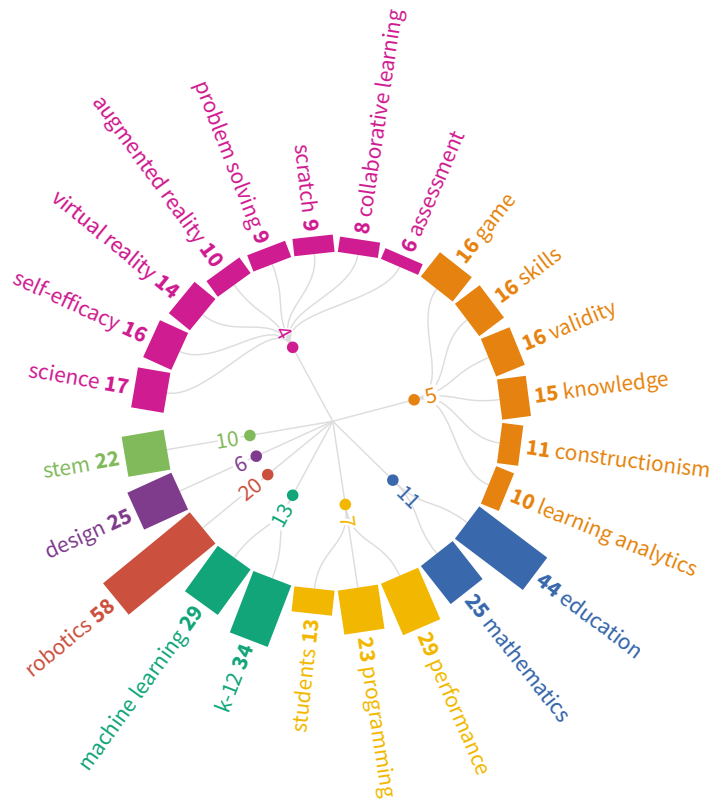


Figure 2.8: Keywords distribution in the co-word network map from $\mathbb{D}2$

As previously discussed, the co-word network mapping of $\mathbb{D}2$ also provides us with the cluster information listed in Table 3.1. Upon closer examination, the main topics addressed in the field of CT-AI are grouped under five clusters, representing the thematic areas. These clusters can be written as: *Cluster 1* contains keywords primarily related to teaching CT-AI using CT techniques and tools. The keywords “Artificial-intelligence”, “computational thinking”, “problem-solving”, “self-efficacy”, “scratch”, “skills”, and “students” suggest a focus on equipping students with the CT knowledge necessary to work with CT-AI. The keyword in the cluster traces back to the publication’s title from $\mathbb{D}2$. For instance, the keyword “scratch” relates to the study using the Scratch programming language to teach high school students the fundamentals of two of the most well-known AI algorithms [21]. *Cluster 2* includes terms related to technology use in education, specifically augmented reality, virtual reality and gamification. The keywords “education,” “science,” and “validity”

Table 2.2: Cluster information from Co-word knowledge map of the keyword terms from $\mathbb{D}2$

Cluster	Theme	Keywords
1	CT-AI in education	Artificial-intelligence, computational thinking, mathematics education, problem solving, scratch, self-efficacy, skills, students
2	Augmented and virtual reality as educational resources	augmented reality, design, education, game, robotics, science, validity, virtual reality
3	Robotics based learning methodologies	collaborative learning, educational robotics, knowledge, learning analytics, performance, stem, stem education
4	AI and Machine Learning in K-12	artificial intelligence, assessment, k-12, machine learning
5	Mathematics and Programming	constructionism, mathematics, programming

suggest how technology enriches the learning experience, allowing students to interact with content in engaging ways. For instance, the “virtual reality” keyword relates to the research on enhancing artificial intelligence of things(AIoT) CT skills by plotting image-based VR [22]. *Cluster 3* has keywords related to STEM education, robotics, and collaborative learning. The keywords “knowledge”, “learning analytics”, “performance”, and “robotics” suggest a focus on the use of robotics to improve student learning outcomes. For example, the keyword “robotics” traces to the “RoboCupJunior” study, which focuses on robotics competitions as a means of promoting STEM education through technological advancement [23]. *Cluster 4* contains keywords related to teaching ML. The terms “assessment”, “K-12”, and “machine learning” suggest a focus on the use of ML in primary to secondary school education. For instance, the keyword “k-12” and “machine learning” keyword co-occurs in the study focusing on exploring pedagogical pathways for AI education and teaching ML in K-12 classrooms [24]. *Cluster 5* has keywords related to teaching mathematics and programming. The keywords “constructionism”, “mathematics”, and “programming” suggest that CT-AI are no longer just independent fields of teaching but also a field that is being integrated into STEM concepts, especially with mathematics. For instance, the keyword “mathematics” is

associated with the study investigating the use of computer applications of AI in mathematics education [25].

Citation analysis of CT-AI publications

In citation analysis, we assess a publication's influence in the field of CT AI based on the number of citations. This analysis examines the $\mathbb{D}2$ dataset's articles, conference papers, books, and associated citation metrics. Table 3.2 lists the top publications ranked by the number of citations. We observe that the RoboCupJunior study is leading the research on CT-AI with the most citations 84 times, focusing on technological advancement through robotics competitions [23]. Furthermore, this leads to identifying Amy Eguchi as the critical author in the CT-AI field whose contribution has been influential since 2016. The PopBots research is the second most cited research work with 34 citations, focusing on designing an AI curriculum for preschool education [26]. The study on gentle introduction to AI follows, with 24 citations introducing AI to high-school students using Scratch [21]. Other research works focus on enhancing AIoT CT skills through VR, enhancing educational innovation, and leveraging deep learning models for stealth assessment in game-based learning environments with 19 to 22 citations [22, 27, 28]. Studies on designing ML applications for primary school children and educational robotics have 18 citations [29, 30]. Collectively, these research works demonstrate the potential of CT-AI in educational settings and their ability to advance STEM education.

Author citation analysis of CT-AI publications

In this author citation analysis, we examine the overall citations to an author's works to determine their influence in the CT-AI field [18]. The analysis helps to identify connections between central authors, their citation networks, and their impact. We intend to observe how the author's programming pedagogy development path transforms from their domains to their subsequent pursuit of AI. To begin with, we aim to identify the influential authors from a programming pedagogical perspective using the $\mathbb{D}1$ dataset. For the author citation map from 2606 authors, 66 connected authors meet the threshold of having a minimum of 4 documents published. Figure 2.9 displays the rendered map, with the node size correspond-

Table 2.3: Top research works leading CT-AI research with citations as on 21st February, 2023

Authors	Title	Citations in WoS	Citations in Google Scholar	Published Year
Eguchi, A	RoboCupJunior for promoting STEM education, 21st century skills, and technological advancement through robotics competition [23]	84	251	2016
Williams, R et al.	PopBots: Designing an Artificial Intelligence Curriculum for Early Childhood Education [26]	34	106	2019
Estevez, J et al.	Gentle Introduction to Artificial Intelligence for High-School Students Using Scratch [21]	24	75	2019
Lai, Y et al.	Study on enhancing AIoT computational thinking skills by plot image-based VR [22]	22	37	2021
Prendes, E et al.	Advanced technologies to face the challenge of educational innovation [27]	20	10	2021
Min, W et al.	DeepStealth: Leveraging Deep Learning Models for Stealth Assessment in Game-Based Learning Environments [28]	19	38	2015
Toivonen, T et al.	Co-Designing Machine Learning Apps in K-12 With Primary School Children [29]	18	36	2020
Evripidou, S et al.	Educational Robotics: Platforms, Competitions and Expected Learning Outcomes [30]	18	47	2020

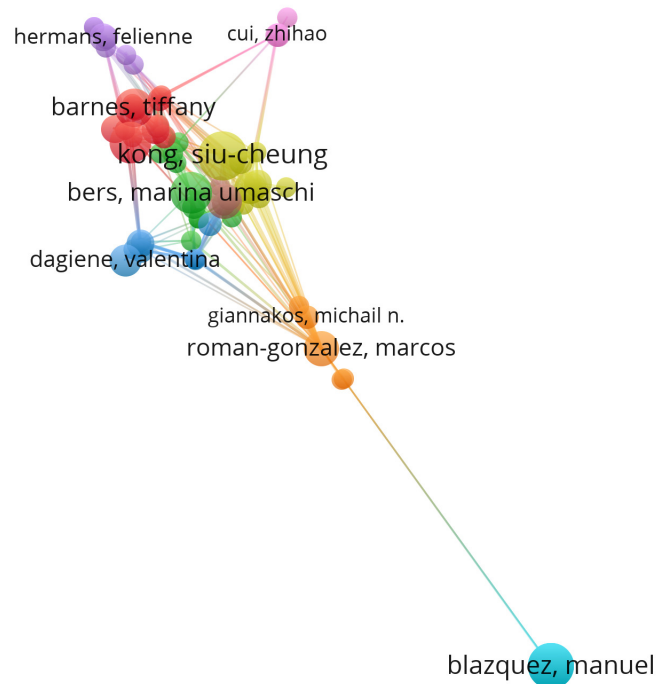


Figure 2.9: Author citation map in the field of programming pedagogy

ing to the relative number of citations and the link representing their citation relationship. It shows that these authors can be considered significant contributors to the field of programming pedagogy. To extrapolate on this, we list the influential authors of programming pedagogy, along with the number of publications and their total citations in Table 3.3. The authors are ranked based on their number of citations, with Marina Umaschi Bers having published the most works with ten publications and the highest number of citations at 540. Regardless, Bers has not diverged into the AI field. Closely followed by author Marcos Román-González who is a significant contributor in the field of programming pedagogy with a focus on CT, with eight publications and a total of 279 citations. Similarly, we evaluate the rest of the authors based on their research’s relevance to CT-AI fields by referring to their presence in the $\mathbb{D}2$ dataset. We chose the authors with the highest cumulative contribution rank and their respective contribution to CT-AI from dataset $\mathbb{D}2$ and listed in Table 2.5. Here, the research focuses on teaching CT skills and using AI-related technology and tools. We observe the general shift in trend from traditional CT with new topics, such as learning-

Table 2.4: Top influential programming pedagogy authors with most citations

Authors	Documents	Citations
Bers, Marina Umaschi	10	540
Román-González, Marcos	8	279
Kafai, Yasmin B.	4	276
Hsu, Ting-Chia	4	219
Weintrop, David	6	206
Repenning, Alexander	6	194
Kong, Siu-Cheung	12	182
Koh, Kyu Han	4	180
Grover, Shuchi	4	174
Basu, Satabdi	5	168
Basawapatna, Ashok	5	161
Robles, Gregorio	10	134

by-modeling environments, ML, and App Inventor. For instance, Roman-Gonzalez, M, who primarily worked in “Scratch” for CT, advocates AI and ML projects. Rodriguez-Garcia, JD et al. (2020) research work “LearningML” is co-authored with Roman-Gonzalez, M, an influential researcher in the field of CT since 2015 [1]. The strong citation network Roman Gonzalez has with other authors is prominent in Figure 6. The LearningML study is a new platform for learning supervised ML using CT. They also discuss the applicable AI that offers new opportunities and introduces ethical implications that need to be considered. Overall, the CT-AI research highlights the importance of an interdisciplinary approach, linking CT with science and engineering.

Table 2.5: Top CT-AI research works by influential programming pedagogy authors

Authors	Title	Publication year
Kafai, Y et al.	Survey: Artificial Intelligence, Computational Thinking and Learning [31]	2022
Hsu, T et al.	The Effects on Secondary School Students of Applying Experiential Learning to the Conversational AI Learning Curriculum [32]	2022
Kong, S et al.	From Mathematical Thinking to Computational Thinking: Use Scratch Programming to Teach Concepts of Prime and Composite Numbers [33]	2021
Román-González, M et al.	LearningML: A Tool to Foster Computational Thinking Skills Through Practical Artificial Intelligence Projects [1]	2020
Basu, S et al.	Studying the Interactions Between Science, Engineering, and Computational Thinking in a Learning-by-Modeling Environment [34]	2020
Román-González, M, Robles, G et al.	Developing Computational Thinking at School with Machine Learning: An exploration [35]	2019
Basu, S et al.	A Systematic Approach for Analyzing Students' Computational Modeling Processes in C2STEM [36]	2019

2.5 Conclusion

In this chapter, we provide a comprehensive bibliometric analysis of the convergence of computational thinking and artificial intelligence (CT-AI) literature as a facet of the “Programming Pedagogy” from 2009 to 2023. We utilize the Web of Science database to extract two datasets on programming pedagogy and CT-AI analysis. Subsequently, we apply performance analysis and science mapping methodologies to investigate these datasets. A thorough overview of the CT-AI field’s development is provided through measures including publications, multidisciplinary analysis, geo-contribution, and journal leadership, which are examined as part of the performance analysis. In addressing RQ1, we discovered an exponential increase in CT-AI publications in 2017, with “Education Educational Research” being the most prominent research area. Correspondingly, the RQ2 investigation demon-

strated that the United States of America contributes the most to this field globally. The top-cited journal *Robotics and Autonomous Systems* has the most significant contribution. According to the publication citation analysis, the RoboCupJunior study by Amy Eguchi is the most cited publication [23]. Additionally, citation analysis indicates that Marcos Román-González is among the most influential computational thinking authors contributing to research progress in this area. In response to RQ3, thematic area identification using co-word analysis highlights several initiatives and programs that promote CT-AI education at various levels of K-12 education. The clusters allowed us to establish connections with emerging fields such as robotics, virtual reality, and machine learning. Ultimately, this study aims to bring CT-AI research to the forefront of computer science education research and serve as an essential reference for future studies.

Acknowledgments

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), [funding reference number PROSC/556957-2020] for facilitating the execution of this project.

Chapter 3

Evaluating the efficacy of Python Scaffolding to teach Computational Thinking in Remote Schools

All of this chapter has been accepted in the HCI international 2023 conference as the following peer-reviewed article:

- Kamak, LP., Mago V. (2022). Assessing the impact of using Python to teach Computational Thinking for remote schools in a Blended Learning environment

As the demand for programming in STEM education continues to grow, computational thinking skills are becoming even more important for young students. In order to optimize the learning process while simultaneously introducing advanced programming language to a young audience, we designed visual scaffolding blocks on Python as a teaching aid. The scaffolding emphasizes the flow of execution, similar to block-based programming such as scratch. This study lays out a comprehensive curriculum aligned with Ontario learning outcomes with the mission of nurturing computational talent and removing geographical barriers to build sustainability in First Nation Schools in Northwestern communities of Canada.

Keywords : Computational Thinking, Computational Literacy, Blended Learning, K-12 Education, Learning and Assessment, Programming, Learning-by-design, Scaffolding

3.1 Introduction

Programming has become an integral part of modern education. It is now more imperative than ever for students to learn the fundamentals to adapt to science, technology, engineering and math (STEM) related fields. However, teaching programming can be challenging due to the complexity of the subject. Educators are working towards developing engaging resources to help students understand and apply their learning. Our investigation began by examining the importance of designing teaching resources that incorporate mathematics through programming. One of the major driving forces of this study came with the adoption of Coding in the Ontario Mathematics Curriculum for grades 1 to 8 in 2020 [37]. The curriculum update was considered a transformational change by many [38]. There was broad consensus that this curriculum update affects nearly 4,000 schools, roughly 1.4 million students, and more than 80,000 teachers. As Ontario is implementing an explicit curriculum direction for remediating mathematics concepts, it calls for altering the “representational infrastructure” available to schools [39, 40]. Explicit coding requirements were added to the algebra section of the Ontario curriculum in 2020 for grades 1 through 8 to prepare students from solving problems to generating computational representations of mathematical situations [41]. Gadanidis et al. exhibit suitable ways to represent mathematical situations in code supported by the reformulation in the Ontario context [38]. For instance, they demonstrate how inequalities can be reformulated using a coding environment by using a conditional statement. The coding expectations are as follows in Table 3.1.

Table 3.1: Coding expectations in the Ontario mathematics curriculum for grades 1 to 8 [38]

Grade	Coding Concepts
1	Sequential events
2	Concurrent events
3	Repeating events
4	Nested events
5	Conditional statements
6	Efficiency of code
7	Defined count and/or subprogram
8	Analysis of data

Our considerations for remediating mathematics using programming, given this updated requirement, were based on popular studies. As we analyzed the pedagogical elements that might benefit students most, we considered the approach necessary to ensure success while factoring in the implications of limited resources. Among the many pedagogical ways to teach programming, CT has recently become one of the most widely used. In K-12 education, Computational Thinking (CT) is increasingly being integrated into the curriculum as an essential 21st-century skill [42, 43]. CT primarily develops skills in algorithmic thinking, abstraction, problem decomposition, pattern recognition, logical thinking, and data representation [2]. Numerous studies suggest that CT skills are essential to developing cognitive skills for students that may apply to everyday problems [44, 45]. It teaches students to be comfortable with ambiguity while constructing computational models using technology. Schools are now utilizing coding and CT as a tool to teach core subjects like math, science, and language arts. However, educators need a standard curriculum to teach CT to students as it is still in the early stages of curation [42].

For our curriculum development, we chose Python as the medium for the programming language for students to learn. Python programming language is a popular coding language used in many educational settings. In a recent study, researchers found that students who were acquainted with Python demonstrated a greater understanding of computational concepts than those who were taught other languages, such as Scratch, a block-based visual programming environment [46]. To optimize the learning process, we introduced Python alongside the scaffolding teaching aid that is similar to a block-based language in our curriculum, which are discussed in Section 3.3. As we developed the course, we faced the challenge of meeting the needs of students from remote schools. School students had limited access to internet resources and computer labs compared to their city counterparts [47]. During the school days, they commute for one hour to school every day, and more during harsh winters. Attending the course was further complicated by the rapidly changing COVID-19 scenarios. In order for the students to be successful, they needed a dynamic environment that was seamlessly integrated into their current environment. After evaluation, we chose blended learning (BL) as the most appropriate delivery method for this course. The methods and tools used are further discussed in Section 3.3.

The goal of this thesis is mainly twofold: to verify the cogency of the proposed curriculum

assessment tool for teaching programming and CT efficiently, given sparse resources and tight timelines. Can scaffolding help students acquire CT skills and become computationally talented? More specifically, we look at the following research questions:

- RQ1: What is the student’s level of preparedness for a blended learning environment?
- RQ2: What is the effectiveness of the Python scaffolding technique for teaching computational thinking?

Overall, this study aims to develop a content-based accelerated environment for computer science education in middle schools using widely accepted CT learning methodologies. Furthermore, the study focuses on methods to optimize the learning process for students in remote middle schools. We evaluated the results for the efficacy of the curriculum, the level of students’ learning efficiency, and their interest in computational thinking at the end of the course. These results are further discussed in Section 3.4.

3.2 Background

Based on much research, there are multiple definitions of CT. We chose to refer to “computational thinking” using the following definition from BBC: “Computational thinking allows us to take a complex problem, understand what the problem is and develop possible solutions. We can then present these solutions in a way that a computer, a human, or both, can understand” [2]. It is a problem-solving process that includes the following components.

- Decomposition: Breaking down a complex problem into smaller and manageable parts
- Pattern recognition: Looking for similarities within problems
- Abstraction: Ignoring irrelevant detail and focusing on the relevant information
- Algorithms: Developing a step-by-step solution to the problem
- Evaluating solutions: Ensuring that the solution satisfactorily and efficiently resolves the problem [2].

Content-based Acceleration in Middle School Computer Science Education

Research suggests that “Computationally talented” students can be identified in middle school and can advance through the CS education more quickly than typical students. Marcos et al. empirically evaluated the predictive validity of the Computational Thinking Test (CTt) utilizing Spanish students. They suggest that the developing computing curriculum must consider these individual variations in computational ability [48]. There is evidence in their study suggesting that switching from ‘block-based’ to ‘text-based’ programming languages can be used to identify computationally talented students in middle school. On this basis, we sought to introduce scaffolding techniques to them using visual aids and transform Python into a block-based format to accelerate learning in a limited time and facilitate efficient learning [49].

Another similar research framework that has been established in this field is the Metacognitive Empowerment by Computational Thinking (ME-CoT) learning module [46]. It has been recognized for cultivating computational thinking using programming skills to encourage metacognitive awareness in biology students. The module has high stability reliability with a consistency of the measures, or Pearson’s $r = 0.974$, to determine stable dependability because the study sample was less than 30 pupils.

We adapted the research work of the following surveys: (i) The Callysto CTt assessment tool Questionnaire developed by Cybera and the Pacific Institute for Mathematical Sciences, with funding from the Government of Canada to develop modules that promote data literacy and computational thinking [50]. (ii) The blended learning questionnaire by Poon, J. (2013) investigates the benefits of institutionalized blended learning to student’s educational experiences [51]. The study helps to overcome the substantial barrier in the synergistic learning integration of CT with STEM education since there needs to be a greater understanding regarding the difficulties students face when using an integrated curriculum and their learning styles.

The potential benefit of this study is the increased computational thinking capabilities and practical programming skills at a young age. Computational skills are not only limited to programming but also help develop well-rounded problem-solving skills required in other

fields [42]. The research also aims to encourage the development of computational abilities to increase interest in coding. A recent study on CT showed that children were more interested in programming after developing CT skills [42]. As members of a remote First Nation, Metis and/or Inuit (FNMI) communities in Northwestern (NW) Ontario, the participants often need more opportunities to develop these skills [52]. The FNMI benefits of this research would be to provide objective results that may allow for future CT skill-building options amongst other remote FNMI communities.

3.3 Methodology

The proposed research evaluates the grade 7 and 8 student's performance and receives feedback for the programming course using computational thinking methods. A customized blended learning (BL) environment with online and in-person classroom teaching materials was provided on Desire 2 Learn (D2L), an online learning management system. The research was executed in two phases - the curriculum development and the evaluation. During the development phase, curriculum content was designed based on grade 7 and 8 Ontario curriculum guidelines. We evaluated the student's CT abilities during the evaluation phase and received feedback on the BL course delivery method.

Development Phase

The course "Coding with Python" offers basic Python programming integrated with block-based visual aids and widely used CT learning methodologies. The research team provided the teachers with instructional materials and visited the school in person to build relationships with students and communicate effectively from time to time. All the course materials are hosted on the D2L platform by Lakehead University. D2L is an emerging platform for post-secondary education and provides students with the necessary experience to navigate course activities. The resources included programming video tutorials, worksheets, and an instructor's manual. Teachers conducted the courses at the School Site using the materials and instructor's manual. Activities for the school's teachers and students were planned for one academic year. Figure 3.1 illustrates an example of a Python program that uses visual

Which algorithm gets the corn to the beans and squash using the red path?



A) `while(corn_collects_all == true):`

`if(block != 'chickens'):`

`move_forward(1)`

`turn_north()`

`move_forward(1)`

`turn_east()`

B) `while(corn_collects_all == true):`

`if(block != 'chickens'):`

`move_forward(2)`

`turn_east()`

`move_forward(2)`

`turn_north()`

Figure 3.1: Curriculum sample - Chapter 7: Computational Thinking with Cultural Context: Python-based block code with a visual aid to highlight the flow of execution

blocks similar to Scratch to teach students programming concepts. In order to summarize the course materials in a book format, we developed a lab manual with 150 pages and seven chapters, including interactive computational thinking activities. The seven chapters in the course are: Introduction to Programming, Data Types, Expressions, Conditions, Functions, Loops, and Computational Thinking with Cultural Context. During the course, suggestions were offered on handling errors and debugging to avoid frustration when coding. Since the schools in NW Ontario are located remotely and have limited internet access, online and offline materials were provided to students to enable a smooth transition.

Evaluation Phase

The pilot survey was administered in parts towards the end of the course. The coding course is separate from the survey and is part of the school's math curriculum. The survey was a research opportunity provided to students after learning the python course. The survey was designed to evaluate the computational thinking that the students are expected to demonstrate at the end of the course. This survey was entirely voluntary, and students had the option to participate or not. Participation or lack thereof in the survey did not affect any grades for the students. The teachers collected the survey results and shared them with DaTALab, Lakehead University. Mr Corey Dagenais, the Armstrong Public School

Principal, approved the school's participation in the course and provided permission for the Lakehead researchers to conduct the study at Armstrong Public School. A raffle was held for participants who participated in the surveys. The prize was an educational toy of approximately \$25 appropriate for students. The Python assessment and Computational Thinking information were introduced at the course's beginning. Detailed survey information was available in the participant information letter provided to the students seven days before the first survey. Before each survey, the students were required to confirm they had read the information letter and provided consent by checking the box affirming their understanding and willingness to participate in this research. The students were declared to have the capacity to provide consent for themselves. As per article 4.6 of the TCPS 2, the capacity to consent of student participants is not based on age but on the likelihood that they or their group might benefit from the study [53]. In addition, participants must be able to understand the risks, benefits, and nature of the research; the potential risks were very low. These criteria were clearly provided in the information letter and consent form. However, due to poor internet connectivity, paper consent forms and surveys were provided, and the students provided consent using the consent form. In this situation, paper copies of the information letter were also available to ensure participants were fully informed.

Research participants

A total of 12 students from Armstrong Public School participated in the survey from the Whitesand and Gull Bay First Nations communities. The survey was taken by grade 7 and 8 students (ages 12-14) attending the "Coding with Python" course at Armstrong Public School. A total of 27 students were enrolled in the course, but the pilot survey was administered to 12 students due to low attendance on that particular day. Teachers and mentors aided them during this study, and they strictly adhered to the protocols for COVID-19 safety. Members of vulnerable populations are involved as participants in this community-based research. Our participants include students who identify as First Nation, Metis and/or Inuit. The study involved Aboriginal (FNMI) people as an "incidental" representative group in research. This research has been approved by the Research Ethics Board (REB) of Lakehead University. Appendix 6.1 shows the REB application form and approval letter.

Instruments

The survey was conducted as three-part surveys - S1, S2, and S3 over the course duration. Participants had 20 minutes to complete each survey. Most survey questions are multiple-choice and ranking questions using the 5-point Likert scale, while some problems require long-form answers in the computational thinking section of the survey. Our survey is adapted from the following two questionnaires; the Callysto CTt assessment tool Questionnaire (CTQ) and the Blended Learning Questionnaire (BLQ) [50] [51]. The survey questions will mainly assess their knowledge in programming concepts, computational thinking skills, and learning interests. Three surveys adapted from the Blended Learning Questionnaire (BLQ) and the Computational Thinking Questionnaire (CTQ) were conducted by the research team to collect the test scores.

Survey Translation

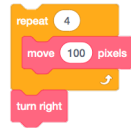
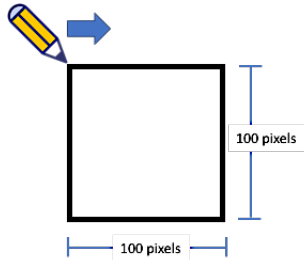
Despite Python's readability, many students needed help to identify the flow of execution of statements. Consequently, we conjectured that scaffolding techniques using graphic visual aids would enable students to comprehend Python program flow in the CT section of our survey and curriculum. The original CTQ Scratch code was replaced with Python-based block code, and the questions were intricately reviewed and tailored to the student's grades. The elements measured in the translated CT section are - digital literacy, attitude towards computational thinking, previous coding experience, data literacy, spatial CT skills, problem-solving skills, and python programming skills. Figure 3.2 displays the sample question from the original survey with Scratch code and the translated survey with Python code respectively. The complete translated CTQ section of the survey is available at Appendix 6.2.

In the original BLQ, students took full-time and part-time courses in higher education [51]. We adapted it to accommodate middle school student's readability levels and demographics. In the BL section of our survey, we manually reworded the questionnaire and instructions with context more suitable for 7th and 8th-grade students. In addition to collecting feedback on our course delivery method, the BLQ questionnaire also examines the student learning experience. The following are some examples, with the translated text



Question 2:

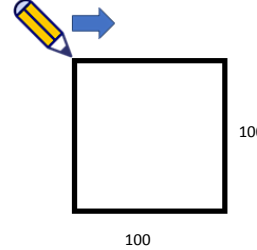
The instructions below should make the pencil draw the square **once**. The pencil starts at the upper-left corner position and moves in the direction of the arrow.



What is the **error** in the instructions?

- (a) The "repeat" loop should repeat 2 times.
- (b) The "move" block should be outside the loop.
- (c) The "turn right" block should be inside the loop, after the "move" block.
- (d) The "turn right" block should be inside the loop, before the "move" block.

The instructions below should make the pencil draw the square **once**. The pencil starts at the upper-left corner position and moves in the direction of the arrow.



```
for x in range(4):
    move_forward(100)
turn_right()
```

What is the **error** in the instructions?

- (a) The "repeat" loop should repeat 2 times.
- (b) The "move" block should be outside the loop.
- (c) The "turn right" block should be inside the loop, after the "move" block.
- (d) The "turn right" block should be inside the loop, before the "move" block.

Figure 3.2: Comparison of the original Callisto CTt survey item sample and the translated CT survey item with visual blocks for (a) Health organizations and (b) Leaders of the top-10 COVID-19 resilient countries.

Question sample 1 from the original BL survey : *“In comparison to the traditional classroom teaching, how would you describe the QUALITY OF FEEDBACK on coursework assessment that is received if the module is taught by blended learning?”* [51].

Question sample 1 from the translated BL survey : *“How is the QUALITY OF FEEDBACK on your exercises in blended learning when compared to classroom teaching?”*

Question sample 2 from the original BL survey : *“In your what are the advantages of using blended learning as a teaching method? (Please identify up to THREE advantages.)*

- Convenience of not having to come to campus as often
- Flexibility of being able to complete assignments any place/any time
- It is a requirement for course/module
- It was the only available option course that fitted into my timetable
- Job responsibilities and other commitments make it difficult to attend face-to-face classes
- I have a disability that makes travel inconvenient”[51].

Question sample 2 from the translated BL survey : “*What are the advantages of blended learning to you? Choose up to three.*”

- The convenience of not having to come to **school** as often*
- The flexibility of being able to complete assignments any place/any time*
- It is a requirement for course/module*
- It was the only available option course that fitted into my timetable*
- Home** responsibilities make it difficult to attend face-to-face class*
- External factors such as poor weather or COVID restrictions cancelling face-to-face classes.***
- I have a disability that makes travel inconvenient”.*

Readability Tests

Once the survey was translated, we performed preliminary reviews to see if the survey was easy to understand and appropriate for grades 7 and 8. We gathered text statistics and performed readability tests on the survey questionnaire. The following grade-level indicators assess and equate the readability of the text to the school grade-level system. With the grade level score, text can be measured to determine the grade level at which the survey content is appropriate for the student. The readability tests listed below are utilized in this study:

$$FleschKincaidGradeLevel = 0.39 \left(\frac{words}{sentences} \right) + 11.8 \left(\frac{syllables}{words} \right) \quad (3.1)$$

$$GunningFogScore = 0.4 \left[\frac{words}{sentences} + 100 \left(\frac{complexWords}{words} \right) \right] \quad (3.2)$$

$$ColemanLiauIndex = 5.89 \left(\frac{characters}{words} \right) - 0.3 \left(\frac{sentences}{words} \right) \quad (3.3)$$

$$AutomatedReadabilityIndex = 4.71 \left(\frac{characters}{words} \right) + 0.5 \left(\frac{words}{sentences} \right) \quad (3.4)$$

A key difference between Coleman Liau Index, Automated Readability Index and the other tests is that they are calculated with the count of characters, words, and sentences. Unlike other tests, which also assess complex words and syllable counts [54]. The corresponding results and discussion are provided in Section 3.4.

Procedures

Data Collection To maintain confidentiality, the information was de-identified after data collection. Individual information was not collected; instead, students were allotted random ID numbers. This student ID number linked all the three submitted surveys. The raw survey results were only used to find aggregate information and correlation between CT training and programming scores surveyed.

Data Analysis To begin, the data set was examined to see if there were any missing data. As a result of missing values, the figures might not add up to 100%, mainly because participants preferred not to answer certain questions. As the variables from the 5-point Likert scale responses were coded to measure the attitudes and beliefs of the participants, they were analyzed with a variety of methods, such as non-parametric tests, quantitative and qualitative comparative analysis. All the closed-ended responses were analyzed using traditional descriptive statistical techniques. Charts were created to visualize and interpret the student's attitudes towards CT and programming-related elements. The `item_difficulty_index` was assigned to the CT questions ranging from easy, medium, difficult, and very difficult. We graded them as good, fair, or poor based on how closely the students answered the correct answer. We calculated the frequency distribution of these variables and observed that the data was not normally distributed. Therefore, non-parametric tests were conducted to determine the relationship between Python Programming, CT and course performance [55].

Table 3.2: Descriptive text statistics of the original BL survey and translated BL survey

Text Statistics	Original Survey	Translated Survey
Sentences	25	26
Words	336	278
Complex Words	47	35
Percentage of Complex Words	13.99	12.59
Average words per sentence	13.44	10.69
Syllables in an average word	1.65	1.55

3.4 Results and Discussion

Curriculum and assessment have been well received by students and teachers, and have been validated by education experts. In order to further discuss the results, we have divided them into qualitative and quantitative analyses.

Qualitative Results

Readability Tests

We began by calculating text statistics for the original BLQ survey. In the survey, there are 25 sentences in original text with 336 words, 47 complex words in this text with 13.99% of complex words, 13.44 words in an average sentence, and there are 1.65 syllables in an average word. We did not include a few questions we deemed redundant and whose results did not contribute to our research question. The text statistics for both the original and translated survey is provided below in Table 3.2.

Furthermore, it is essential to note that the students were exposed to complicated words throughout the course and had knowledge of their appropriate definitions. For instance, the students were familiar with the complex terms ‘Computational Thinking’, ‘Decomposition’, ‘Abstraction’ and ‘Blended Learning’. This was also supported and reinforced by including the definitions of the complex words in the survey. For the following readability test results, we consider 7 to 8 optimum and 12 and higher to be very complex for students. According to our observations in Table 3.3, the original BLQ was designed for undergraduate students

Table 3.3: Readability test scores of the original BL survey and translated BL survey

Readability Tests	Original Survey	Translated Survey
Flesch Kincaid Grade Level	9.163	6.976
Gunning Fog Score	10.053	8.554
Coleman Liau Index	10.512	8.600
Automated Readability Index	12.300	11.357
Average Score	10.507	8.872

and had an average readability score of 10.507. We achieved a readability score of 8.872 after simplifying the text and reducing the complexity of the questions. It demonstrates that the survey has been re-designed for the middle school students of grades 7 and 8, making it more suitable and appropriate for them.

Quantitative Results

Course Performance

Our analysis shows that students who received Python-based instruction also gained adequate computational thinking skills required for programming. After receiving the course, a majority of participants (83%) performed well in applying basic CT concepts like algorithmic thinking, decomposition of complex problems, spatial reasoning, and abstraction, as shown in Figure 3.3. As the difficulty scale increased, 50% could successfully apply pattern matching and debug errors. Overall, their performance is fair in the programming section. Learning computational thinking in school was exciting for 50% of them. Despite not having any previous experience with blended learning, most participants (83%) were still able to engage with both online and offline material and learn the basics of programming. When encountering an error in their code, 41.7% feel frustrated and want to give up, and 50% require guidance when debugging the error. Our study found that 4 out of 12 students are computationally talented, with three scoring well above our expectations - 89.58%, 88.54%, 80.21% and one scoring 73.96%. These statistics aid the school board in qualifying and implementing future STEM programs for the school.

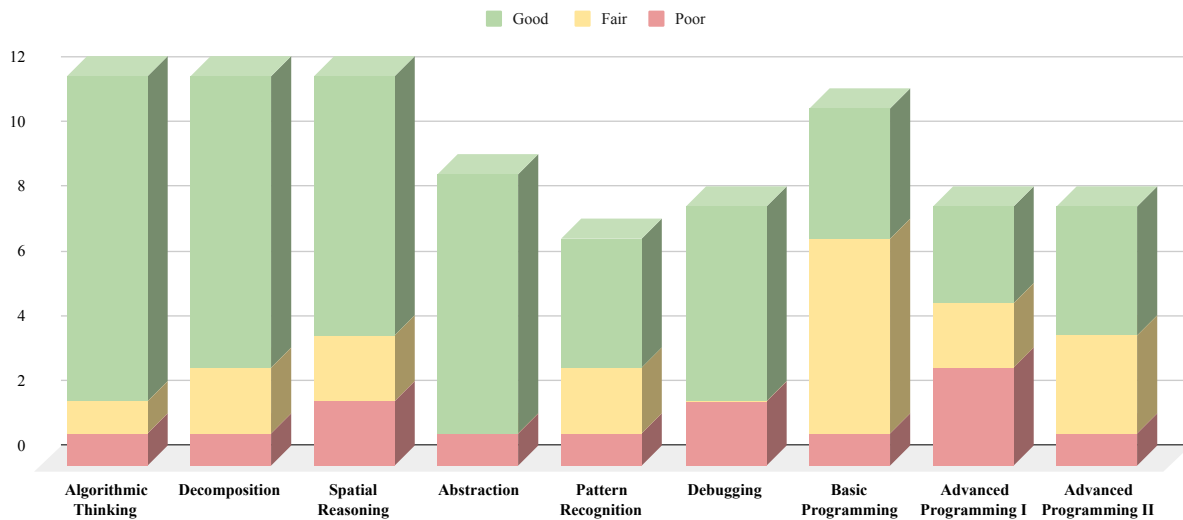


Figure 3.3: Graph depicting the student performance metrics in CT and programming skills

Computational Thinking Questionnaire Analysis

Below are the features captured from the CTQ analysis section of the survey:

Digital literacy: The student comfort level with technology and their agreement with the digital literacy statements are shown in Figure 3.4. A large majority of the students had a positive experience in handling technology and agree that they enjoy and are comfortable navigating it. However, 50% either disagree or strongly disagree with the confidence statement in troubleshooting a technical issue. This shows that the students have a sufficiently clear understanding of operating technology but require support in handling issues with the computer. It is relevant to note that diagnosing computer problems can be challenging even for undergraduate students [56]. Additionally, they must understand the underlying principles of how computers work to apply their knowledge to the problem at hand [57]. It is a complex process that requires patience, analytical thinking, and problem-solving skills.

Experience in Computational Thinking: Regarding algorithmic thinking, students are neutral towards the idea of creating their own algorithm; however, most of them stated that they could follow an algorithm when given and explained the steps required to follow. A

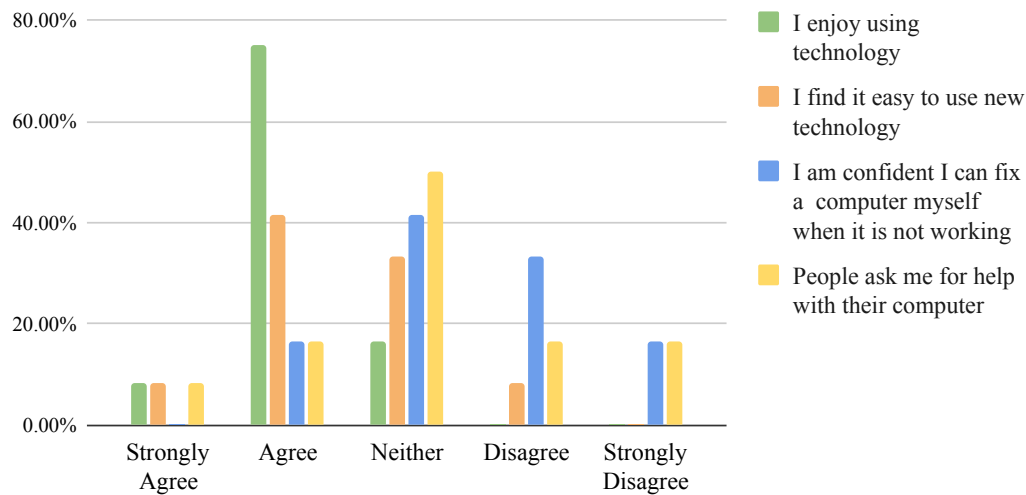


Figure 3.4: Student's perception of their comfort level with technology

key step towards problem-solving is problem decomposition, and one positive finding from this graph illustrated in Figure 3.5 is the degree of confidence. Predominantly students (58.3%) either strongly agree or agree that they can use decomposition independently, while only 8.3% disagree. Similarly, when it comes to pattern detection, the majority of students (58.3%) think that they can recognize patterns in problems, while just 8.3% disagree. This view is in line with the student's course performance, where 83% were adept at basic CT concepts, and half were able to apply pattern matching successfully. The normal distribution of the results is desirable for their first-time programming course because it clearly indicates their range of abilities. Moreover, this allows for tracking students' progress over time and identifying computationally talented students.

Interest in Computational Thinking: Student's attitudes towards their comprehension of computational thinking education and its purpose are depicted in Figure 3.6. A clear majority of students assert that mastering computational thinking is essential and that the enthusiasm for learning is generally favorable. In this context, it is interesting that students feel mostly neutral and not well-equipped to teach CT to their peers. This could be related to learning environment factors, where 50% of students think more interaction in a blended learning environment, and 41.7% felt it is decreased. There appears to be a need for more

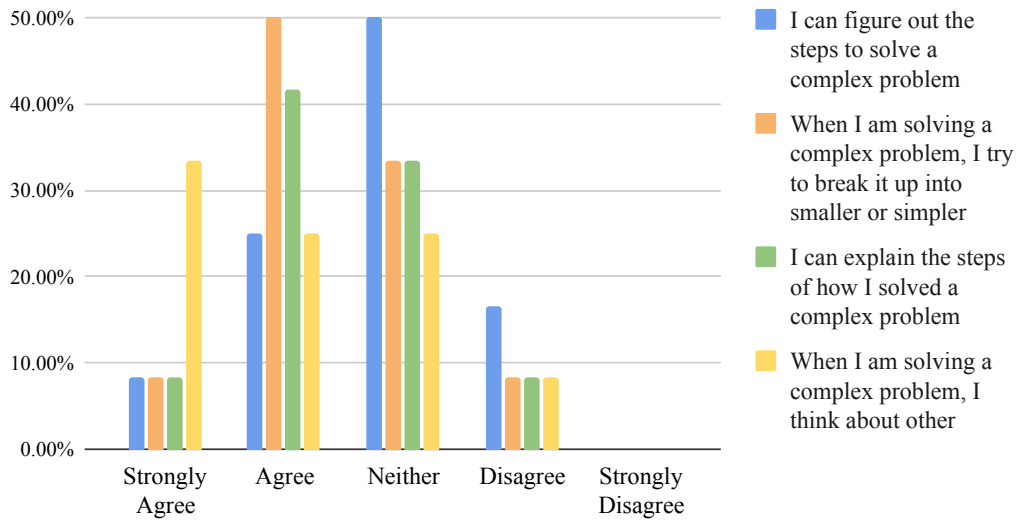


Figure 3.5: Student’s self evaluation of their computational thinking skills

resources for peer teaching or collaborative learning, an essential aspect of future training approach considerations.

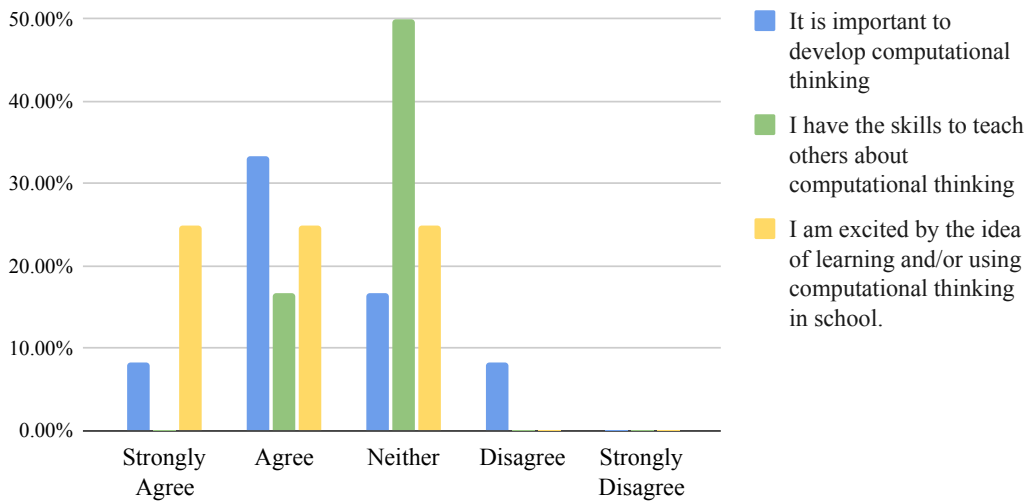


Figure 3.6: Student attitudes towards computationally thinking learning interests

Programming Experience: We observe that the results presented in Figure 3.7 show a much lesser level of confidence regarding programming in students. Although there is some

discomfort in writing code to solve problems as well as debugging independently, on the whole, the students find the challenge of coding primarily positive. The above observation demonstrates that students understand the subject matter in general but need more comprehensibility when translating algorithms into code and need guidance and motivation when they encounter difficulties with coding.

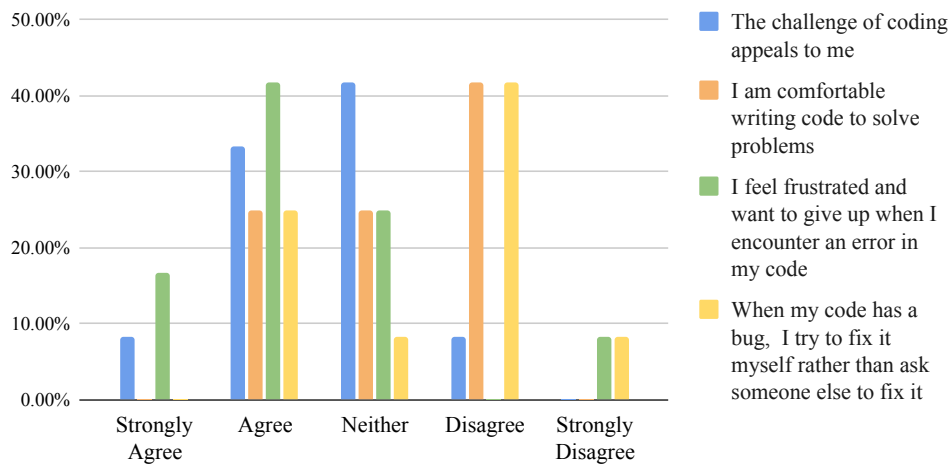


Figure 3.7: Student attitudes towards programming experience

Blended Learning Questionnaire Analysis

Below are the variables captured from the BLQ analysis section of the survey:

Advantages of Blended Learning: Figure 3.8 displays the advantages the students believe are most beneficial for their course delivery (they could choose up to 3 items). Blended learning has several advantages to students, the most significant of which is flexibility, allowing students to do their assignments anywhere and at any time. In addition, the second most common reason cited was that this was their only option. Furthermore, the recent COVID-19 pandemic and frequent adverse weather in NW Ontario further complicate travel restrictions. Although there is only a tiny percentage, it is pertinent to note that there are students with home responsibilities that make it challenging to attend in-person classes and students with disabilities that find it difficult to travel.

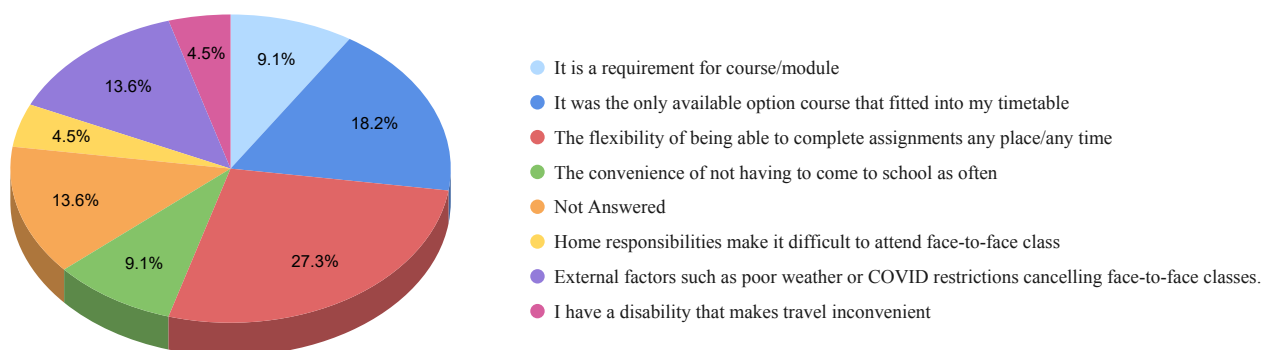


Figure 3.8: Student’s subjective advantages in blended learning

Blended Learning Experience: The overall outcome of the student’s blended learning experience was positive as observed in Figure 3.9. Most students confirmed they are comfortable with the blended learning environment and the lab manuals are easy to follow and adequate to complete their lab work. All students, except for 16.67%, choose to enroll in another blended learning course with us in the future. This indicates that they were satisfied with the course experience, that the teaching delivery was compelling and engaging, and provided a quality learning experience.

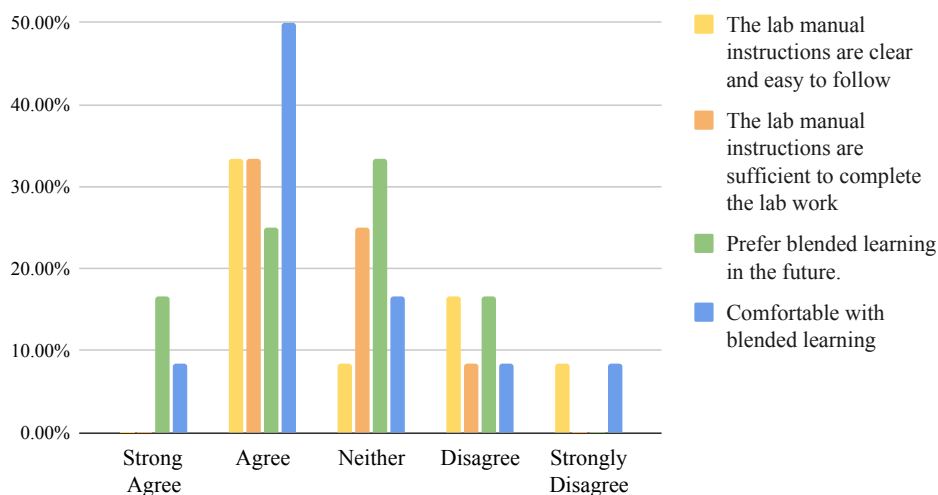


Figure 3.9: Student feedback on their blended learning course experience

Comparison of Blended Learning to Face to Face Learning Experience: Unlike face-to-face classroom learning, blended learning has enhanced interaction between students and their teachers. The majority of students believe that in blended learning, the interaction with teachers and classmates and the quality of feedback on exercises have increased. These results can also be observed in Figure 3.10.

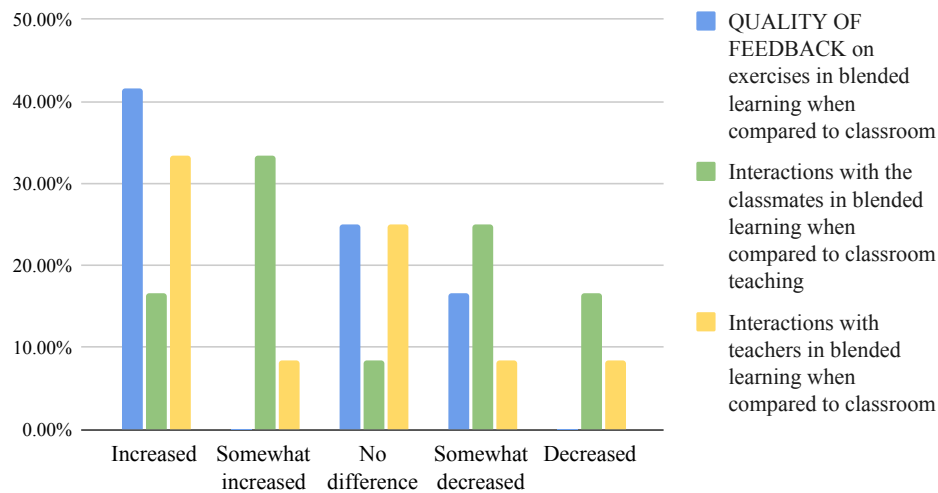


Figure 3.10: Comparison of student’s experience in blended learning and in-person classes

Correlation Analysis of Computational Thinking Questionnaire Elements and Course Performance

The computational thinking-related elements obtained in the CTQ section of the survey using the Likert scale were used in Spearman’s correlation for qualitative analysis. The pertinent CTQ variables considered in this experiment are Course Performance, Programming Experience, CT experience, and CT interest. CT experience and CT interest were merged into one entity called Computational Thinking because they are dependent variables. The Course Performance variable was derived from the student’s performance in Python Assessment based on CT and programming concepts. Using the `item_difficulty_index` in our view for CT evaluation, we weighted each question according to its difficulty level. Then using the appropriate weights assigned to categories, we calculated the score. In order to compare

the final score with the variables derived from the Likert point scale, it was normalized to a 5-point scale. For absolute values of r , we consider 0.40-0.65 as moderate [58]. Figure 3.11 illustrates a moderate correlation between the student's overall experience in Computational Thinking and Programming with an r -value of 0.64. This closely relates to our supposition that learning Python allows students to gain CT skills. Furthermore, their course performance is also moderately correlated with their CT and programming experience.

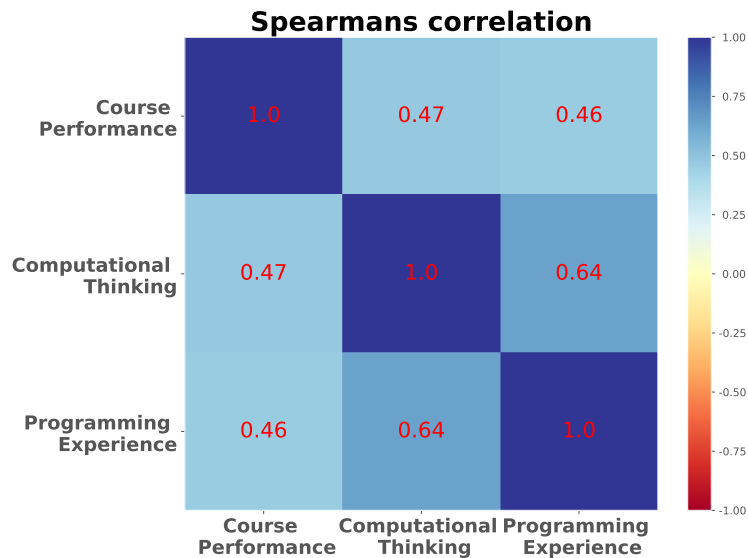


Figure 3.11: Spearman's correlation analysis of computational thinking related elements

This survey data is a benchmark for our curricula and the efficacy of the delivery method. In future, we will use these results to compare and measure the progress of student's improved computational thinking skills gradually over the years.

3.5 Conclusion

Students responded well to our scaffolding intervention technique. Our innovative teaching method of using Python integrated with block-based visual aids in K12 education effectively combines the best of both block and text-based programming. Python programming provided a more direct learning experience, while block-based visual aids helped give a visual representation of the concepts being taught. In the end, the students were enthusiastic about

learning CT, more familiar with coding syntax, acquired experience in debugging errors and were willing to learn in a BL environment in the future. However, it is also noted that more training and resources are still required to facilitate the students. On a very significant note, our data also indicates computationally talented students from NW Ontario remote schools who could qualify to pursue advanced STEM programs and have scope for possible STEM careers. Ultimately we made a significant contribution to the limited computational thinking teaching resources for the Ontario curriculum with a custom-designed Curriculum and Assessment tool accounting for the various factors behind NW Ontario remote schools.

Acknowledgments

The research is a collaborative community-based project by Lakehead University and the Armstrong Public School, funded by NSERC Promoscience. The program is organized as a part of the Indigenous Outreach Program by Nijjii Indigenous Mentorship at Lakehead University for the students in grades 7-8 of Armstrong Public School. We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), [funding reference number PROSC/556957-2020], Armstrong Public School, Nijjii Indigenous Mentorship at Lakehead University, for facilitating the execution of this project.

Chapter 4

Assessing the Impact of AI-Assisted Coding in Self-Directed Reinforcement Learning MOOC

All of this chapter is submitted to a peer-reviewed journal as:

- Kamak, LP., Mago V. (2022). Assessing the Impact of AI-Assisted Learning on Self-Directed Massive Open Online Courses.

To better understand the academic potential of accessible artificial intelligence, this chapter presents a study with graduate students taking a Massive Open Online course in reinforcement learning using artificial intelligence and non-artificial intelligence coding references. The study introduces a new survey instrument that investigates participant interaction using a combination of human-artificial intelligence interaction frameworks derived from human-computer interaction frameworks. The results of this study demonstrate that artificial intelligence-based tools can enhance students' overall programming experiences with the significant technology acceptance of ChatGPT, though with caution for potential information bias and measures to facilitate future coding assistant design.

Keywords: Artificial Intelligence, Coding assistance, Massive Open Online Courses (MOOCs), Self-directed learning, Information bias, Personalized learning, Reinforcement learning, Game AI, ChatGPT, Education

4.1 Introduction

The field of Human-Artificial Intelligence Interaction (HAI) gained significant research importance with the third wave of Artificial Intelligence (AI) [59]. Human-Computer Interaction (HCI) professionals are now collaborating with AI engineers to design useful and usable AI applications. Many research-intensive universities such as University of California and Stanford University have adopted it and established valuable HAI studies [60, 61]. The HAI framework emphasizes enhancing human capabilities rather than replacing them, requiring HCI design. One of the classic applications of the HAI approach is to design systems that enable human operators to effectively and quickly take control of the AI system in an emergency to avoid accidents. Therefore, HAI research design is crucial in ensuring that both humans and society benefit, making it a relevant field of study. Through this HAI study, we provide valuable insights for future research and development of AI-based education tools.

As AI technology evolves, they are increasingly integrated into various aspects of education. At the same time, Massive Open Online Courses (MOOCs) are becoming increasingly popular with college students who want to learn and improve their skills at their own pace [62]. MOOCs can provide personalized learning experiences catered to each student's individual needs with the aid of AI-assisted tools [63]. However, there is concern that information bias could influence the student's self-directed learning journey using unguided AI technology [64]. Information bias has been defined in many ways, here is one definition we have chosen: "Information bias is any systematic difference from the truth that arises in the collection, recall, recording and handling of information in a study, including how missing data is dealt with" [65]. The potential for AI to revolutionize education still requires investigation and understanding to ensure its effectiveness.

In this thesis, we examine the learning experience for university graduate students in self-directed reinforcement learning (RL) MOOCs using AI assistance for programming. To achieve this, our pilot study examines the potency of participants in producing precise and efficient code in both AI and non-AI contexts. This study compares the role of OpenAI GPT Models versus the Google search engine in supporting student learning processes in programming [66]. As the experimental group tool, ChatGPT allows users to answer questions and generate texts and code based on natural language processing. On the other hand, users

in the control group used the Google search engine to obtain information for coding-related queries. We uncover areas of AI-assisted code generation limitations that can be improved to enhance student educational experience. Overall, we evaluated technology acceptance as a coding assistance tool for graduate students. This study explores the following research questions in this context:

- RQ1: How do students perceive AI-assisted learning experience in terms of compatibility and trust?
- RQ2: Can using AI-assisted learning in MOOCs lead to information bias for students?
- RQ3: What is the extent of technology acceptance of ChatGPT as a coding assistance tool among beginner and advanced programmers?

This chapter follows the following structure: Section 2 discusses the various ChatGPT studies, and MOOC surveys carried out in similar educational domains. Section 3 demonstrates the methodology for the survey, with the HAI techniques used in the research instrument. Section 4 highlights the main findings of the analysis on coding performance and HAI metric evaluation, such as *compatibility*, *trust*, *perceived usefulness* and *ease of use*. Section 5 presents the study's conclusion with key findings for the above mentioned research questions.

4.2 Related Works

In light of the popularity of ChatGPT, many review studies aim to explore how educational research shapes current educational practices. A study conducted a large-scale survey to investigate the effect of AI assistant tools on code security in 2022 [67]. In their study, participants were given six questions covering security and programming languages and asked to solve them while being observed for interaction behaviour. They demonstrate that participants with access to the AI code generation model- codex-davinci-002 produced comparatively less secure code than participants who wrote secure code manually. Interestingly, participants with access to AI assistants perceived their code as secure despite having

more security problems. This study highlights the potential negative impact of AI-assisted coding on code security and the importance of caution when relying on AI assistants for programming tasks.

Similarly, numerous user studies employing ChatGPT have been conducted focusing on the coding perspective. A recent study evaluated ChatGPT for automated program repair on the QuixBugs benchmark set [68]. They compare ChatGPT's bug-fixing performance with several other approaches reported in the literature and discover that ChatGPT is remarkably better. The study also highlights ChatGPT's unique features, such as expected output or observed error messages, which can be entered to improve its success rate. The results show that by providing such hints to ChatGPT, its success rate can be further increased, fixing 31 out of 40 bugs and outperforming other approaches. Similarly, another study was performed in an educational setting assessing the potential impact of ChatGPT on traditional forms of instruction [69]. Using an established software testing curriculum, the authors evaluated ChatGPT's performance in solving queries. Their findings show that ChatGPT can provide partially correct or correct answers and explanations in 44% and 57% of cases, respectively. Furthermore, they observe that providing additional prompts results in a slightly higher rate of correct answers. Based on their findings, the authors discuss the potential promise and dangers of using ChatGPT by both students and instructors.

From the MOOC community, a study investigating the influence of learning styles on MOOCs was conducted in 2015 [70]. One hundred eighty-four undergraduate student's learning experiences and intentions to use MOOCs were analyzed. Their results show that high-reflective learners tend to have less experience using MOOCs. The researchers clustered multiple learning styles with the K-means algorithm and found a 90% probability that the type of learning style influences learner intentions. Finally, the authors analyzed learning approaches that could assist learning in MOOCs and examined the correlation between learning styles and user intention.

Recently, GPT-4, multimodal model that outperforms humans on various professional and academic benchmarks including passing the bar exam with top scores, was released by OpenAI (2023) [71]. GPT models have also been tested on a popular online coding learning platform - Leetcode [72]. Table 4.1 provides the GPT models programming exam scores tested at different proficiency levels.

Table 4.1: GPT coding exam scores for real-time exam simulations

Exam	GPT-4 (no vision)	GPT-3.5
Leetcode (easy)	31 / 41	12 / 41
Leetcode (medium)	21 / 80	8 / 80
Leetcode (hard)	3 / 45	0 / 45

Although the studies mentioned above offer insightful information about ChatGPT, they have yet to explore complex subjects such as reinforcement learning. Likewise, incorporating AI-assisted tools for MOOCs also requires a comprehensive study of student perceptions. This thesis provides a detailed HAI analysis to comprehend the growing trend of AI integration into self-directed MOOC learning on reinforcement learning and ultimately hopes to strengthen existing academic literature on this subject matter.

4.3 Methodology

This section outlines the design of our research instrument along with the frameworks and metrics involved. It also describes the coding tasks questionnaire, the research participants and the study procedure.

Research Instruments

A HAI theoretical framework is essential to ensure the validity and reliability of this study's social implications. The framework helps develop a systematic approach to survey design and identifies the right metrics. Our research instrument is a carefully curated survey questionnaire using a hybrid HAI framework. It gathers HAI metrics on ChatGPT and RL coding experience data using the Likert scale, multiple choice and open-ended questions.

HCI and HAI metrics

For HCI research to be effective, participant's perceptions must be measured [73]. *Task completion time, error rate, success rate, usability, learnability, recall, and attention* are

commonly used to quantify participant behaviour [74]. The user's subjective experience also determines *user satisfaction*, *ease of use*, and *interface efficiency* usually gathered through surveys and interviews [75]. Similarly, evaluating participant behaviour is critical in HAI research to determine the effectiveness of the AI system's interaction with humans [76, 77]. While user satisfaction and ease of use are standard metrics for HAI and HCI research, HAI also requires additional metrics including *trust*, *perception of fairness*, *transparency*, and *explanation quality* [78]. A *trust* scale can determine whether a user believes the AI system can complete the task correctly [79, 80, 81]. *Transparency* measures how well a participant comprehends the reasoning behind AI system decisions, whereas *perception of fairness* assesses how fair the system's decisions appear to the participant [82]. The quality of the explanation the AI system provides for its decisions can be measured as *explanation quality* [83]. Because of the involvement of an AI system, HAI evaluation metrics and methods are incredibly complex, and these metrics are vital for making informed decisions to improve user interaction.

HAI Evaluation Framework

HAI research requires a framework to evaluate AI technology and user acceptance. The survey research instrument is developed using a hybrid framework combining the Technology Acceptance Model (TAM), the Unified Theory of Acceptance and Use of Technology (UTAUT), the Innovation Diffusion Theory (IDT) and the Cognitive Fit theory. Figure 4.1 illustrates the hybrid HAI evaluation framework designed to understand student perceptions during the AI-Assisted Code Generation study [84]. TAM is a widely used framework to assess *perceived usefulness* and *ease of use* [85, 86]. It assesses attitudes towards technology, including positive or negative feedback. It also considers social influence, such as peer's influence. It evaluates subjective norms, such as stakeholder's expectations and beliefs. The UTAUT framework expands upon TAM and includes three additional factors: *effort expectancy*, *performance expectancy* and *facilitating conditions* [62, 87]. *Effort expectancy* is a measure of user belief that technology will be easy. *Performance expectancy* refers to an individual's confidence that technology will improve their performance. *Facilitating conditions* measure the user's perceived availability of resources and support for using the technology.

UTAUT is commonly used in HAI research to evaluate AI systems user acceptance and usage. The IDT framework focuses on innovation adoption and diffusion and includes four factors: *relative advantage*, *compatibility*, *complexity*, and *trialability* [88]. *Relative advantage* refers to the perception of superiority over existing technology. *Compatibility* measures perceived alignment with user needs and values. *Complexity* measures perceived difficulty of use, while *Trialability* refers to the technology’s availability for trial and testing before adoption. It is commonly used in both HCI and HAI research to evaluate technology adoption and diffusion. Cognitive Fit theory explores the fit between an individual’s cognitive style and technology’s cognitive demands [89, 90]. It is commonly used in HCI and HAI research to evaluate user acceptance and usage.

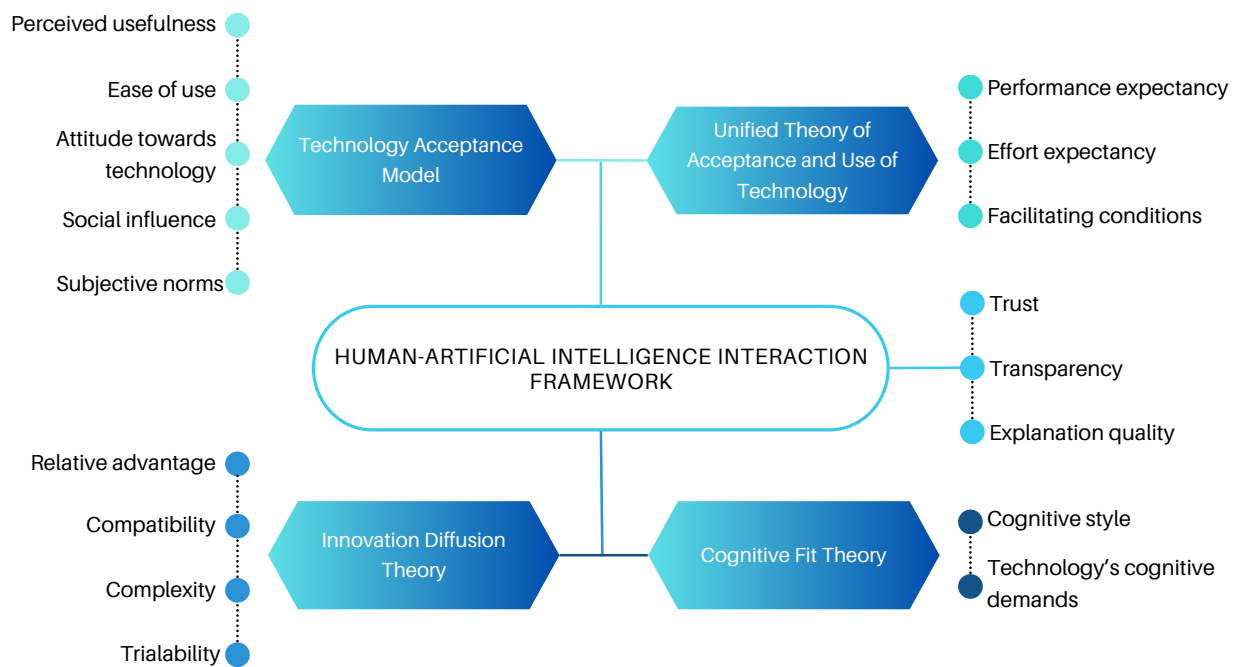


Figure 4.1: Human-Artificial Intelligence Interaction Evaluation Framework

Reinforcement Learning Evaluation

The most common definition for reinforcement learning is “learning what to do - how to map situations to actions - in order to maximize a numerical reward signal” [91]. Another survey states, “Reinforcement learning is the problem faced by an agent that must learn behavior

through trial-and-error interactions with a dynamic environment” [92]. The agent learns a set of rules that optimizes its rewards over time by using feedback from the environment as rewards or penalties. These topics have been covered in the Kaggle course “Intro to Game AI and Reinforcement Learning”, the chosen MOOC for this study [93]. It teaches how to create a game-playing agent by writing simple algorithms with one step look ahead, followed by N-step look ahead using the minimax algorithm, and finally, deep reinforcement learning using machine learning techniques. We evaluate the participant’s understanding of the Kaggle course and RL coding knowledge using five questions as part of the survey. They were evaluated in key areas of implementing Reinforcement Learning based on the programming language of their choice, setting reward function, Minimax Algorithm, bug Fixing in Q-values and Alpha-Beta pruning algorithm [92, 93, 94]. We used Python for coding tasks, as it is the coding language used in Kaggle course exercises. The assessment tests their ability to think critically while bug-fixing, problem-solving using algorithms and potentially identifying information bias in the process. All questions were provided in non-selectable image format to prevent participants from simply selecting and searching online for the results. The researchers have attempted all the RL tasks in both AI and non-AI contexts to see how the participants might approach the problem. Both the AI and non-AI context test groups undertook the same RL assessment. This ensured that the survey was conducted objectively and consistently. The complete task details, including related diagrams and code, are available in Appendix 6.3, 6.4.

Readability tests

Following the creation of the survey, we ensured that the ChatGPT interactions were appropriate for reinforcement learning guidance and checked the explanation quality with domain experts. To determine compatibility, we examined several ChatGPT prompt answers. We gathered text statistics and performed readability tests. The grade-level indicators assess text readability and correlate it to the school grade-level system. The following readability tests are used in this study: Flesch Kincaid Grade Level, Gunning Fog Score, Automated Readability Index, Coleman Liau Index, and Smog Index [95]. The majority of the tests measure complex words and syllable counts, whereas the Coleman Liau Index and Automated

Readability Index examine the amount of characters, words, and sentences.

Ethics Statement

As per Tri-Council Policy Statement (TCPS 2, 2022), Ethical Conduct for Research Involving Humans, Article 2.5 states that “Quality assurance and quality improvement studies, program evaluation activities, and performance reviews, or testing within normal educational requirements when used exclusively for assessment, management or improvement purposes, do not constitute research for the purposes of this Policy, and do not fall within the scope of REB review” [96, 97]. This research focuses on activities commonly used in the normal course of academic training for self-development, in which students solicit information anonymously. We followed the guidelines to ensure the research met all ethical requirements. Therefore, this study is exempt from ethical review as it falls under educational research conducted exclusively for professional development.

Procedure

We recruited computer science graduate students from Lakehead University, from year 1 to year 2, to have diverse students with different reinforcement learning experiences. Through class mailing lists, we invited participants by providing study information along with the opportunity to win a \$25 gift card for participation. We divided the 43 students into two groups at random: a control group that had to complete the coding task manually by referring to Google in a non-AI context, and an experimental group with access to ChatGPT for AI assistance [67]. Table 4.2 summarises our participant’s demographics. 62.79% of participants had no prior knowledge of reinforcement learning, while 37.21% had prior experience or formal training. We refer to them as beginners and advanced programmers throughout the rest of the thesis.

Table 4.2: Summary of Participant Demographics

Experience in Reinforcement Learning	Number of Participants
Beginner	27
Advanced - Prior Experience	16
Advanced - Formal training	6

In terms of *trialability*, the opportunity to test ChatGPT is available as a free research preview tool [66, 98]. This allows participants to create an account to explore the tool for research. They used GPT 3.5, with one participant having research access to GPT-4. We identified the participant information and assigned random participant IDs after collecting the data via the online Google survey form. Descriptive statistics techniques were applied to categorical data, the Welch test was employed for statistical significance, and the data analysis assumed unequal mean and variance [99]. It is imperative to note that this data only represents a snapshot of graduate students willing to use MOOCs to upskill their technical expertise.

4.4 Results and Discussion

In this section, we examine the outcomes of the various HAI metrics based on the survey data and their statistical results.

RQ1: How do students perceive AI-assisted learning experience in terms of compatibility and trust?

General attitude towards coding assistance tools

In our study, we first gather participant’s general viewpoints regarding coding assistance tools used in their learning process nowadays. In terms of *compatibility*, preference can vary depending on individual cognitive style and needs. According to Cognitive fit theory, effectiveness of technology depends on the extent to which it matches the user’s cognitive capabilities and the task at hand [89]. Figure 4.2 shows that Google search engine is used

by programmers of both skill levels at the same rate of 23%. However, ChatGPT is used more often by beginner programmers (39.53%) than advanced programmers (13.95%). This suggests that ChatGPT might be a better option for new programmers seeking guidance on coding-related questions. However, advanced programmers might prefer to rely on their own knowledge and specialized tools.

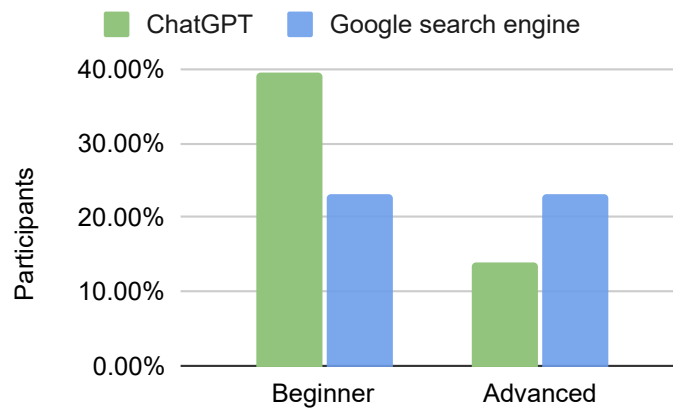


Figure 4.2: General preferences of graduate students based on their coding experience

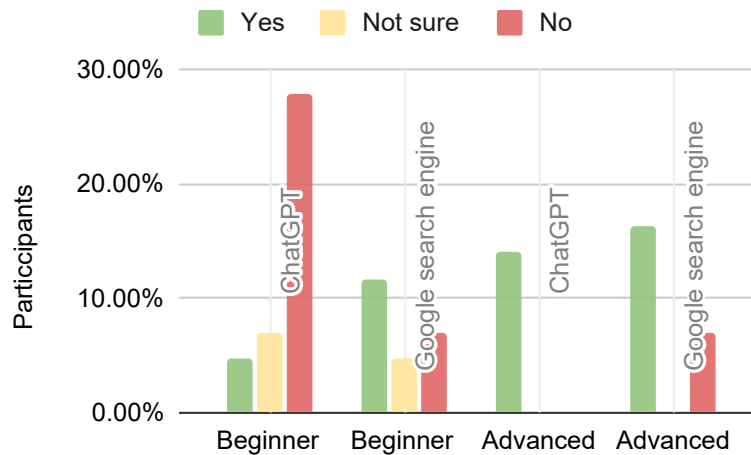


Figure 4.3: General attitude towards potential bias in coding assistance tools

We also surveyed if they had previously encountered wrong information in Google search results or potential bias in ChatGPT to collect insights into the level of *trust*. Figure 4.3

Table 4.3: Reading ease of the ChatGPT prompts on RL explanation

Readability tests	Grade level
Flesch Kincaid Grade Level	8.54
Gunning Fog Score	9.92
Smog Index	10.22
Automated Readability Index	10.04
Coleman Liau Index	7.55
Average Score	9.254

reveals that 41.9% of the participants did not encounter potential bias or incorrect information, with most of them being beginners (27.91%). Of the 46.57% of participants who reported running into inaccurate information in Google search results or noticing a possible bias in ChatGPT results, 30.2% were advanced programmers. This suggests that advanced programmers are evaluating the outcomes critically and are capable of identifying incorrect information because of their experience [100].

Readability test

To test another aspect of *compatibility* and *explanation quality*, we performed readability tests to measure average reading scores for the ChatGPT answer explanations, also shown in Table 4.3. The average readability was 9.254 based on the multiple tests. This indicates that the prompts are at a comfortable level for a graduate student to understand a complex subject.

RQ2: Can using AI-assisted learning in MOOCs lead to information bias for students?

This section investigates the performance and feedback of participants from both the AI and non-AI context groups on the reinforcement learning coding tasks described in Section 4.3. We examined the results for the correctness and the types of algorithms they employed and how they explored the tasks. The transcript of one of the participants interacting with ChatGPT during the RL coding task is available for reference in Appendix Section 6.5.

Table 4.4: The variety of Reinforcement Learning approaches implemented by participants for Question 1

AI context Participants
N-Step SARSA algorithm
Q-learning algorithm with a Q-value table with greedy policy
Q-learning algorithm with Q-value estimator with epsilon-greedy policy
Q-learning algorithm with N-step look ahead using neural networks
Non-AI context Participants
Minimax algorithm
One step look ahead algorithm
Q-learning algorithm with the epsilon-greedy policy
Pseudocodes of the Q-learning algorithm
Pseudocodes of Minimax algorithm

Question 1: The participants completed the RL task by employing a variety of algorithms, as shown in Table 4.4. Participants in the AI context group used algorithms such as N-Step SARSA and Q-learning with variations in handling Q-values and exploring the environment using greedy, epsilon-greedy, and N-step look-ahead using neural networks. As for the non-AI context participants, they implemented the RL task using algorithms such as one-step look ahead, Minimax, and Q-learning with an epsilon-greedy policy. Also, some participants provided pseudocodes of these algorithms. This shows that the students incorporated a variety of algorithms, and their *creativity* was not at risk. It is observed that Google users, who felt they had limited time, tended to rely on the Minimax algorithm readily available in tutorials and blogs. On the other hand, ChatGPT users had easy access to more complex algorithms. However, the results of this question were not available for all 42 participants since the task was optional due to time constraints.

Question 2 to 5: As observed in Figure 4.4, there is a significant difference in performance between the non-AI and AI context groups for each task. The success rate of the AI context is comparatively lower for all the tasks. This suggests additional challenges and complexities when performing these tasks with AI. The “Difficulty index” represents the measure of assigned difficulty of each task. The detailed numerical representation of the difficulty index assigned to each task and the respective student performance is provided in

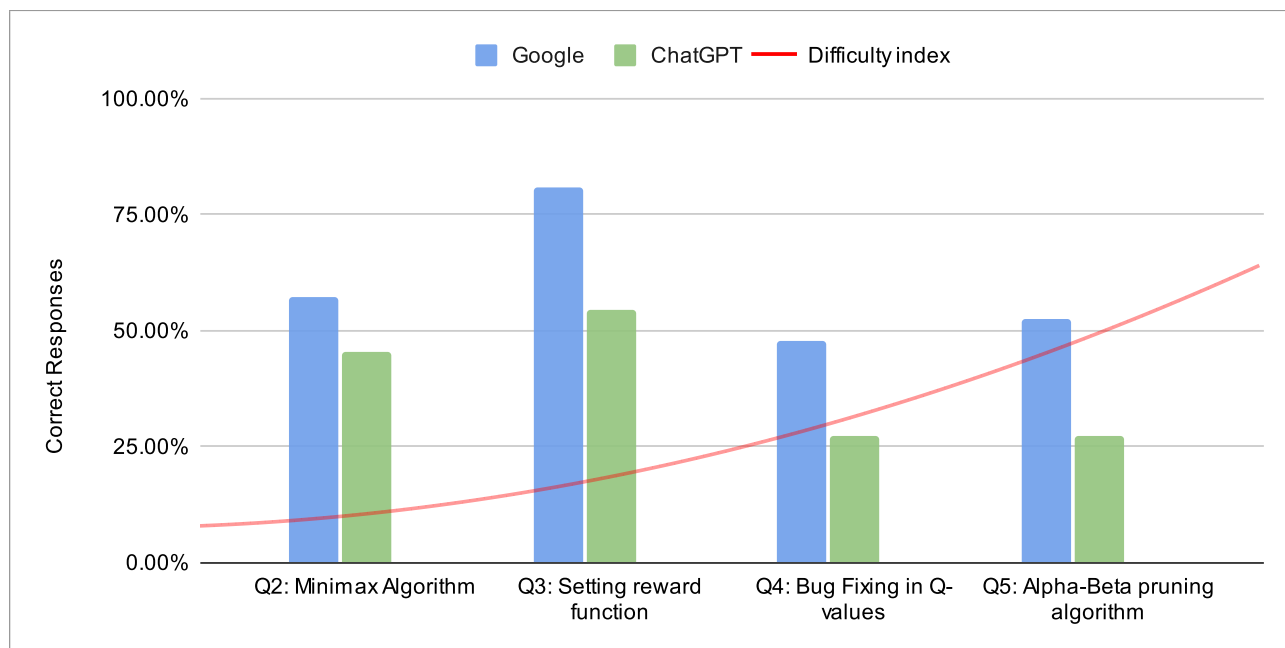


Figure 4.4: Reinforcement learning coding test performance

Coding assistant tool	RL Coding tasks	Correct responses	Difficulty index
Google	Q2: Minimax algorithm	57.14%	1
ChatGPT	Q2: Minimax algorithm	45.45%	1
Google	Q3: Setting reward function	80.95%	2
ChatGPT	Q3: Setting reward function	54.55%	2
Google	Q4: Bug fixing in Q-values	47.62%	3
ChatGPT	Q4: Bug fixing in Q-values	27.27%	3
Google	Q5: Alpha-Beta pruning	52.38%	5
ChatGPT	Q5: Alpha-Beta pruning	27.27%	5

Figure 4.5: Explicit numerical information on the difficulty index and tasks supporting Figure 4.4

Figure 4.6 compares AI and non-AI context groups regarding overall test performance, *perceived difficulty*, and *confidence*. In terms of *perceived difficulty*, the data shows that for

the effort for MOOC category, ChatGPT is rated as more difficult than Google. However, for the effort in RL test category, Google is rated more difficult than ChatGPT. This implies that in terms of *effort expectancy*, ChatGPT is easier to use than Google for RL tasks. With the p-value of 0.048 for *effort expectancy*, the result is statistically significant. Additionally, Google and ChatGPT participants rated relatively high confidence in RL programming. Furthermore, in the confidence RL explanation category, ChatGPT participants have slightly higher confidence with a p-value of 0.032. In the confidence in the coding assistance tool category, ChatGPT users have relatively higher confidence in their coding assistance tool, with a p-value of 0.014. Finally, the majority of Google participants (23.26%) performed “Very Good” in the RL test, while the most common performance in ChatGPT was “Fair” (20.93%) and “Poor” (20.93%). Interestingly, 2.33% of ChatGPT participants scored “Very Good”, indicating positive model interaction for the users. The test performance results are statistically significant, with a p-value of $2.7586e-6$. Even though the majority of ChatGPT users performed fairly, the positive ChatGPT interactions indicate that users could perform well using the tool. This implies that ChatGPT as a coding assistance tool has room for improvement in terms of *user experience*. In summary, although many users have much higher confidence in ChatGPT, they mostly perform poorly in the RL test. This is supported by strong statistically significant results of $p < 0.05$ and $p < 0.001$ using Welch’s test in the test performance and confidence in their coding assistance tool [67, 99]. This also supports that using AI-assisted learning in MOOCs may lead to potential information bias among self-directed learners as they may rely too heavily on the incomplete recommendations and feedback the AI system provides.

RQ3: What is the extent of Technology Acceptance of ChatGPT as a coding assistance tool among beginner and advanced programmers?

This section presents the result for the Technology Acceptance Model metrics - *Perceived Usefulness*, *Ease of use* for ChatGPT as a coding assistance tool among beginner and advanced programmers.

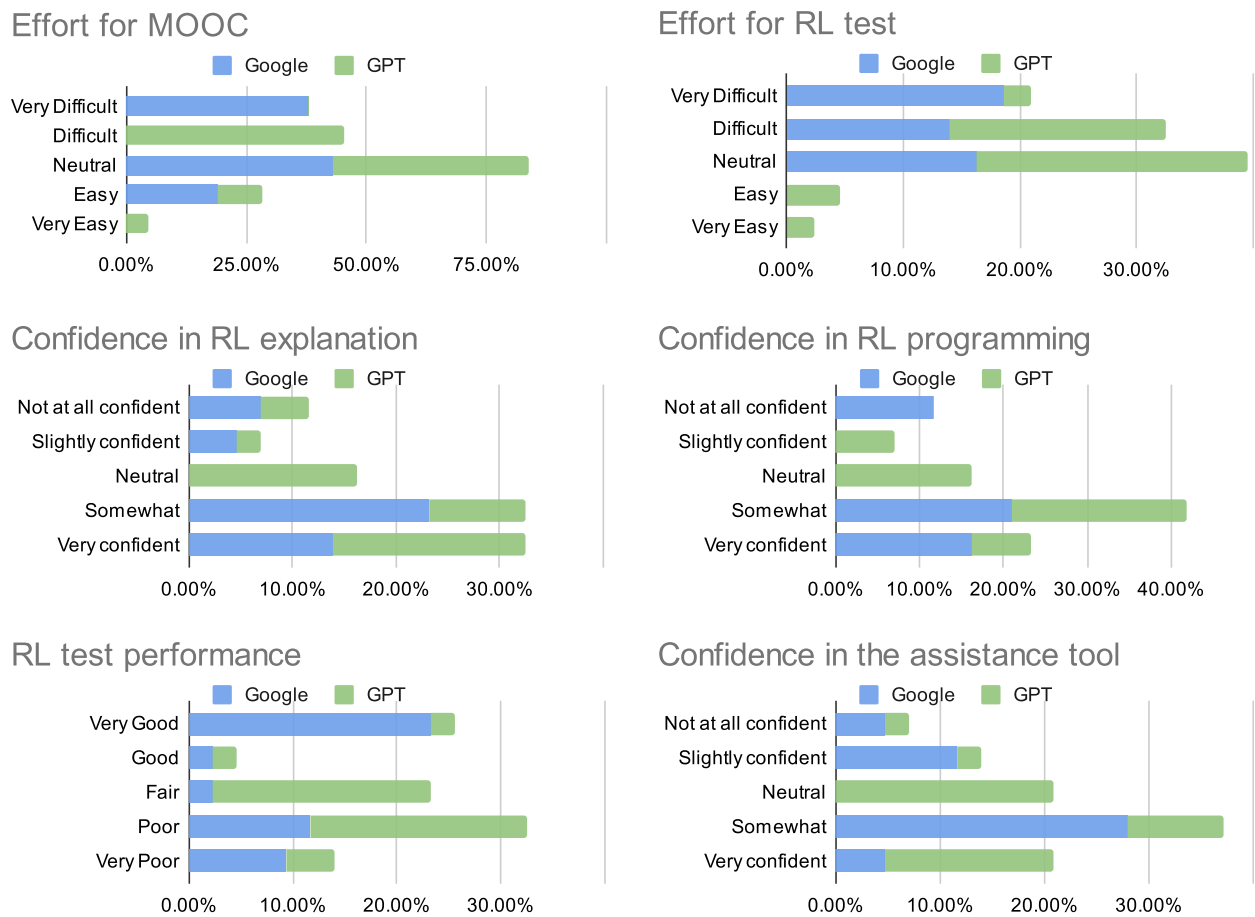


Figure 4.6: Distribution of the test performance, perceived difficulty and confidence of AI and non-AI context groups

Perceived Usefulness

This section discusses the user’s perceived usefulness in ChatGPT as a coding assistant tool. Figure 4.7 illustrates the perceived impact on critical thinking on using ChatGPT pre-test. At the beginning of the study, most participants (55.8%) felt that ChatGPT negatively impacts critical thinking. Interestingly, most students (37.2%) who felt that ChatGPT positively impacts critical thinking were beginners. Figure 4.8 illustrates the significant difference in attitude among advanced programming students after the test. 86.05% of all the participants from both groups felt that their respective coding assistance tool impacted their coding self-journey positively, out of which 81.8% of the ChatGPT participants also

felt it was a positive impact. This also shows the relative advantage and compatibility for the ChatGPT participants. In terms of *complexity*, 79.41% of the Google group felt that AI assistance could have made their self-learning journey easier. Likewise, regarding *performance expectancy*, 58.82% of the ChatGPT group felt they noticed improvements in their learning experience since incorporating AI-assisted learning. Finally, the overall motivation for participants to use their respective coding assistance tool is displayed in Figure 4.9. It is observed that saving time and increasing efficiency, and improving the accuracy of data processing results are the top priorities for all participants. Keeping up with the latest technological advancements is placed at higher importance by Google participants. Interestingly, reducing the need for manual data processing is less critical in ChatGPT and Google. This could be because students prefer to code manually to custom-fit their data to the problem [101].

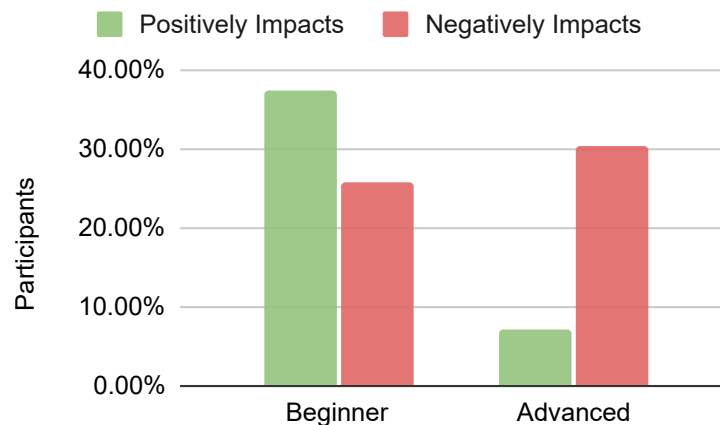


Figure 4.7: Perceived impact on critical thinking on using ChatGPT pre-test

Ease of Use

This section analyzes the participant's comments on their challenges in their MOOC and coding journey. A few noteworthy observations are provided below:

- Some participants highlighted that they preferred to spend less time looking for information during their learning process.

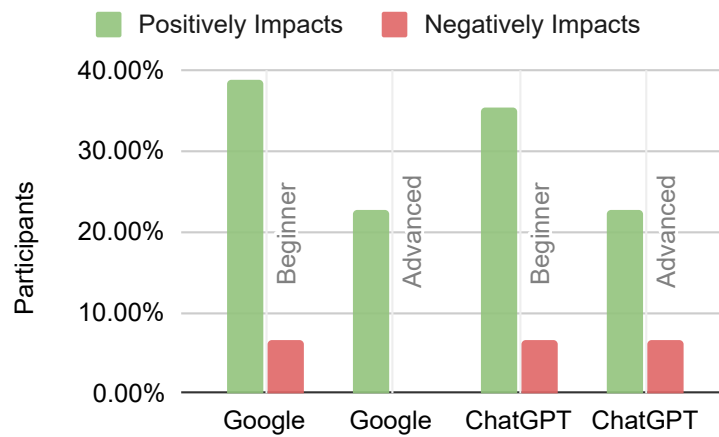


Figure 4.8: Perceived impact on their self-direction with the respective coding assistance tool post-test

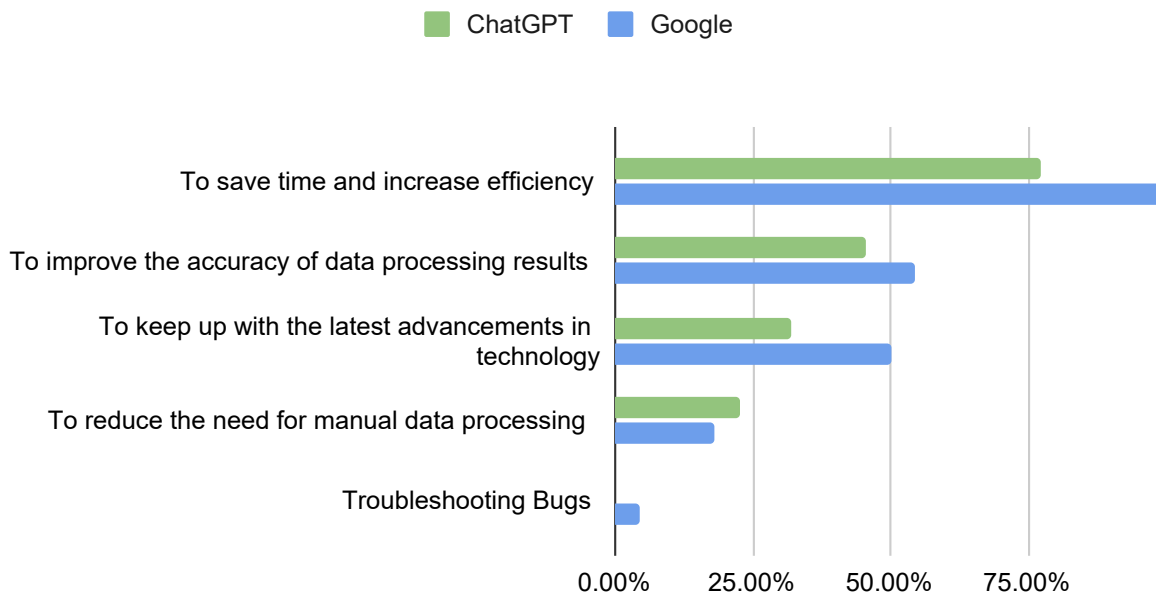


Figure 4.9: Perceived usefulness of the coding assistance tools

- One participant noticed that the ChatGPT code responses were mostly incomplete, and the participant was required to request ChatGPT to complete the code on those instances.

- A participant said they prefer ChatGPT to sometimes address cold start¹ challenges and receive boilerplate code².

Regarding *transparency*, we highlight participant concerns about ChatGPT often being unable to cite accurate sources.

Overall concerning technology acceptance, both groups have expressed a strong interest in using ChatGPT as a coding support tool due to their *perceived usefulness*. In terms of *usability*, there are situations when beginners can only rely on ChatGPT to a limited extent. However, they can still receive its assistance during their learning process. Advanced programmers prefer ChatGPT because it saves time over coding manually and using specialised tools to troubleshoot faults.

4.5 Conclusion

In this chapter, we provide a comprehensive analysis of the human artificial-intelligence interaction components to explore the potential of OpenAI ChatGPT tool in assisting university students in the learning process of coding complex problems. We also compare its effectiveness with manual coding, referring to Google search results. Using a combination of human artificial-intelligence interaction frameworks, we designed a new survey instrument to collect human artificial-intelligence interaction metrics while evaluating their reinforcement learning knowledge. In addressing RQ1, we investigated human artificial-intelligence interaction metrics - *compatibility* and *trust* in artificial intelligence. In terms of *compatibility*, we observe that ChatGPT is highly preferred by beginners seeking guidance on coding-related questions. In comparison, advanced programmers prefer to rely on their knowledge and tools. Additionally, the ChatGPT prompt's readability is at an introductory level that allows a graduate student to comprehend the complicated subject easily. In terms of the *trust*, beginners are comparatively unaware of incorrect information in their coding jour-

¹The “cold-start problem” refers to a programmer’s challenge when starting a new project with little or no prior experience in the relevant domain. In a data-driven approach, programmers often face insufficient data to generate pseudocode initially [102]

²“Boilerplate code” refers to generic code that is a basic setup before implementing complex functionality [103]. It saves time and effort for a programmer by eliminating the need to rewrite the same code for similar projects.

ney, while advanced programmers are more aware and critical. Correspondingly, for RQ2, our study confirms a potential bias in ChatGPT participants who report high confidence while performing fairly in the reinforcement learning evaluation tasks. We suggest *facilitating conditions* in future studies to evaluate whether participants can access the necessary resources and support to address any biases they perceive. In response to RQ3, we evaluated the technology acceptance of ChatGPT as a coding assistance tool among beginner and advanced programmers. ChatGPT is widely accepted with high *perceived usefulness*, *ease of use* and few concerns about *reliability*. In conclusion, this study offers insights into integrating artificial intelligence technology in academia while assessing its effectiveness.

Acknowledgments

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), [funding reference number PROSC/556957-2020] for facilitating the execution of this project.

Chapter 5

Conclusion

This thesis contributes to the limited resources available for computational thinking education by providing curriculum with scaffolding support for middle school students learning Python. Additionally, it also demonstrates the potential of artificial intelligence-assisted coding for more experienced students. All in all, this thesis proposes innovative and effective programming pedagogy strategies catering to students' diverse learning needs and presenting research findings to enhance programming education that can be adopted in remote Northwestern First Nation communities.

Chapter 2 of this thesis determined the convergence of computational thinking and artificial intelligence in education as an emerging research interest from 2017 using performance analysis and science mapping techniques employed on bibliometric data. Through measures like top-cited publications, multidisciplinary impact analysis, geo-contribution, and journal leadership, the most crucial computational thinking authors contributing to the future direction of this field's research are identified, allowing for a thorough overview of the field's development. Thematic area identification utilizing co-word analysis allows us to identify the impact of studies popularly within Education Educational Research initiatives at various levels of education.

In Chapter 3, our first study using human-computer interaction design contributes to limited computational thinking teaching resources with a custom-designed Curriculum and Assessment tool specifically for remote schools in NorthWestern Ontario that will be taught within a blended learning setting and successfully adopted by other schools as well. The

novel design includes a scaffolding intervention technique integrating Python with block-based visual aids to effectively combine block and text-based programming. Additionally, our data also indicate computationally talented kindergarten through grade 12 students who have the potential to pursue science, technology, engineering, and mathematics careers.

Then, shifting to a newer aspect of human-computer interaction research known as human-artificial intelligence interaction, Chapter 4 demonstrated the potential of artificial intelligence in the self-directed learning approach most prevalent in graduate students. Regarding technology acceptance, OpenAI's ChatGPT is widely accepted among beginner and advanced programmers as a coding assistance tool, with metrics addressing perceived usefulness, usability, compatibility, trust, and reliability. This research demonstrates the possibility of an information bias among ChatGPT participants and hence emphasizes the value of creating the facilitating conditions for students to access the resources to verify any biases they may notice.

Computational thinking and artificial intelligence-assisted coding are promising areas for future exploration based on the evidence presented in this dissertation. Future work could focus on incorporating the scaffolding prototype design from Chapter 3 into learning technology. This involves building a custom graphical user interface integrating the Python console with visual aids to highlight control flow. Through this tool, novice young beginners could transition effortlessly from block-based to text-based programming, enabling them to overcome many debugging problems. As a result, students can eventually move on and work with more traditional Python consoles or advanced integrated development environments. Another potential direction for future research could be developing and evaluating novel pedagogical strategies incorporating robotics and artificial intelligence-assisted coding. Robotics based learning methodologies is among the top thematic areas, and its technological advancements are highlighted in our bibliometric analysis. It could involve designing revised curricula and assessments tailored to different learning environments.

This thesis emphasizes the promising foundation for future computational thinking and artificial intelligence-assisted coding applications. Ultimately, the thesis also demonstrates the importance of leveraging technology and design principles from human-artificial intelligence interaction and human-computer interaction to optimize learning outcomes from kindergarten through grade 12 and university programming education.

Bibliography

- [1] Juan David Rodriguez Garcia et al. “LearningML: a tool to foster computational thinking skills through practical artificial intelligence projects”. In: *Revista de Educación a Distancia (RED)* 20.63 (2020).
- [2] “What is computational thinking? - introduction to computational thinking - KS3 computer science revision - BBC bitesize”. en. In: *BBC News* (). Online Available. URL: <https://www.bbc.co.uk/bitesize/guides/zp92mp3/revision/1..>
- [3] Sdenka Zobeida Salas-Pilco. “The impact of AI and robotics on physical, social-emotional and intellectual learning outcomes: An integrated analytical framework”. In: *British Journal of Educational Technology* 51.5 (2020), pp. 1808–1825.
- [4] Clare Baek and Tenzin Doleck. “A Bibliometric Analysis of the Papers Published in the Journal of Artificial Intelligence in Education from 2015-2019”. In: (2020).
- [5] Binar Kurnia Prahani et al. “Artificial Intelligence in Education Research During the Last Ten Years: A Review and Bibliometric Study.” In: *International Journal of Emerging Technologies in Learning* 17.8 (2022).
- [6] Dimitrios Kalles, Kyriakos Giagtzoglou, and Konstantinos Mitropoulos. “Gaming Ecosystems for Education and Research: Where Artificial Intelligence meets with Software Engineering, at scale”. In: *Proceedings of the 11th International Conference on Management of Digital EcoSystems*. 2019, pp. 199–204.
- [7] Andreas F Gkontzis et al. “Polarity, emotions and online activity of students and tutors as features in predicting grades”. In: *Intelligent Decision Technologies* 14.3 (2020), pp. 409–436.

- [8] Dimitrios Kalles and Eirini Ntoutsis. “Interactive verification of game design and playing strategies”. In: *14th IEEE International Conference on Tools with Artificial Intelligence, 2002. (ICTAI 2002). Proceedings*. IEEE. 2002, pp. 425–430.
- [9] Mehmet Tekdal. “Trends and development in research on computational thinking”. In: *Education and Information Technologies* 26.5 (2021), pp. 6499–6529.
- [10] Hui E Chen et al. “Visualising trends in computational thinking research from 2012 to 2021: A bibliometric analysis”. In: *Thinking Skills and Creativity* 47 (2023), p. 101224.
- [11] Caroline Birkle et al. “Web of Science as a data source for research on scientific and scholarly activity”. en. In: *Quantitative Science Studies* 1.1 (2020), pp. 363–376.
- [12] Jane Waite and Sue Sentance. “Teaching programming in schools: A review of approaches and strategies”. In: (2021).
- [13] D. Weintrop. “Block-based programming in computer science education”. fr. In: *Communications of the ACM* 62.8 (2019), pp. 22–25.
- [14] John Maloney et al. “The scratch programming language and environment”. In: *ACM Transactions on Computing Education (TOCE)* 10.4 (2010), pp. 1–15.
- [15] Nguyen-Think Le et al. “A review of AI-supported tutoring approaches for learning programming”. In: *Advanced computational methods for knowledge engineering* (2013), pp. 267–279.
- [16] Carsten Schulte. “Block Model: an educational model of program comprehension as a tool for a scholarly approach to teaching”. In: *Proceedings of the fourth international workshop on computing education research*. 2008, pp. 149–160.
- [17] Sue Sentance, Jane Waite, and Maria Kallia. “Teaching computer programming with PRIMM: a sociocultural perspective”. In: *Computer Science Education* 29.2-3 (2019), pp. 136–176.
- [18] Naveen Donthu et al. “How to conduct a bibliometric analysis: An overview and guidelines”. In: *Journal of Business Research* 133 (2021), pp. 285–296.
- [19] Nees Jan van Eck and Ludo Waltman. “VOSviewer manual”. en. In: *Manual for VOSviewer version 1.0* (2011).

- [20] Raminta Pranckutė. “Web of Science (WoS) and Scopus: The titans of bibliographic information in today’s academic world”. In: *Publications* 9.1 (2021), p. 12.
- [21] Julian Estevez, Gorke Garate, and Manuel Graña. “Gentle introduction to artificial intelligence for high-school students using scratch”. In: *IEEE access* 7 (2019), pp. 179027–179036.
- [22] Ying-Hsun Lai et al. “Study on enhancing AIoT computational thinking skills by plot image-based VR”. In: *Interactive Learning Environments* 29.3 (2021), pp. 482–495.
- [23] A. Eguchi. “RoboCupJunior for promoting STEM education, 21st century skills, and technological advancement through robotics competition”. en. In: *Robotics and Autonomous Systems* 75 (2016), pp. 692–699.
- [24] Matti Tedre et al. “Teaching machine learning in K–12 classroom: Pedagogical and technological trajectories for artificial intelligence education”. In: *IEEE Access* 9 (2021), pp. 110558–110572.
- [25] Michael Gr Voskoglou and Abdel-Badeeh M Salem. “Benefits and Limitations of the Artificial with Respect to the Traditional Learning of Mathematics”. In: *Mathematics* 8.4 (2020), p. 611.
- [26] Randi Williams et al. “Popbots: Designing an artificial intelligence curriculum for early childhood education”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 01. 2019, pp. 9729–9736.
- [27] Ma Paz Prendes Espinosa and Fernando Cerdan Cartagena. “Advanced technologies to face the challenge of educational innovation”. In: *RIED-Revista Iberoamericana de Educacion a Distancia* 24.1 (2021), pp. 35–53.
- [28] Wookhee Min et al. “DeepStealth: Leveraging deep learning models for stealth assessment in game-based learning environments”. In: *Artificial Intelligence in Education: 17th International Conference, AIED 2015, Madrid, Spain, June 22-26, 2015. Proceedings 17*. Springer. 2015, pp. 277–286.
- [29] Tapani Toivonen et al. “Co-designing machine learning apps in K–12 with primary school children”. In: *2020 IEEE 20th International Conference on Advanced Learning Technologies (ICALT)*. IEEE. 2020, pp. 308–310.

- [30] Salomi Evripidou et al. “Educational robotics: Platforms, competitions and expected learning outcomes”. In: *IEEE access* 8 (2020), pp. 219534–219562.
- [31] Nina Bonderup Dohn et al. “Survey: Artificial Intelligence, Computational Thinking and Learning”. In: *KI-Künstliche Intelligenz* 36.1 (2022), pp. 5–16.
- [32] Ting-Chia Hsu, Hal Abelson, and Jessica Van Brummelen. “The effects on secondary school students of applying experiential learning to the Conversational AI Learning Curriculum”. In: *International Review of Research in Open and Distributed Learning* 23.1 (2022), pp. 82–103.
- [33] Siu Cheung Kong and Wai Ying Kwok. “From mathematical thinking to computational thinking: Use scratch programming to teach concepts of prime and composite numbers”. In: *Proceedings of 29th International Conference on Computers in Education Conference. Asia-Pacific Society for Computers in Education*. 2022, pp. 549–558.
- [34] Ningyu Zhang et al. “Studying the interactions between science, engineering, and computational thinking in a learning-by-modeling environment”. In: *Artificial Intelligence in Education: 21st International Conference, AIED 2020, Ifrane, Morocco, July 6–10, 2020, Proceedings, Part I 21*. Springer. 2020, pp. 598–609.
- [35] Juan David Rodriguez Garcia et al. “Developing computational thinking at school with machine learning: an exploration”. In: *2019 international symposium on computers in education (SIIE)*. IEEE. 2019, pp. 1–6.
- [36] Nicole Hutchins et al. “A systematic approach for analyzing students’ computational modeling processes in C2STEM”. In: *Artificial Intelligence in Education: 20th International Conference, AIED 2019, Chicago, IL, USA, June 25-29, 2019, Proceedings, Part II 20*. Springer. 2019, pp. 116–121.
- [37] Ontario Ministry of Education. *Ontario Math Curriculum Expectations, Grades 1-8*. en. Retrieved January 9, 2023. 2020. URL: <https://www.dcp.edu.gov.on.ca/en/curriculum/elementary-mathematics>.

- [38] G. Gadanidis et al. “Coding in the Ontario Mathematics Curriculum, 1-8: Might it be transformational?” In: *Journal of Computers in Mathematics and Science Teaching* 40.4 (2021), pp. 357–373.
- [39] Andrea A. DiSessa. “Computational literacy and “the big picture” concerning computers in mathematics education”. In: *Mathematical Thinking and Learning* 20.1 (2018), pp. 3–31.
- [40] Philippe J Giabbanelli and Vijay K Mago. “Teaching computational modeling in the data science era”. In: *Procedia Computer Science* 80 (2016), pp. 1968–1977.
- [41] Ontario Ministry Education. “The Ontario curriculum, grades 1-8: Mathematics”. en. In: *Queen’s Printer for Ontario* (2020).
- [42] Maya Israel et al. “Understanding barriers to school-wide computational thinking integration at the elementary grades: Lessons from three schools”. In: *Computational thinking in PreK-5: Empirical evidence for integration and future directions*. 2022, pp. 64–71.
- [43] Xiaodan Tang et al. “Assessing computational thinking: A systematic review of empirical studies”. In: *Computers & Education* 148 (2020), p. 103798.
- [44] Isabella Corradini, Michael Lodi, and Enrico Nardelli. “Conceptions and misconceptions about computational thinking among Italian Primary School Teachers”. In: *Proceedings of the 2017 ACM Conference on International Computing Education Research* (2017). DOI: 10.1145/3105726.3106194.
- [45] I. Jong and J. Jeuring. “Computational thinking interventions in higher education: A scoping literature review of interventions used to teach computational thinking”. en. In: *Koli Calling’20: Proceedings of the 20th Koli Calling International Conference on Computing Education Research*. Nov. 2020, pp. 1–10.
- [46] Nagalaxmy Markandan, Kamisah Osman, and Lilia Halim. “Integrating computational thinking and empowering metacognitive awareness in STEM education”. In: *Frontiers in Psychology* 13 (2022). DOI: 10.3389/fpsyg.2022.872593.

- [47] *Connecting to Success: Technology in Ontario schools (2019)* People for Education. en. URL: <https://peopleforeducation.ca/report/connecting-to-success-technology-in-ontario-schools/>.
- [48] Marcos Román-González et al. “Can computational talent be detected? predictive validity of the computational thinking test”. In: *International Journal of Child-Computer Interaction* 18 (2018), pp. 47–58. DOI: 10.1016/j.ijcci.2018.06.004.
- [49] Satabdi Basu et al. “Identifying middle school students’ challenges in computational thinking-based Science Learning”. In: *Research and Practice in Technology Enhanced Learning* 11.1 (2016). DOI: 10.1186/s41039-016-0036-2.
- [50] Maria Cutumisu et al. “Callysto Computational Thinking Test (CCTt) Teacher Version [Measurement instrument]”. In: (2021). URL: <https://callysto.ca>.
- [51] Joanna Poon. “Blended learning: An institutional approach for enhancing students’ learning experiences”. In: *Journal of online learning and teaching* 9.2 (2013), p. 271.
- [52] J Cooper. *Incorporating Indigenous Cultures and Realities in STEM*. en. Ottawa: The Conference Board of Canada, 2020.
- [53] Interagency Advisory Panel on Research Ethics Government of Canada. *Consent*. Aug. 2018. URL: https://ethics.gc.ca/eng/policy-politique_interpretations_consent-consentement.html.
- [54] Readability WebFX. en. Retrieved November 30, 2022, from. Sept. 6, 2022. URL: <https://www.webfx.com/tools/read-able/>.
- [55] Diana Pérez-Marín et al. “Can computational thinking be improved by using a methodology based on metaphors and scratch to teach computer programming to children?” In: *Computers in Human Behavior* 105 (2020), p. 105849.
- [56] J. Robert. “Students and Technology Report: Rebalancing the Student Experience”. en. In: *Research report* (2022).
- [57] A.T. Ottenbreit. *Effects of Computer Troubleshooting on Elementary Students’ Problem Solving Skills*. fr. 2003.

- [58] H. Akoglu. *User's Guide to Correlation Coefficients*. *Turkish journal of emergency medicine*. fr. Aug. 7, 2018.
- [59] Wei Xu. "Toward Human-Centered AI: A Perspective from Human-Computer Interaction". In: *Interactions* 26.4 (June 2019), pp. 42–46. ISSN: 1072-5520. DOI: 10.1145/3328485. URL: <https://doi.org/10.1145/3328485>.
- [60] *Center for human-compatible Artificial Intelligence – Center for Human-compatible AI is building exceptional AI for humanity*. URL: <https://humancompatible.ai/>.
- [61] Nikki Goth Itoi, Shana Lynch, and Edmund L. Andrews. *Guiding Human-Centered AI*. Jan. 2023. URL: <https://hai.stanford.edu/research>.
- [62] Sodiq Onaolapo and Olawale Oyewole. "Performance expectancy, effort expectancy, and facilitating conditions as factors influencing smart phones use for mobile learning by postgraduate students of the University of Ibadan, Nigeria". In: *Interdisciplinary Journal of e-Skills and Lifelong Learning* 14.1 (2018), pp. 95–115.
- [63] Zhonggen Yu, Wei Xu, and Liheng Yu. "Constructing an online sustainable educational model in COVID-19 pandemic environments". In: *Sustainability* 14.6 (2022), p. 3598.
- [64] Ali S Tejani et al. "Artificial Intelligence and Radiology Education". In: *Radiology: Artificial Intelligence* 5.1 (2022), e220084.
- [65] Clare R Bankhead, Elizabeth A Spencer, and David Nunan. *Information bias*. <https://catalogofbias.org/biases/information-bias/>. Accessed: April 27, 2023. 2019.
- [66] OpenAI. *OpenAI ChatGPT API*. Online. URL: <https://platform.openai.com/docs/models/overview>.
- [67] Neil Perry et al. "Do Users Write More Insecure Code with AI Assistants?" In: *arXiv preprint arXiv:2211.03622* (2022).
- [68] Dominik Sobania et al. "An analysis of the automatic bug fixing performance of chatgpt". In: *arXiv preprint arXiv:2301.08653* (2023).
- [69] Sajed Jalil et al. "Chatgpt and software testing education: Promises & perils". In: *arXiv preprint arXiv:2302.03287* (2023).

- [70] Ray I Chang, Yu Hsin Hung, and Chun Fu Lin. “Survey of learning experiences and influence of learning style preferences on user intentions regarding MOOC s”. In: *British Journal of Educational Technology* 46.3 (2015), pp. 528–541.
- [71] OpenAI. *GPT-4 Technical Report*. <https://openai.com/research/gpt-4>. Accessed: March 25, 2023.
- [72] LeetCode. *The world’s leading online programming learning platform*. <https://leetcode.com/>. Accessed: March 25, 2023. no date.
- [73] Chin-Chao Lin. “Exploring the relationship between technology acceptance model and usability test”. In: *Information Technology and Management* 14.3 (2013), pp. 243–255. DOI: 10.1007/s10799-013-0162-0.
- [74] Jiahui Wang et al. “Exploring relationships between eye tracking and traditional usability testing data”. In: *International Journal of Human-Computer Interaction* 35.6 (2019), pp. 483–494.
- [75] Ugur Kuter and Cemal Yilmaz. “Survey methods: Questionnaires and interviews”. In: *Choosing Human-Computer Interaction (HCI) Appropriate Research Methods* (2001).
- [76] Saleema Amershi et al. “Guidelines for human-AI interaction”. In: *Proceedings of the 2019 chi conference on human factors in computing systems*. 2019, pp. 1–13.
- [77] Babak Abedin et al. “Designing and managing human-AI interactions”. In: *Information Systems Frontiers* 24.3 (2022), pp. 691–697.
- [78] Andrew Silva et al. “Explainable artificial intelligence: Evaluating the objective and subjective impacts of xai on human-agent interaction”. In: *International Journal of Human-Computer Interaction* (2022), pp. 1–15.
- [79] John Zerilli, Umang Bhatt, and Adrian Weller. “How transparency modulates trust in artificial intelligence”. In: *Patterns* (2022), p. 100455.
- [80] Philipp Schmidt, Felix Biessmann, and Timm Teubner. “Transparency and trust in artificial intelligence systems”. In: *Journal of Decision Systems* 29.4 (2020), pp. 260–278.

- [81] Alon Jacovi et al. “Formalizing trust in artificial intelligence: Prerequisites, causes and goals of human trust in AI”. In: *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*. 2021, pp. 624–635.
- [82] Theo Araujo et al. “In AI we trust? Perceptions about automated decision-making by artificial intelligence”. In: *AI & society* 35 (2020), pp. 611–623.
- [83] Wojciech Samek and Klaus-Robert Müller. “Towards explainable artificial intelligence”. In: *Explainable AI: interpreting, explaining and visualizing deep learning* (2019), pp. 5–22.
- [84] I Gede Agus Pertama Yudiantara et al. “Analysis of Acceptance and Success Implementation of Academic Information System (SIAk) Based on Technology”. In: *International Conference on Tourism, Economics, Accounting, Management, and Social Science (TEAMS 19)*. Atlantis Press. 2019, pp. 292–299.
- [85] William J Doll, Anthony Hendrickson, and Xiaodong Deng. “Using Davis’s perceived usefulness and ease-of-use instruments for decision making: a confirmatory and multi-group invariance analysis”. In: *Decision sciences* 29.4 (1998), pp. 839–869.
- [86] Nikola Marangunić and Andrina Granić. “Technology acceptance model: a literature review from 1986 to 2013”. In: *Universal access in the information society* 14 (2015), pp. 81–95.
- [87] Michael D Williams, Nripendra P Rana, and Yogesh K Dwivedi. “The unified theory of acceptance and use of technology (UTAUT): a literature review”. In: *Journal of enterprise information management* 28.3 (2015), pp. 443–488.
- [88] Yi-Hsuan Lee, Yi-Chuan Hsieh, and Chia-Ning Hsu. “Adding innovation diffusion theory to the technology acceptance model: Supporting employees’ intentions to use e-learning systems”. In: *Journal of Educational Technology & Society* 14.4 (2011), pp. 124–137.
- [89] Nina Hollender et al. “Integrating cognitive load theory and concepts of human–computer interaction”. In: *Computers in human behavior* 26.6 (2010), pp. 1278–1288.

- [90] John Hayes and Christopher W Allinson. “Cognitive style and the theory and practice of individual and collective learning in organizations”. In: *Human relations* 51.7 (1998), pp. 847–871.
- [91] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
- [92] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. “Reinforcement learning: A survey”. In: *Journal of artificial intelligence research* 4 (1996), pp. 237–285.
- [93] *Learn intro to game AI and reinforcement learning tutorials*. URL: <https://www.kaggle.com/learn/intro-to-game-ai-and-reinforcement-learning>.
- [94] *Minimax algorithm in Game theory: Set 4 (alpha-beta pruning)*. Jan. 2023. URL: <https://www.geeksforgeeks.org/minimax-algorithm-in-game-theory-set-4-alpha-beta-pruning/>.
- [95] Shixiang Zhou, Heejin Jeong, and Paul A. Green. “How Consistent Are the Best-Known Readability Equations in Estimating the Readability of Design Standards?” In: *IEEE Transactions on Professional Communication* 60.1 (2017), pp. 97–111. DOI: 10.1109/TPC.2016.2635720.
- [96] Interagency Advisory Panel on Research Ethics Government of Canada. *TCPS 2 (2018) – Chapter 2: Scope and approach*. Sept. 2019. URL: https://ethics.gc.ca/eng/tcps2-eptc2_2018_chapter2-chapitre2.html.
- [97] *Activities exempt from Human Ethics Review*. URL: <https://research.utoronto.ca/ethics-human-research/activities-exempt-human-ethics-review>.
- [98] *ChatGPT Introduction*. URL: <https://openai.com/blog/chatgpt>.
- [99] Andrea Arcuri and Lionel Briand. “A practical guide for using statistical tests to assess randomized algorithms in software engineering”. In: *Proceedings of the 33rd international conference on software engineering*. 2011, pp. 1–10.
- [100] Anthony Robins, Janet Rountree, and Nathan Rountree. “Learning and teaching programming: A review and discussion”. In: *Computer science education* 13.2 (2003), pp. 137–172.

- [101] Robert De Levie. *Advanced Excel for scientific data analysis*. Oxford University Press, USA, 2004.
- [102] Sammi Chow et al. “Automated data-driven hints for computer programming students”. In: *Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization*. 2017, pp. 5–10.
- [103] Ralf Lämmel and Simon Peyton Jones. “Scrap your boilerplate: a practical design pattern for generic programming”. In: *ACM SIGPLAN Notices* 38.3 (2003), pp. 26–37.
- [104] Jason Fox. *Tic Tac Toe: Understanding the minimax algorithm*. Nov. 2021. URL: <https://www.neverstopbuilding.com/blog/minimax>.
- [105] Jeremy Zhang. *Reinforcement learning-implement TicTacToe*. Aug. 2019. URL: <https://towardsdatascience.com/reinforcement-learning-implement-tictactoe-189582bea542>.
- [106] *CS 540 - Introduction to Artificial Intelligence*. URL: <https://pages.cs.wisc.edu/~dyer/cs540/>.
- [107] OpenAI’s ChatGPT AI language model. *Transcript of ChatGPT responses to the study’s reinforcement learning coding questions from a participant*. Online Transcription. Mar. 2023. URL: <https://chat.openai.com/chat>.

Chapter 6

Appendix

6.1 Research Ethics Approval

Research Ethics Board Application Form

Project Info.

File No: 1469193

Project Title: Assessing the impact of using Python to teach Computational Thinking for remote schools in a Blended Learning Environment

Principal Investigator: Dr. Vijay Mago (Science and Environmental Studies\Computer Science)

Start Date: 2022/04/01

End Date: 2023/04/27

Keywords: Computational Thinking, Computational Literacy, Coding, K-12 Education, Learning and Assessment, Programming

Related Awards:

Award File No	Principal Investigator	Project Title	Funding Snapshot	Notes
1468341	Vijay Mago	Nijii Indigenous Mentorship: Coding for the North/Science Odyssey	KPDSB ProgramResearch: Grant Type: Research Grant Requested CAD: 0.00 Awarded CAD : 0.00 LPS/LDSB ProgramResearch: Grant Type: Research Grant Requested CAD : 0.00 Awarded CAD : 0.00 MMSS ProgramResearch: Grant Type: Research Grant Requested CAD : 0.00 Awarded CAD : 0.00 NSERC ProgramPromoScience Grant Type: Research Grant Account Requested CAD: ██████████ Awarded CAD: ██████████ NSERC ProgramPromoScience Supplement for Science Odyssey Type: Research Grant Requested: ██████████ CAD Awarded CAD ██████████ TBCDSB ProgramResearch: Grant Type:Research Grant Request CAD 0.00Awarded CAD PROJECT TOTALS: Requested: CAD ██████████ Awarded : CAD ██████████	N/A

Project Team Info.

Principal Investigator

Prefix: Dr.

Last Name: Mago

First Name: Vijay

Affiliation: Science and Environmental Studies\Computer Science

Position: ASSO

Email: vmago@lakeheadu.ca

Phone1:

Phone2:

Fax:

Primary Address:

Institution: Lakehead University

Country: Canada

Comments: tenured

Other Project Team Member

Baxter	Denise	VP Academic & Provost \Indigenous Initiatives	Co- Investigator	dlbaxter@lakeheadu.ca
Harris	Lisa	VP Academic & Provost \Indigenous Initiatives	Co- Investigator	laharris@lakeheadu.ca
Kamak	Lakshmi Preethi	Science and Environmental Studies\Computer Science	Project Staff /Research Assistant	lkamak@lakeheadu.ca

Common Questions

1. General Information

#	Question	Answer
1.1	Type of Participants:	Children
1.2	Estimated Number of Participants (#):	28
1.3	Where will the research be conducted?	Lakehead Campuses (Thunder Bay or Orillia) Local (outside of campuses)
1.4	Have you received approval, or are you seeking approval from any other ethics committee?	No
1.5	If "Yes", which Ethics Committee(s)?	
1.6	Is this project's funding administered through Lakehead University?	Yes
1.7	Is this project's funding administered outside of Lakehead University?	No
1.8	If funded, Name of Granting Agency:	NSERC PromoScience

2. Preliminary Checklist

#	Question	Answer
2.1	Will your study involve more than minimal physical risk to your participants? (For a definition of minimal risk, see TCPS2, Chapter 2, Section B.)	No
2.2	Will your study involve the use of high-risk test instruments, i.e. surveys that may reveal that the participant intends to participate in dangerous activities such as self harm or harm to others?	No
2.3	Will your study involve more than minimal psychological risk to your participants?	No
2.4	Will your study likely lead to the discovery of your participants' involvement in illegal activities?	No
2.5	Will your study involve participants who are members of vulnerable populations?	Yes
2.6	If "Yes" to any of the above, please elaborate briefly:	Students in grades 7-8 of Armstrong Public School will be given the opportunity to participate in the classes and survey, which includes students who have identified as FNMI.
2.7	Will your study involve clinical research?	No
2.8	If "Yes", please elaborate briefly:	
2.9	Will your study involve First Nation, Metis and/or Inuit (FNMI) peoples as a "scheduled" representative group in the research?	No
	If "Yes", describe your collaboration or community engagement plan for guiding and monitoring the research with appropriate Aboriginal (FNMI) groups, including how the community has been engaged and plans for future engagement. For FNMI people residing outside of First Nations communities (ie. in urban centres), a similar plan of engagement with a representative FNMI group is required.	

2.10	For FNMI community research, provide additional details about your discussions around OCAP and CARE with the communit(ies). Include how you will implement OCAP and CARE Principles in your design and post-completion of the research. Does this include a research agreement with the communit(ies)?	
2.11	Will your study involve, effect or impact Aboriginal (FNMI) peoples as an "incidental" representative group in research?	Yes
2.12	If "Yes", please refer to TCPS 2, Article 9.2 and indicate your awareness of how your study may involve or impact FNMI peoples, in the short or long term. Demonstrate awareness by describing how your study will involve, effect or impact FNMI peoples. For example, is there a benefit of FNMI peoples in the longer-term research outcomes?	Long-term benefits include learning computational thinking skills and programming expertise at a young age, which is a critical 21st-century skill. Computational thinking skills are not only limited to programming but also help in the development of well-rounded problem-solving skills required in other fields [1]. This will help them to understand and equip themselves with in-demand skills and give them equal opportunities in the job market in the future. The course also teaches students how to adapt to a blended learning environment by introducing D2L. As members of a remote FNMI community, the participants often lack opportunities to develop these skills. The FNMI benefits of this research would be to provide objective results that may allow for future CT skill-building options amongst other remote FNMI communities.
2.13	I confirm that if I become aware that my participant pool contains a sizeable proportion of FNMI participants, I will advise the REB of my new research situation by submitting an amendment/addition through the Romeo Research Portal.	Yes

2.14	<p>*Applicable if working with First Nations communities - I acknowledge that when collecting data with First Nation communities, the data is owned by the First Nation community in accordance with the OCAP principles.</p>	
------	---	--

3. Research Ethics Review Criteria

#	Question	Answer
3.1	<p>LAY DESCRIPTION: Provide a brief lay-word summary of the proposed project (40 words or less, similar to the statement you would prepare for a granting agency for public dissemination).</p>	<p>The proposed research will evaluate and compare the grade 7, 8 student performance and receive feedback for the programming course taught using computational thinking methods. A customized blended learning environment with online and in-person classroom teaching materials is provided on the D2L platform. The student feedback allows for a dynamic learning environment to assimilate feedback efficiently.</p>
3.2		<p>Computational Thinking (CT) is an essential 21st-century skill and it is being increasingly integrated into the K-12 curriculum across the globe. Numerous studies [1] [2] suggest that CT skills are an essential part of the development of cognitive skills for students that may apply in everyday problems. CT primarily develops skills in algorithmic thinking, abstraction, problem decomposition, pattern recognition, logical thinking, and data representation. However, there is no standard way for educators to teach and improve CT to students. It is still in the early phases of curation and is not widely used in standard school curriculums. This study uses approved and widely used CT learning methodologies to highlight the need for CT training for students. A recent study [3] on CT has shown that children were more interested in programming after CT skills development. The research aims to encourage the development of computational abilities to increase interest in coding. The teaching resources have been carefully designed to combine with students' programming and mathematical concepts,</p>

	<p>SUMMARY OF PURPOSE OF RESEARCH: Be sure to include sufficient detail, described in terms that do not require extensive field-specific knowledge. Include your research question(s).</p>	<p>allowing them to better understand their subject. Survey questions are carefully curated from validated CT tests and tailored to the student's grade.</p> <p>The structure of the teaching materials is based on the Ontario curriculum's learning outcomes for coding concepts for grades 7 and 8. The research team provides the teachers with instructional materials. All the course materials are hosted on the D2L platform by Lakehead University. D2L is an emerging platform for post-secondary education and provides students with the necessary experience to navigate the course activities. The resources will include</p> <p>programming video tutorials, worksheets, and an instructor's manual. Teachers will</p> <p>conduct the School Site courses using the materials and instructor's manual. The researchers will also have in-person visits to build relationships with students and communicate effectively from time to time. Activities with the school's teachers and students are planned based on the academic year. The Blended learning platform also allows students to adapt to the dynamic learning environment with the recent lockdowns due to the pandemic. The course will cover one academic year. The survey and test will be administered in parts towards the end of the course. The survey questions will mainly assess their knowledge in programming concepts, computational thinking skills, and learning interests. Three surveys adapted from the Blended Learning Questionnaire (BLQ) and the Computational Thinking Questionnaire (CTQ) will be conducted, and the test scores will be collected by the research team through the D2L platform. We'll evaluate and assess the results for efficacy, level of students' learning efficiency, as well as their interest in learning.</p>
--	--	---

3.3	<p>RESEARCH PARTICIPANTS: Describe required characteristics and number of participants.</p>	<p>The surveys will be taken by grade 7 and 8 students who are attending the “Coding with Python” course at Armstrong Public School. This is the only school that is included in this research, no other schools will be involved. Age is used to describe the characteristics of survey participants. The students will take the survey on the D2L platform hosted by Lakehead University.</p>
3.4	<p>DATA COLLECTION: Explain the method of data collection and analysis. Explain exactly what will be expected of participants (length of time commitment, etc.) All questionnaires and research instruments must be included as appendices.</p>	<p>In order to evaluate the survey results, we use a variety of analytical methods. The analytical methods include exploratory factor analysis and qualitative comparative analysis. Since the dataset is small, the data will not be normally distributed and non-parametric tests will be performed[3]. During the course, participants are requested to complete a survey on the D2L platform. D2L is introduced as part of the python curriculum. The course and survey both use the same D2L account for student convenience. However, in the event of poor internet connectivity, paper surveys will be provided and the student's performance in the paper surveys will be uploaded to their D2L account. The survey introduction serves as a registration page and will collect the student's name, student ID and age(optional).The characteristics of survey participants are described using age. To maintain confidentiality, the information will be de-identified after data collection.</p> <p>Individual information will not be collected; instead, students will be identified by their student ID number. All the submitted surveys will also be aggregated together by the student ID number. Each survey session and examination would be used to collect various test scores and feedback information. This includes a CT skill assessment, programming concepts, student learning interests, and course feedback. The survey will be conducted in three parts over the course duration.</p>

		<p>Participants will have 20 minutes to complete each survey. The short surveys enable the student to bring more focus and attention to the subject. The majority of the survey questions will be multiple-choice and ranking questions using the Likert scale. There are also problems that require long-form answers in the Computational Thinking section of the survey. The surveys are adapted from the following two questionnaires; the Blended Learning Questionnaire (BLQ) and the Computational Thinking questionnaire (CTQ). The CTQ questionnaire will question participants regarding CT skills. The CTQ questionnaire asks participants about their experience in programming, for example, how much they are interested in data analysis, and solving complex problems. It mainly focuses on python programming concepts and analyzing code snippets. The BLQ questionnaire will contain regarding the structure of the course, their learning interests, and feedback on our course.</p>
3.5	<p>SECONDARY DATA: For research involving the use of secondary data (data which has been previously collected for a purpose other than the research project itself), REB review is not required if the data is anonymized so long as the process of data linkage or recording or dissemination of results does not generate identifiable information (see TCPS 2, Chapter 2, Article 2.4). For secondary data that is identifiable, please see TCPS 2, Chapter 5, Section D.</p>	<p>No secondary data will be used.</p>
	<p>RECRUITMENT PROCEDURES: Describe how potential participants will be selected</p>	<p>The coding course - "Coding with Python" is part of the school curriculum and separate from the survey. The coding classes form the basis for the coding component of the school's math program. The survey is a research opportunity provided by teaching the python course. The survey is designed to evaluate the computational thinking that the students are expected to demonstrate at the end of the course. This survey is entirely voluntary and students can decide to participate or not.</p>

3.6	<p>and contacted. Include a copy of any advertisements used to recruit participants. If you intend to offer an incentive to participants, please describe the incentive, monetary or otherwise, along with justification as to the use and level of the incentive (TCPS2, Chapter 3, Article 3.1). Where incentives are offered to participants, they should not be so large or attractive as to encourage reckless disregard of risks. The voluntariness of consent must remain.</p>	<p>Participation or lack thereof in the survey will not affect any grades for the students. The survey will be offered to the grade 7 and 8 students who are participating in the “Coding with Python” Course at the Armstrong Public School.</p> <p>The teachers will collect the survey results and share them with DaTALab, Lakehead University. Corey Dagenais, the Armstrong Public School Principal, has approved the School’s participation in the course and provided permission for the Lakehead researchers to conduct the study at Armstrong Public School (Appendix - B: Letter of Support). A raffle will be held for participants who participate in all surveys. The prize will be an educational toy of approximately \$25 appropriate for students.</p>
3.7	<p>HARM and/or POTENTIAL RISKS to PARTICIPANTS: (a) State clearly any potential harm or risks - physical, psychological, injury to reputation or privacy, and breach of any relevant law - for participants or for third parties (those affected by the research but who are not active research subjects); (b) If there is any apparent or potential harm or risk, clearly explain all steps that are being taken to reduce this.</p>	<p>There are no expected physical or psychological risks to participants.</p>
3.8	<p>DECEPTION: If deception is part of the research program, the researcher must: (a) State clearly why no alternative methodology, which does not involve deception, can fruitfully be used to answer the research question; (b) Provide evidence that the participant is not put at risk by the deception. If appropriate, provide a debriefing letter to participants disclosing the deception.</p>	<p>Deception is not a part of this research.</p>
		<p>The potential benefit of this study is the increased computational thinking capabilities and effective programming skills at a young age. Computational skills are not only limited</p>

<p>3.9</p>	<p>BENEFITS to PARTICIPANTS and/or SOCIETY: Describe in detail the potential benefits of the research for both participants and to general knowledge.</p>	<p>to programming but also help in the development of well- rounded problem-solving skills required in other fields. This will help them to understand and equip themselves with in- demand skills like programming and give them equal opportunities in the job market in the future. It also teaches students how to use their strengths and points of view to contribute to future work or academic environments. The learning outcomes of these courses align with the Ontario coding curriculum. The CT assessment tools assist us in determining where students are falling behind and can assist teachers in better explaining concepts to children using computational metaphors. As a result, students who need to focus on specific ideas will receive more individual attention. Since there is a limited understanding of this subject, this study also helps teachers develop a culture and mindset around this subject and implement it efficiently into schools.</p>
<p>3.10</p>	<p>INFORMED CONSENT: Clearly outline the measures that will be used to ensure the informed consent of all research participants. Cover letters and consent forms must be attached as appendices on Lakehead University (or NOSM if appropriate) letterhead.</p>	<p>The coding course - “Coding with Python” is conducted as a part of the school’s curriculum. The grades from this course are considered a component when determining the math grades. Teachers at the school will be responsible for developing the examination, and grading for the course. The course is separate from the survey. The survey is optional and the score will not have any impact on the student’s grades or the school’s standing. Detailed survey information will be in the participant information letter provided to the students 7 days before the first survey.</p> <p>Before each survey, the students will be required to confirm they have read the information letter and provide consent by clicking a box affirming their understanding and willingness to participate in this research [4]. However, in the event of poor internet connectivity, paper consent forms and surveys will be provided and the students can provide consent using the consent form. In this situation, paper copies of the information letter</p>

		<p>will be available to ensure participants are fully informed. Based on Article 4.6 of the TCPS 2 [4], the student participants' capacity to consent is not based on age, but on the following conditions: The research has no expected or minimal expected risk. The research presents the prospect of benefits to participants or to a group that is the focus of the research and to which the participants belong. The research participants are capable of understanding the risks, benefits and nature of the research. These criteria are clearly provided in the information letter and consent form, additionally, the potential risks are very low. Thus, the students should have the capacity to provide consent for themselves [4].</p>
3.11	<p>CAPACITY TO CONSENT: Capacity refers to the ability of prospective or actual participants to understand relevant information presented about a research project, and to appreciate the potential consequences of their decision to participate or not participate (TCPS 2, Chapter 3, Section C). Will the research participants sufficiently understand the nature of the research project, and the risks, consequences, and potential benefits associated with it?</p>	Yes
3.12	<p>If "No", please state why this vulnerable group is necessary to the study, and elaborate on the consent process (i.e. parental consent, caregiver consent).</p>	
3.13	<p>RIGHT TO WITHDRAW: The researcher must illustrate that participants will be informed of their right to withdraw from the study at any time without penalty of any kind, and that they may choose not to answer any question asked as part of the research. For participants submitting information anonymously, the participants must be informed that withdrawal post-submission is not possible due to the anonymous nature of their data. Will the participants have the right to withdraw?</p>	Yes

3.14	If "No", please elaborate:	
3.15	<p>ANONYMITY and/or CONFIDENTIALITY:</p> <p>The researcher must outline the procedures that will be used to guarantee confidentiality and/or anonymity for participants. Participants who wish to be named and to waive their right to privacy and confidentiality must provide written evidence.</p>	<p>Participants' student IDs and demographic information will be stored separately and securely on a password-protected database. The raw survey results will not be shared publicly and it is the only used to find a correlation between CT training and examination scores surveyed. Only the aggregate results will be shared in a detailed report, made available to the administration and participants upon request. Care will be taken to ensure that results will not serve to identify individual students.</p>
3.16	<p>STORAGE of DATA: Detail how the data will be securely stored for a minimum of 7 years following completion of the research, at Lakehead University, as per Lakehead University policy.</p>	<p>The student ID will be stored separately on an encrypted and password-protected database. Other collected information such as survey response data, exam score data will be stored on secure password-protected cloud servers of DaTALab Lakehead University for a period of 5 years under the responsibility of Dr. Vijay Mago, the supervisor for this study.</p>
3.17	<p>PEER REVIEW: State the intention, or non-intention to have the proposal peer reviewed by an external granting agency or thesis committee, if appropriate. If the REB determines the project to be of more than minimal risk, peer review may be required by the addition of ad-hoc members to the REB, even if the granting agency for the project does not require this, or if the project is not funded.</p>	<p>We seek additional peer review for this research as it is instrumental to the research member's thesis. The thesis document will be reviewed by a committee. We also intend to publish the paper in journals or conferences upon completion.</p>
3.18	<p>RESEARCH PARTNERS and STUDENTS: Clearly state whether or not the research will involve student researchers or if the research will involve researchers at another university/institution. TCPS2 Tutorial Certificates for all research partners and students must be attached.</p>	<p>The research involves student researcher - Ms. Lakshmi Preethi Kamak from Lakehead University, a Graduate Student of Computer Science who is also working as Research Assistant in DaTALab under the supervision of Dr. Vijay Mago. The TCPS2 tutorial has been completed by all the researchers and the certificate is also attached for everyone.</p>

3.19	<p>MULTI-JURISDICTION RESEARCH: If you are involved in multi-jurisdictional research, provide evidence that ethical approval is also being sought at any other institution where direct research with human participants will be undertaken. Ethical approval from another institution, while essential in a multi-jurisdiction project, is not itself sufficient for the commencement of research with human participants at Lakehead University.</p>	<p>Our study does not involve multi- jurisdictional research.</p>
3.20	<p>CONFLICT of INTEREST: Disclose any real, perceived or potential conflicts of interest (professional, personal or financial) to the Research Ethics Board. NOTE: It is preferable to avoid or prevent being in a conflict of interest, when possible. When it is not possible to avoid a conflict of interest, then it should be disclosed to the appropriate people and steps taken to minimize or manage the conflict.</p>	<p>We have no conflict of interest to declare.</p>
3.21	<p>DISSEMINATION of RESEARCH RESULTS: Clearly state the means by which the research will be disseminated and by which research participants may be made aware of the findings of the study. When requested by participants, research results must be provided in easily understandable language and approved by the Principal Investigator.</p>	<p>Research results will be disseminated in publications in professional conferences and journals. Open-access journals will be favoured, thus providing access to our findings to all participant</p>
3.22	<p>I have completed the TCPS 2 Tutorial: Course on Research Ethics (CORE) and have attached a copy of my completion certificate to this application. *Please note that all investigators listed on this application must submit their certificates.</p>	<p>Yes</p>
3.23	<p>I am familiar with the Agreement on the Administration of Agency Grants and Awards by Research Institutions, and the Tri-Council Policy Statement 2: Ethical Conduct for Research Involving Humans and I agree to comply with these guidelines, and the procedures approved by the REB, in carrying out this proposed research.</p>	<p>Yes</p>
3.24	<p>I attest that all information submitted to the REB is complete and truthful. I understand the consequences, for myself and for the</p>	<p>Yes</p>

	institution, of failure to comply with the above regulations.	
3.25	Researchers are required to report to the REB any changes in research design, procedures, sample characteristics, and so forth that are contemplated after REB approval has been granted. Changes may not be implemented until approved by the REB. If any unforeseen incident occurs during the course of research that may indicate risk to participants, I will immediately cease research and inform the REB.	Yes
3.26	I understand that my protocol will be subject to random review for compliance by the Office of Research Services.	Yes
3.27	I will inform the REB when the research is complete by completing the Final Report Form (see New Event Forms in the Romeo Research Portal).	Yes

4. COVID-19 Measures (05/2022)

#	Question	Answer
4.1	Does your protocol involve face-to-face/in-person research?	Yes (If Yes, please complete this tab).
4.2	DOES NOT APPLY after April 30, 2022. All Lakehead University research team members conducting in-person research must have taken the Lakehead University COVID-19 Awareness Training found through the MyCourseLink (D2L) portal and be fully vaccinated against COVID-19. Have all Lakehead University team members completed the training and are fully vaccinated?	Yes
4.3	DOES NOT APPLY after March 16, 2022, please proceed to 4.7. Rationale for In- Person Research (check all that apply):	The research is time sensitive. The research poses minimal risk to the transmission of COVID-19.
4.4	DOES NOT APPLY after March 16, 2022, please proceed to 4.7. Provide details if selected "Other" above:	The researchers will also have in-person visits to build relationships with students and communicate effectively from time to time. Blended learning is part of the study; hence, in-person is crucial for the research.

4.5	DOES NOT APPLY after March 16, 2022, please proceed to 4.7. Explanation and justification for in-person research, including if alternatives have been considered:	Yes, The research is designed for a dynamic blended learning environment with alternative methods like online classes via the D2L platform.
4.6	DOES NOT APPLY after March 16, 2022, please proceed to 4.7. Describe any known participant vulnerabilities, i.e., immunocompromised, existing medical conditions, elderly, working in close contact with those who may be infected by COVID-19, recent traveler, socially and economically vulnerable populations, etc. If there are participant vulnerabilities, please describe how you will mitigate these vulnerabilities with additional safety precautions.	There is no declared participant vulnerability.
4.7	If applicable, has the partner approved in-person research during COVID-19? Attach evidence of partner approval, i.e., letter of support, email, etc. Examples of partners include Indigenous communities (see questions below), school boards, long-term care homes, hospital/health care facilities, etc. Explain below.	Letter of Support for the School principal is attached.
4.8	Indigenous communities have developed their own policies and regulations regarding COVID-19. Researchers working with/in Indigenous communities must be aware of and conform to these regulations.	I have consulted with the Indigenous community(s) in which I am partnering regarding their COVID-19 regulations. I will follow this direction and am prepared to and agree to halt the research at any time if requested by the Indigenous community(s).
4.9	It is the duty of the Principal Investigator (PI) to seek out information regarding the community(s) in which they are working to determine if in-person research is allowable during this time. The PI must take the lead from the community and follow their regulations as appropriate. Please detail below the community regulations, i.e., quarantining necessary upon arrival in community and how this will be accomplished, outdoor activities only allowed, regulations for medical mask wearing/physical distancing, etc.	The School regulations regarding COVID safety for students and visiting researchers will be strictly followed. Face masks will be worn all the time. Activities are designed such that there will be no physical contact. The physical distancing will always be maintained. Sanitizers will be used from time to time to ensure hygiene.

<p>4.10</p>	<p>DOES NOT APPLY after March 16, 2022, please proceed to 4.12. Checklist of information to include in the Human Participant Safety Plan.</p>	<p>While conducting in-person research, research team members must confirm DAILY that they are not currently experiencing, or have experienced within the past 14 days, any symptoms of illness, and they have not had recent contact with a known or presumed COVID-19 patient. If yes to any of these questions, their participation as part of the research team is halted. All research team members will complete a COVID-19 self-assessment DAILY (Ontario Ministry of Health Self- Assessment online) or the Lakehead University MobileSAFETY app. Even though safety measures are in place, if a research team member is showing flu-like symptoms, the research is suspended immediately. Researchers must contact research.ethics@lakeheadu.ca, or 807- 343-8283, immediately should this occur.</p> <p>The REB will assess the situation and determine how to proceed, as per TCPS2, Article 6.15. Physical distancing of 2m is maintained whenever possible. Wear medical masks when it is not possible, even if outdoors. Research team members have medical masks available and are wearing them correctly. Hand sanitizer is available for research team members use, and use is required at the beginning and end of the data collection process. Research team members are reminded of cough/sneeze etiquette and hand washing protocols. Each team member is required to maintain a personal contact log describing, day by day, all persons that he/she has been in contact with, including “off” hours (e.g. research participants, other team members, family members, etc.). The contact log shall be kept for 30 days, and then safely destroyed. Below is language to include in participant Information Letters and Consent Forms regarding contact tracing.</p>
	<p>DOES NOT APPLY after March 16, 2022, please proceed to 4.12. Please confirm you have added the following statement to your Information Letters and Consent Forms (for</p>	

4.11	<p>contact tracing purposes only): "Please note that our research team is required to keep logs for the purposes of contact tracing beyond our social circle. We will request your name and telephone number for this purpose. If a research team member or research participant(s) contracts COVID-19, the log would be shared with health authorities if requested. Only your name and telephone number, and not the reason for contact, would be shared with health authorities. Your information will be combined with all other contacts and you will not be identified as a participant in this research study. Contact logs are kept for 30 days, then all identifying information is destroyed". To add to Consent Forms: "I understand that the research team will take my name and telephone number for contact tracing purposes, and that this information will only be disclosed to health authorities (if requested) should a research team member or participant(s) contract COVID- 19".</p>	Yes
4.12	<p>For Information Letters for risk of in-person research, Researchers must provide revised Information and Consent Letters that include an advisory to participants that, due to the COVID-19 pandemic, in-person research carries greater or additional risk. Include the protective measures you will take as a researcher: for example, meeting outdoors, wearing a medical mask, one-on- one interview instead of focus groups, sanitizing/cleaning to be done, etc. The Consent Form must contain the following statement, "I understand that there are risks of contracting COVID-19 during in- person research."</p>	Yes
4.13	<p>DOES NOT APPLY after April 30, 2022. List any additional physical distancing/safety measures required or to be taken by your team in the Plan.</p>	<p>The Computational Thinking group activities were designed such that there isn't any physical contact involved. All the Armstrong Public school protocols for COVID safety measures were strictly followed. The school had appropriate social distancing, compulsory face mask norms and no contact</p>

		policy.
4.14	<p>DOES NOT APPLY after March 16, 2022, please proceed to 4.15. Plan for Research Participants: As researchers, you play a role in protecting the safety of participants that are engaged in your research projects. This checklist is provided to help you put a plan in place to ensure that your research participants can and are able to participate safely in your study:</p>	<p>Each day that in-person research is being conducted, and for each participant participating on that day (including any that have participated previously), the research team must request that those participants confirm that they are not currently experiencing any symptoms of illness or within the past 14 days and they have not had recent contact with a known or presumed COVID-19 patient. If yes to any of these questions, their participation is halted. All research participants must complete a COVID-19 self-assessment DAILY prior to the start of the data collection (Ontario Ministry of Health Self- Assessment online). Travel has been arranged for the participants that maintains physical distancing. Once at the research site, participants will maintain 2m physical distancing whenever possible. Provide medical masks when/if participants don't have them available, even if outdoors. Shared space is cleaned appropriately between participant use. Hand sanitizer is available for participant to use, and use is required at the beginning and end of each in-person session and as needed throughout. Even though safety measures are in place, if a research team member is showing flu-like symptoms, the research is suspended immediately. Researchers must contact research.ethics@lakeheadu.ca, or 807-343-8283, immediately should this occur. The REB will assess the situation and determine how to proceed, as per TCPS2, Article 6.15. Participants are reminded of cough/sneeze etiquette and handwashing protocols.</p>
4.15	<p>DOES NOT APPLY after April 30, 2022. Checklist of Information to include for research taking place on campus:</p>	

5. Informed Consent Checklist (to assist Applicants)

#	Question	Answer
5.1	General	<p>Cover letters and consent forms are presented on Lakehead University letterhead (or NOSM if appropriate) The language level is appropriate to the age and reading level of the participant population Contact information for the researcher(s) (including the supervisor if a graduate student project) and the REB is always included in the cover letter that the participants will keep after they sign the consent form. Suggested wording: "This study has been approved by the Lakehead University Research Ethics Board. If you have any questions related to the ethics of the research and would like to speak to someone outside of the research team please contact Sue Wright at the Research Ethics Board at 807-343-8283 or research@lakeheadu.ca."</p>
5.2	<p>The Cover Letter/Introductory Information (including electronic letters and consent forms) should include:</p>	<p>The title of the study An explanation of the purpose of the research The identity of the researcher and their affiliation with Lakehead University The funder of the research, if applicable A warm, non- coercive invitation to participate, addressed to the "Potential Participant" The reason why the potential participant is being invited to participate in the research That the individual's participation is voluntary, that they may refuse to participate in any part of the study, and that they may withdraw from the study at any time (other than anonymously submitted information) That participants may decline to answer any question A description of the procedures the participants will be involved in and how much of their time will be required Information regarding any audio or videotaping and explicit consent to such recording Information about any foreseeable risks, harms, or inconveniences Potential benefits (including information that there is no direct benefit, if appropriate) A mechanism for providing referrals, if appropriate (ie. if there is the possibility of emotional distress or</p>

		<p>physical harm) Information regarding who will have access to the data Information about the storage of data (during and after completion of the research) The degree of confidentiality and/or anonymity that will be provided and how this will be maintained (eg. individual participants will not be identified in published results without their explicit consent, data will be published in aggregate form). For research involving anonymous surveys, it should be stated that the survey instrument will not be labelled to identify who completed it Limits on confidentiality, if applicable (eg. confidentiality disclaimer for focus groups) A statement indicating the researcher's intent to publish or make presentations based on the research and whether or not the participant's identify will remain confidential (eg. will pseudonymous be used?) Offer of summary of the results (and a mechanism to provide the summary)</p>
5.3	<p>The Consent Form must state each individual's agreement that:</p>	<p>They have read and understood the cover/information letter for the study They agree to participate They understand the potential risks and/or benefits of the study, and what those are That they are a volunteer and can withdraw from the study at any time, and may choose not to answer any question The data they provide will be securely stored at Lakehead University for a minimum of 5 years following completion of the project If applicable, that they understand that the research findings will be made available to them, and how this will be communicated That they will remain anonymous in any publication/public presentation of research findings.</p> <p>Participants must explicitly agree to have their identities revealed.</p>

5.4	Other Consent Information	<p>All participants must sign and date the consent form then return it to the researcher Consent must also be obtained from all agencies, partners, schools, school boards, etc. that provide access to the subject pools. Separate consent forms must be included for all of the above should this apply While inclusive research is important, the researcher must ensure that consent is obtained from vulnerable populations in a sensitive manner.</p> <p>Vulnerable populations include children, & others not competent to give free and informed consent on their own behalf. In cases like this, parent/guardian (or the individual's representative) consent must be obtained. Please note every effort should be made to ensure that participants understand and consent to their own participation as well. If representative consent will be obtained the researcher must explicitly demonstrate why this is necessary and how the research results could be significantly altered if representative consent was required.</p>
-----	---------------------------	---

Attachments

Doc / Agreement	Version Date	File Name	Description
		08_Section_4.10_COVID-19 Measures Safety Plan.pdf	Appendix - COVID Safety Plan
		08_Appendix_for_Ethics Approval Document.pdf	Appendix- Reference
		07_REB_signature_page_NSERC_PromoScience_Signed.pdf	REB signature
		06_Covid AwarenessSiteCertificate_InpersonCI_LisaHarris.pdf	COVID-19 - LH

		06_Covid AwarenessSiteCertificate_InpersonRA_Lakshmi.pdf	COVID-19 - LPK
		05_tcps2_core_certificate_Lisa_CoInvestigator.pdf	TCPS - LH
		05_tcps2_core_certificate_Lakshmi_ResearchAssistant.pdf	TCPS- LPK
		05_tcps2_core_certificate_Denise_CoInvestigator.pdf	TCPS - DB
		04_tcps2_core_certificate_Vijay_PrincipalInvestigator.pdf	TCPS -VM
		03_Advertisements_CodeathonPoster.pdf	Advertisement
		Invitation - Information Letter Consent form - April 5.pdf	Invitation/Consent form
		PromoScience Survey- April 5.pdf	N/A
		08_Appendix_B_Letter of Support.pdf	N/A
Approval Letter	2022/04/27	1469193 approval letter.pdf	N/A

April 27, 2022

Research Ethics Board
t: (807) 343-8283
research@lakeheadu.ca

Principal Investigator: Dr. Vijay Mago
Co-Investigator: Denise Baxter and Lisa Harris
Project Staff/Research Assistant: Lakshmi Preethi Kamak

Science and Environmental Studies\Computer Science
Lakehead University
955 Oliver Road
Thunder Bay, Ont.

Dear Dr. Vijay Mago and Research Team:

Re: Romeo File No: 1469193
Granting Agency: TBCSB, NSERC, Migizi Miigwanan Secondary School, Lakehead Public Schools, Keewatin Patricia District School Board
Agency Reference #: 1468341

On behalf of the Research Ethics Board, I am pleased to grant ethical approval to your research project titled, "Assessing the impact of using Python to teach Computational Thinking for remote schools in a Blended Learning Environment".

Ethics approval is valid until April 27, 2023. Please submit a Request for Renewal to the Office of Research Services via the Romeo Research Portal by March 27, 2023, if your research involving human participants will continue for longer than one year. A Final Report must be submitted promptly upon completion of the project. Access the Romeo Research Portal by logging into myInfo at:

<https://erpwp.lakeheadu.ca/>

During the course of the study, any modifications to the protocol or forms must not be initiated without prior written approval from the REB. You must promptly notify the REB of any adverse events that may occur.

Best wishes for a successful research project.



Sincerely,



Dr. Kristin Burnett
Chair, Research Ethics Board

/sa

6.2 Student Questionnaire: HCI Study

In this section, you are asked which instructions “Pacman” () should follow to get to the “Ghost” (). The `move_forward(x)` function tells Pacman to move x steps in the *direction* it is currently facing. Pacman can be facing left, right, up, or down. Currently, Pacman is starting at position **A1** on the grid.

EXAMPLE

```
move_forward(1)
```

The Pacman moves forward in one step

```
turn_right()
```

The Pacman turns 90 degrees to the right

```
turn_left()
```

The Pacman turns 90 degrees to the left

Figure 6.1: Computational thinking Algorithm introduction [50]

1. Which sequence of instructions will take Pacman to the Ghost located at position **C3** [50]?

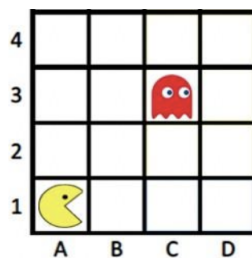


Figure 6.2: Computational Thinking Question 1 [50]

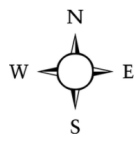
Select the correct sequence:

- A) `move_forward(1)`
`move_forward(1)`
`turn_left()`
`turn_right()`
`turn_left()`
- B) `move_forward(1)`
`move_forward(1)`
`turn_left()`
`move_forward(1)`
`move_forward(1)`
- C) `turn_right()`
`turn_right()`
`turn_left()`
`move_forward(1)`
`move_forward(1)`
- D) `move_forward(1)`
`move_forward(1)`
`turn_right()`
`move_forward(1)`
`move_forward(1)`

Figure 6.3: Computational Thinking Question 1 options

... ..

ey.
 ar
 ng



Instructions:

```
turn_left()
move_forward(1)
turn_right()
move_forward(2)
turn_left()
move_forward(3)
```

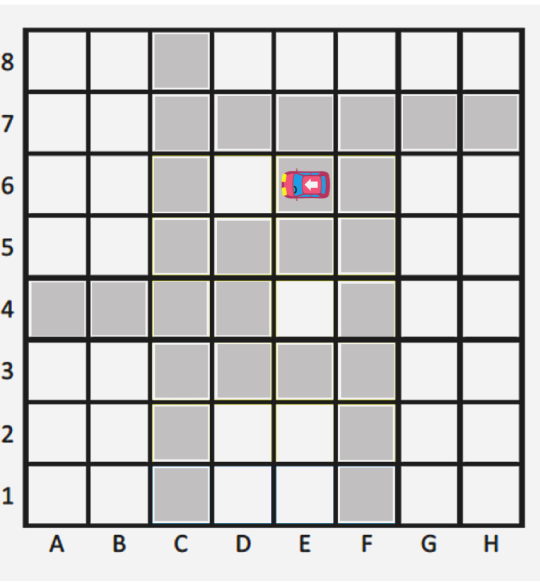


Figure 6.4: Computational Thinking Question 2 [50]

- If the car follows the instructions above, where will it end up?
 - C2
 - C8
 - D4
 - E4
 - At the end of the instructions, which direction will the car face?
 - North
 - West
 - East
 - South
3. Pacman is starting at position **A5** on the grid. Which sequence of instructions will take Pacman to the Ghost located at position **A1** **following the green path** [50]?

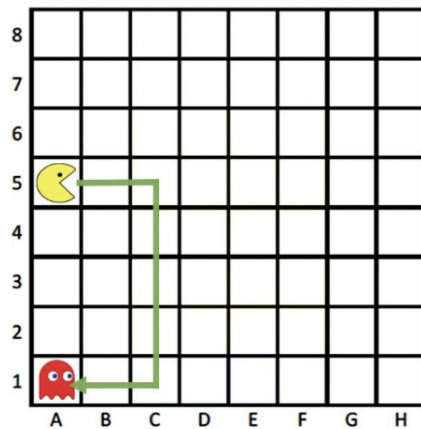


Figure 6.5: Computational Thinking Question 3 [50]

6.2.

A)	<pre>turn_right() turn_right() move_forward(1) move_forward(1) move_forward(1) move_forward(1) turn_left() turn_left()</pre>	C)	<pre>turn_right() turn_right() move_forward(1) move_forward(1) move_forward(1) move_forward(1) turn_right() turn_right()</pre>
B)	<pre>move_forward(1) move_forward(1) turn_right() move_forward(1) move_forward(1) move_forward(1) move_forward(1) turn_left() move_forward(1) move_forward(1)</pre>	D)	<pre>move_forward(1) move_forward(1) turn_right() move_forward(1) move_forward(1) move_forward(1) move_forward(1) turn_right() move_forward(1) move_forward(1)</pre>

Figure 6.6: Computational Thinking Question 3 options

4. The following set of instructions helps the car at **B1** reach home at **orange patterned tile** using one of the paths marked out [50].

Instructions:

```
while(car_destination !=  
'home'):  
    if(front_side == 'clear'):  
        move_forward(1)  
    else:  
        if(left_side == 'clear'):  
            turn_left()  
        else:  
            turn_right()
```

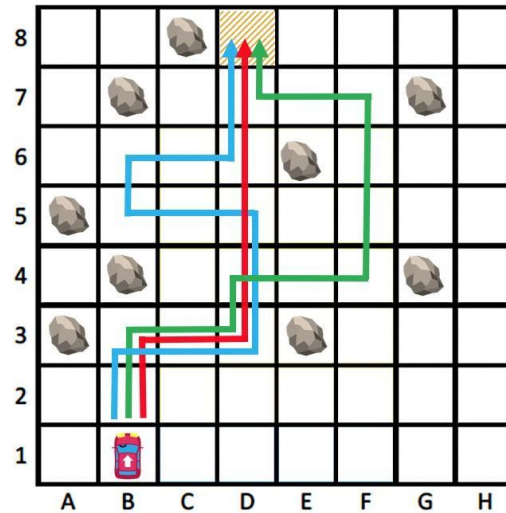


Figure 6.7: Computational Thinking Question 4 [50]

According to the instructions, which coloured path will the car take?

- The red path
- The blue path
- The green path
- None of the above

5. A car starts at position **A2** and faces in the east direction as shown by the white arrow on the roof of the car. The car can never pass through a rock or share the same position as a rock. The car can pass only through the clear space that does not contain a rock and can navigate only within the grid boundary [50].

Which set of instructions correctly take the car to the **orange patterned tile at B1**?

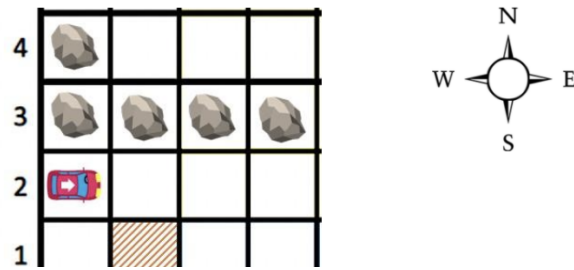
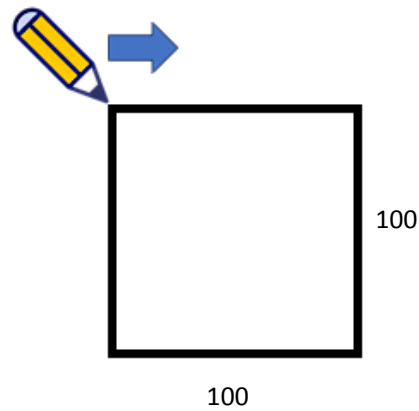


Figure 6.9: Computational Thinking Answer 5

6. The instructions are given at the upper left corner of the square.



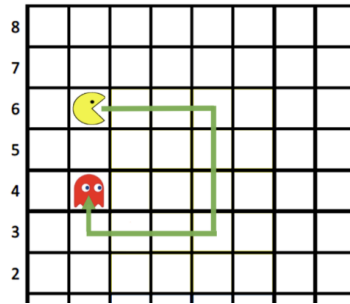
```
for x in range(4):  
    move_forward(100)  
turn_right()
```

Figure 6.10: Computational Thinking Question 6 [50]

What is the **error** in the instructions?

- The “repeat” loop should repeat 2 times.
- The “move” block should be outside the loop.
- The “turn right” block should be inside the loop, after the “move” block.
- The “turn right” block should be inside the loop, before the “move” block.

7. Which set of instructions take the “Pacman” at **B6** to the “Ghost” at **B4** using the path marked out [50]?



- A)

```
for x in range(3):
    move_forward(3)
    turn_right()

move_forward(1)
```
- B)

```
for x in range(4):
    move_forward(3)

turn_left()
move_forward(1)
```
- C)

```
for x in range(4):
    move_forward(3)
    turn_right()

move_forward(1)
```
- D)

```
for x in range(3):
    move_forward(3)

turn_right()
move_forward(1)
```

Figure 6.12: Computational Thinking Question 7 options

8. A car starts at position **F2** and faces in the west direction as shown by the white arrow on the car. The car can never pass through a rock or share the same position as a rock. The car can pass only through the “clear space”, a tile that does not contain a rock and can navigate only within the grid boundary [50].

Which set of instructions correctly take the car to the orange patterned ‘home’ tile at **B5** using the path marked out?

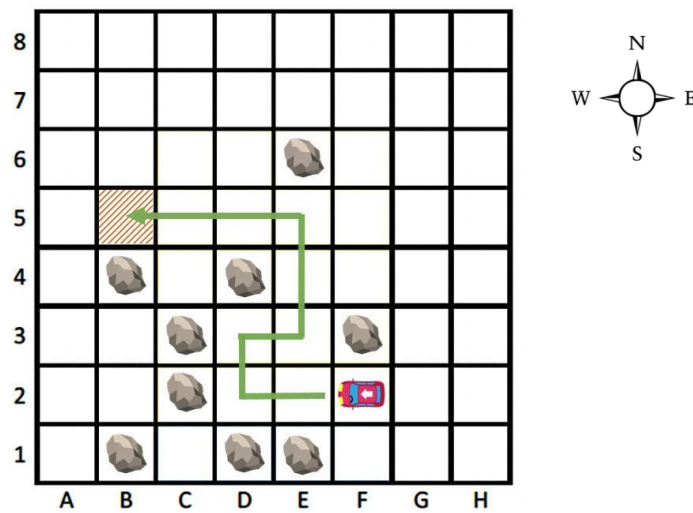


Figure 6.13: Computational Thinking Question 8

A)

```

while(car_destination !=
'home'):
    if(left_side == 'clear'):
        turn_left()
        move_forward(1)
    else:
        turn_left()
        move_forward(1)

```

B)

```

while(car_destination !=
'home'):
    if(left_side == 'clear'):
        turn_left()
    else:
        turn_right()
        move_forward(1)

```

C)

```

while(car_destination !=
'home'):
    if(front_side == 'clear'):
        move_forward(1)
    if(right_side == 'clear'):
        turn_right()
        move_forward(1)
    turn_left()

```

D)

```

while(car_destination !=
'home'):
    if(front_side == 'clear'):
        move_forward(1)
    else:
        if(left_side ==
'clear'):
            turn_left()
        else:
            turn_right()
    turn_left()

```

Figure 6.14: Computational Thinking Question 8 options

Acknowledgements

We acknowledge that this survey was adapted from the established Callysto CTQ works [50].

6.3 Student Questionnaire: AI Context

Coding experience

1. Do you have any prior experience in reinforcement learning?
 - Yes
 - No
2. If yes, have you received formal training or taken a university course on reinforcement learning?
 - Yes
 - No
 - N/A
3. Which tool do you prefer for finding coding related information for your studies nowadays?
 - ChatGPT
 - Google search engine
4. What will be the impact of relying too much on ChatGPT on your critical thinking skills development?
 - Positively impacts the learning process
 - Negatively impacts the learning process
5. Have you ever noticed any wrong information by ChatGPT?
 - Yes
 - No
 - Not sure

Game theory and Reinforcement Learning knowledge test

1. Implement a smart Tic Tac Toe Agent that uses reinforcement learning with an N-step look ahead approach with your chosen programming language and algorithm [93].
2. Consider a Tic Tac Toe game with the below game tree where X is the maximizing player, and O is the minimizing player. Based on the minimax principle, what move should X make? [104]

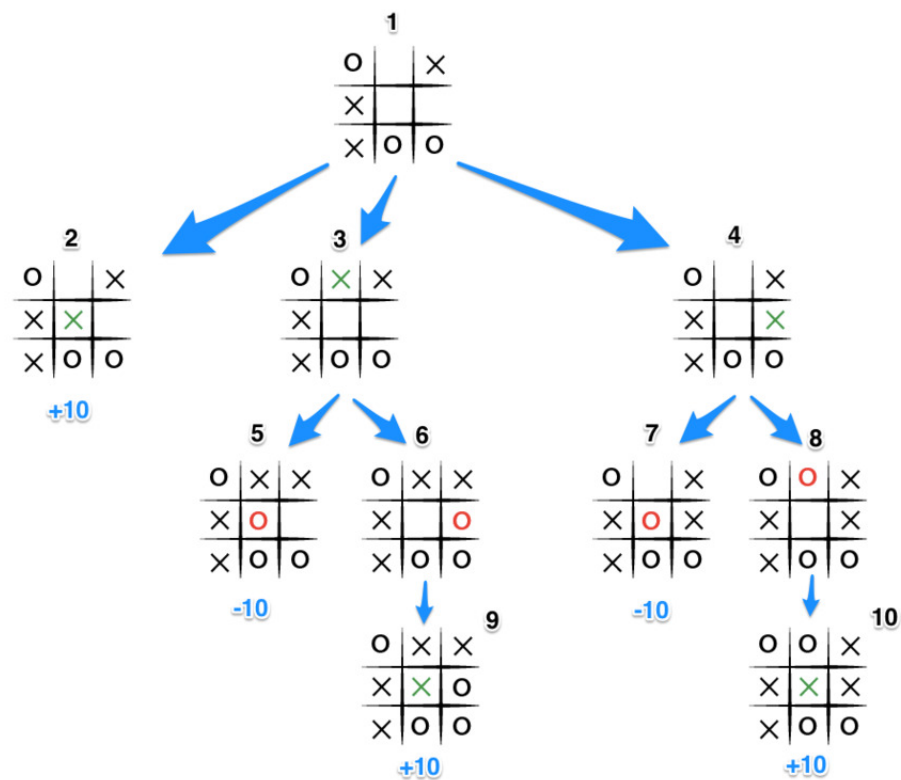


Figure 6.15: Game tree for the Minimax algorithm test [104]

- State 2
- State 3
- State 4
- All remaining moves result in a tie

3. Given below the reward function for a tic tac toe smart agent with p1 as player 1 and p2 as player 2. Would it be possible to change the tie reward to benefit p2 more? Provide values for variables A and B [105].

```
def giveReward(self):
    result = self.winner()
    # backpropagate reward
    if result == 1:
        self.p1.feedReward(1)
        self.p2.feedReward(0)
    elif result == -1:
        self.p1.feedReward(0)
        self.p2.feedReward(1)
    else:
        self.p1.feedReward(A)
        self.p2.feedReward(B)
```

- A= (0.1), B= (0.5)
- A= (0.5), B= (0.1)
- A= (0), B= (0)

4. In your Tic Tac Toe code it appears that the AI agent repeatedly chooses the same option, even though there are other options available. Upon examining the code, you discover that the play() method fails to update the AI agent's Q-values correctly. What is the most likely cause of the bug, given the scenario and the code highlighted in red below? [105]

```
def play():
    board = Board()
    players = [Player("X"), Player("O")]
    index = random.choice([0, 1])
    activePlayer = players[index]
```

```
while not board.is_game_over():
    availablePositions = board.availablePositions()
    action = activePlayer.chooseAction(availablePositions,board.board)
    board.place(activePlayer.symbol,action)
    activePlayer.updateQvalues(board.getState(),action,board.getReward(activePlayer)
        )
    index = (index + 1) % 2
    activePlayer = players[index]

return board.getWinner()

reward = play()
print("Player" + reward + "wins the game")
```

- The Board() class is not implemented correctly.
 - The Agent() class is missing.
 - There is an indentation error in the play() method.
5. Refer to the Alpha-Beta pruning algorithm and prune the game tree below, assuming child nodes are visited from left to right. The final alpha and beta values calculated at each internal node visited are also provided [106].

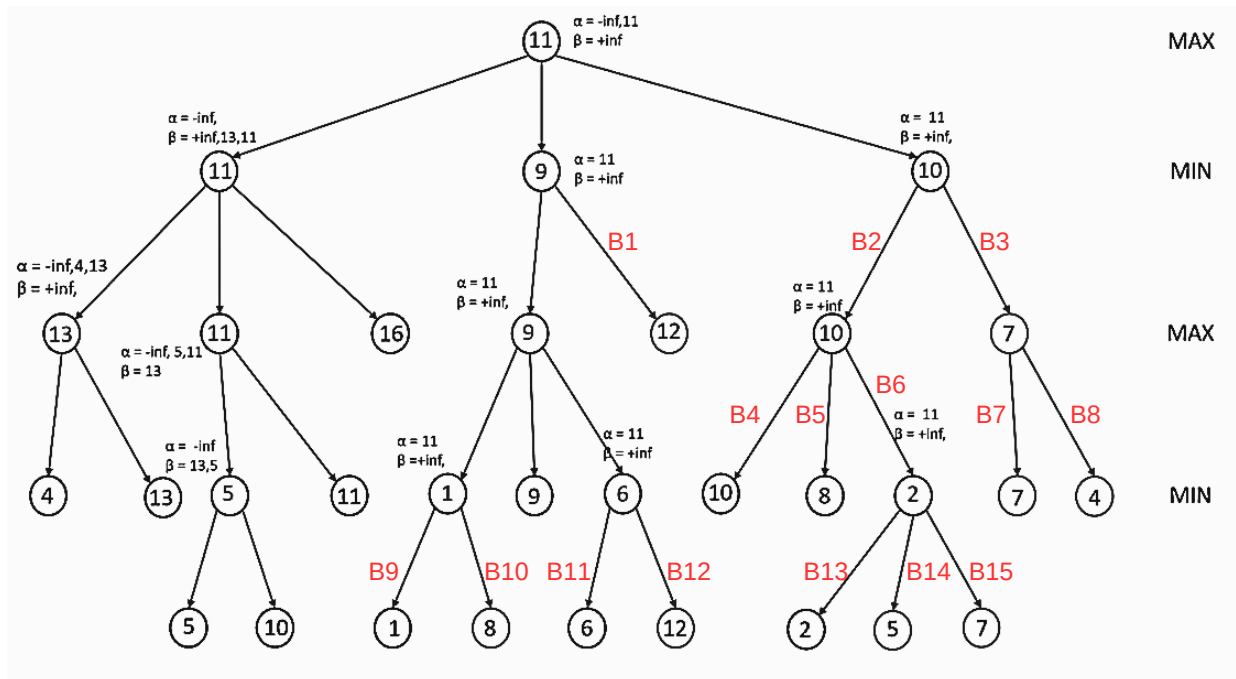


Figure 6.16: Game tree for the Alpha-Beta pruning algorithm test [106]

- B1, B3, B7, B8, B10, B12, B14, B15
- B1, B3, B7, B8, B9, B11, B13
- B4, B5, B7, B8

Feedback

1. What was your level of effort? [In completing the MOOC course]

- Very Easy
- Easy
- Neutral
- Difficult
- Very Difficult

2. What was your level of effort? [In completing the RL knowledge test]

- Very Easy
- Easy
- Neutral
- Difficult
- Very Difficult

3. What was your level of confidence? [In your ability to program a RL agent]

- Not at all confident
- Slightly confident
- Neutral
- Somewhat confident
- Very confident

4. What was your level of confidence? [In explaining RL to someone else]

- Not at all confident
- Slightly confident
- Neutral
- Somewhat confident
- Very confident

5. What was your level of confidence? [To critically evaluate the information provided by the ChatGPT while coding]

- Not at all confident
- Slightly confident
- Neutral
- Somewhat confident
- Very confident

6. Have you noticed any improvements in your learning experience since incorporating AI-assisted learning?
 - Yes
 - No
7. Do you feel that AI-assisted learning enhances or negatively impacts your ability to self-direct your MOOC journey?
 - Positively impacts
 - Negatively impacts
8. Did you encounter any challenges in your MOOC and coding journey? If so, please describe them.
9. What is the main reason for you to consider using ChatGPT for coding/learning?
 - To save time and increase efficiency
 - To improve the accuracy of data processing results
 - To keep up with the latest advancements in technology
 - To reduce the need for manual data processing
 - Other:

6.4 Student Questionnaire: non-AI Context

Coding experience

1. Do you have any prior experience in reinforcement learning?
 - Yes
 - No
2. If yes, have you received formal training or taken a university course on reinforcement learning?

- Yes
 - No
 - N/A
3. Which tool do you prefer for finding coding related information for your studies nowadays?
- ChatGPT
 - Google search engine
4. What will be the impact of relying too much on ChatGPT on your critical thinking skills development?
- Positively impacts the learning process
 - Negatively impacts the learning process
5. Have you ever noticed any wrong information by Google search results?
- Yes
 - No
 - Not sure

Game theory and Reinforcement Learning knowledge test

1. Implement a smart Tic Tac Toe Agent that uses reinforcement learning with an N-step look ahead approach with your chosen programming language and algorithm [93].
2. Consider a Tic Tac Toe game with the below game tree where X is the maximizing player, and O is the minimizing player. Based on the minimax principle, what move should X make? [104]

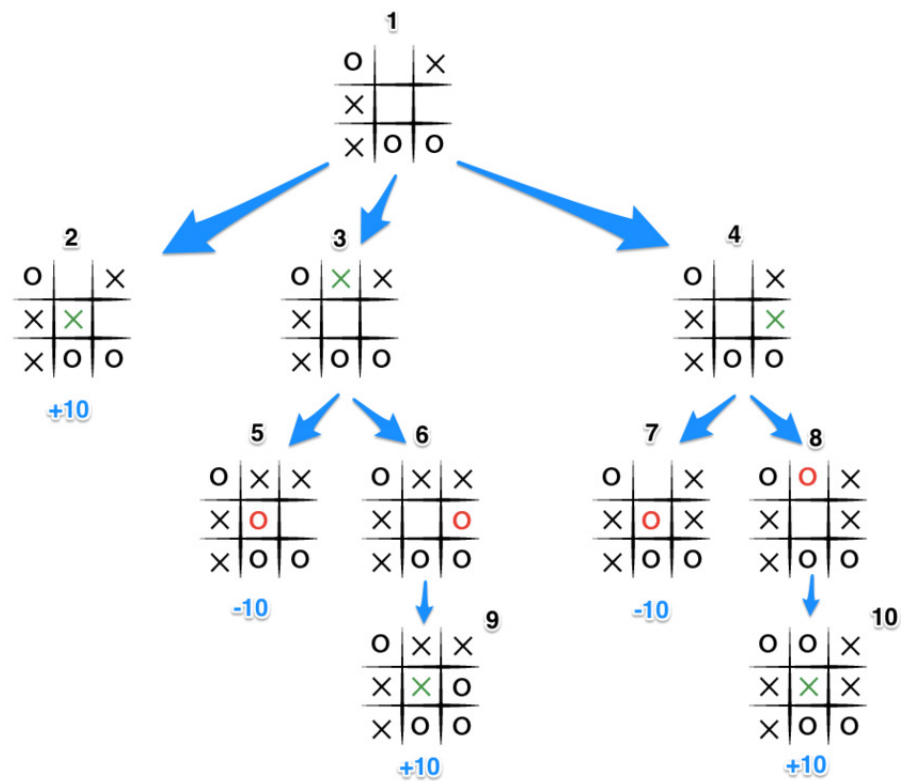


Figure 6.17: Game tree for the Minimax algorithm test [104]

- State 2
 - State 3
 - State 4
 - All remaining moves result in a tie
3. Given below the reward function for a tic tac toe smart agent with p1 as player 1 and p2 as player 2. Would it be possible to change the tie reward to benefit p2 more? Provide values for variables A and B [105].

```
def giveReward(self):
    result = self.winner()
    # backpropagate reward
    if result == 1:
        self.p1.feedReward(1)
```

```

        self.p2.feedReward(0)
    elif result == -1:
        self.p1.feedReward(0)
        self.p2.feedReward(1)
    else:
        self.p1.feedReward(A)
        self.p2.feedReward(B)

```

- A= (0.1), B= (0.5)
- A= (0.5), B= (0.1)
- A= (0), B= (0)

4. In your Tic Tac Toe code it appears that the AI agent repeatedly chooses the same option, even though there are other options available. Upon examining the code, you discover that the `play()` method fails to update the AI agent's Q-values correctly. What is the most likely cause of the bug, given the scenario and the code highlighted in red below? [105]

```

def play():
    board = Board()
    players = [Player("X"), Player("O")]
    index = random.choice([0, 1])
    activePlayer = players[index]

    while not board.is_game_over():
        availablePositions = board.availablePositions()
        action = activePlayer.chooseAction(availablePositions,board.board)
        board.place(activePlayer.symbol,action)
        activePlayer.updateQvalues(board.getState(),action,board.getReward(activePlayer)
        )
        index = (index + 1) % 2
        activePlayer = players[index]

```

```
return board.getWinner()
```

```
reward = play()
```

```
print("Player" + reward + "wins the game")
```

- The Board() class is not implemented correctly.
 - The Agent() class is missing.
 - There is an indentation error in the play() method.
5. Refer to the Alpha-Beta pruning algorithm and prune the game tree below, assuming child nodes are visited from left to right. The final alpha and beta values calculated at each internal node visited are also provided [106].

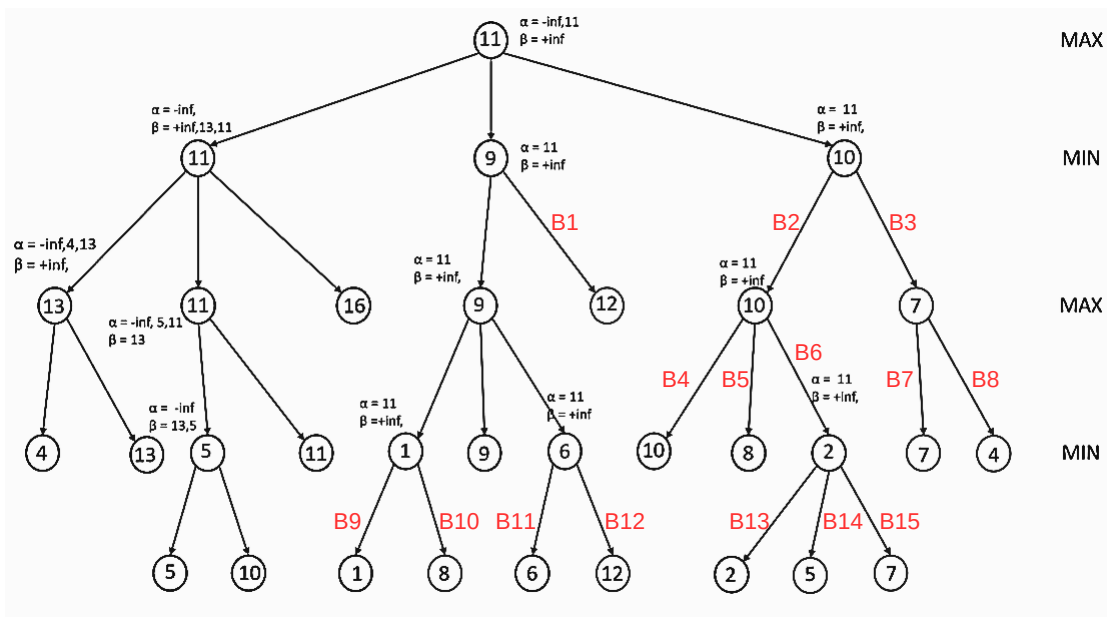


Figure 6.18: Game tree for the Alpha-Beta pruning algorithm test [106]

- B1, B3, B7, B8, B10, B12, B14, B15
- B1, B3, B7, B8, B9, B11, B13
- B4, B5, B7, B8

Feedback

1. What was your level of effort? [In completing the MOOC course]
 - Very Easy
 - Easy
 - Neutral
 - Difficult
 - Very Difficult

2. What was your level of effort? [In completing the RL knowledge test]
 - Very Easy
 - Easy
 - Neutral
 - Difficult
 - Very Difficult

3. What was your level of confidence? [In your ability to program a RL agent]
 - Not at all confident
 - Slightly confident
 - Neutral
 - Somewhat confident
 - Very confident

4. What was your level of confidence? [In explaining RL to someone else]
 - Not at all confident
 - Slightly confident
 - Neutral
 - Somewhat confident

- Very confident
5. What was your level of confidence? [In manually referring to Google search while coding]
- Not at all confident
 - Slightly confident
 - Neutral
 - Somewhat confident
 - Very confident
6. Could AI assistance have made your self-learning journey easier?
- Yes
 - No
7. Do you feel that manually referring to google enhances or negatively impacts your ability to self-direct your MOOC journey?
- Positively impacts
 - Negatively impacts
8. Did you encounter any challenges in your MOOC and coding journey? If so, please describe them.
9. What is the main reason for you to consider using Google search for coding/learning?
- To save time and increase efficiency
 - To improve the accuracy of data processing results
 - To keep up with the latest advancements in technology
 - To reduce the need for manual data processing
 - Other:

6.5 Transcript: Participant and ChatGPT

The transcript below shows Participant 25’s interaction with ChatGPT (GPT-4 model) during reinforcement learning coding tasks [107].

Participant 25 prompt: Hello GPT-4! I would like to conduct a survey on you titled “Assessing the Impact of AI-Assisted Learning on Self-Directed Massive Open Online Courses”. It aims to assess the potential information bias that could arise from utilizing AI-assisted learning in student’s self-directed learning journeys through Massive Open Online Courses (MOOCs). Are you ready for the questions?

GPT-4 response: *Hello! As an AI language model, I’m always ready to help with your questions. However, please note that I am not a human with personal experiences or opinions, but rather a tool that can provide information and insights based on the data I’ve been trained on. If you still want to proceed with the survey, I’ll do my best to provide relevant and useful answers.*

Participant 25 prompt: Sounds great, let’s proceed! Question 1: Implement a smart Tic Tac Toe Agent that uses reinforcement learning with an N-step look ahead approach with your chosen programming language and algorithm.

GPT-4 response: *As an AI language model, I cannot directly implement code for you. However, I can guide you through the process of creating a Tic Tac Toe agent that uses reinforcement learning with an N-step look-ahead approach. I’ll provide you with an outline using Python and Q-learning as an example.*

Initialize the Tic Tac Toe environment.

Define the Q-learning algorithm with an N-step look-ahead.

Train the agent using the Q-learning algorithm.

Test the agent’s performance against a random player or another trained agent.

Here’s a rough outline of the code structure in Python:

python

Copy code

```
import numpy as np
```

```
class TicTacToe:
    def __init__(self):
        # Initialize game state and other variables

    def reset(self):
        # Reset game state

    def get_available_actions(self):
        # Get available actions (empty cells) from the current state

    def step(self, action, player):
        # Apply the action (place a marker) and return the new state, reward, and whether the game is
        over

    def is_winner(self, player):
        # Check if a player has won the game

    def is_draw(self):
        # Check if the game is a draw

class QLearningAgent:
    def __init__(self, alpha, gamma, epsilon, n_look_ahead):
        # Initialize Q-table, learning rate (alpha), discount factor (gamma), exploration rate (epsilon)
        , and look-ahead steps (n_look_ahead)

    def choose_action(self, state, available_actions):
        # Choose an action based on the Q-table and exploration rate

    def learn(self, state, action, reward, next_state, done):
        # Update the Q-table based on the learning rate, discount factor, and N-step look-ahead

def train(agent, env, episodes):
```

Train the agent through the specified number of episodes

```
def test(agent, env, episodes):
```

Test the agent's performance against a random player or another trained agent

This outline should serve as a starting point for implementing the Tic Tac Toe agent. You can adjust the parameters and algorithms as needed for your specific project requirements.

Participant 25 prompt: This is in response to a quiz I would like to conduct on you to measure your accuracy. Can you please fill in the code?

GPT-4 response: *Sure, here's a more detailed version of the Tic Tac Toe agent using Q-learning and an N-step look-ahead approach. Note that this is just one way to implement such an agent, and you might want to tweak the parameters and algorithm for better performance.*

python

Copy code

```
import numpy as np
```

```
import random
```

```
class TicTacToe:
```

```
    def __init__(self):
```

```
        self.board = np.zeros((3, 3), dtype=int)
```

```
        self.player = 1
```

```
    def reset(self):
```

```
        self.board = np.zeros((3, 3), dtype=int)
```

```
        self.player = 1
```

```
    def get_available_actions(self):
```

```
        return list(zip(*np.where(self.board == 0)))
```

```
    def step(self, action, player):
```

```
        self.board[action] = player
```



```

done = self.is_winner(player) or self.is_draw()
reward = 1 if self.is_winner(player) else 0.5 if self.is_draw() else 0
return self.board.copy(), reward, done

```

```

def is_winner(self, player):
    for row in range(3):
        if np.all(self.board[row, :] == player) or np.all(self.board[:, row] == player):
            return True
    if np.all(np.diag(self.board) == player) or np.all(np.diag(np.fliplr(self.board)) == player):
        return True
    return False

```

```

def is_draw(self):
    return not np.any(self.board == 0)

```

```

class QLearningAgent:
    def __init__(self, alpha=0.1, gamma=0.9, epsilon=0.1, n_look_ahead=1):
        self.Q = {}
        self.alpha = alpha
        self.gamma = gamma
        self.epsilon = epsilon
        self.n_look_ahead = n_look_ahead

    def get_q_value(self, state, action):
        return self.Q.get((state.tostring(), action), 0)

    def set_q_value(self, state, action, value):
        self.Q[(state.tostring(), action)] = value

    def choose_action(self, state, available_actions):
        if random.uniform(0, 1) < self.epsilon:
            return random.choice(available_actions)

```

```

q_values = [self.get_q_value(state, action) for action in available_actions]
return available_actions[np.argmax(q_values)]

def learn(self, state, action, reward, next_state, done):
    q_value = self.get_q_value(state, action)
    if done:
        target = reward
    else:
        next_available_actions = [a for a in self.get_available_actions(next_state) if a != action]
        lookahead_rewards = [self.get_q_value(next_state, next_action) for next_action in
            next_available_actions]
        target = reward + self.gamma * np.sum(lookahead_rewards[:self.n_look_ahead]) / self.
            n_look_ahead

    self.set_q_value(state, action, q_value + self.alpha * (target - q_value))

def train(agent, env, episodes):
    for episode in range(episodes):
        env.reset()
        state = env.board.copy()
        done = False
        while not done:
            action = agent.choose_action(state, env.get_available_actions())
            next_state, reward, done = env.step(action, env.player)
            agent.learn(state, action, reward, next_state, done)
            state = next_state.copy()
            env.player = 3 - env.player

def test(agent, env, episodes):
    wins = 0
    draws = 0
    for episode in range(episodes):

```

```

env.reset()
state = env.board.copy()
done = False
while not done

```

Participant 25 prompt: Question 2: Given below the reward function for a tic tac toe smart agent with p1 as player 1 and p2 as player 2. Would it be possible to change the tie reward to benefit p2 more? Provide values for variables A and B.

```

def giveReward(self):
    result = self.winner()
    # backpropagate reward
    if result == 1:
        self.p1.feedReward(1)
        self.p2.feedReward(0)
    elif result == -1:
        self.p1.feedReward(0)
        self.p2.feedReward(1)
    else:
        self.p1.feedReward(A)
        self.p2.feedReward(B)

```

The possible answers are:

A= (0.1), B= (0.5)

A= (0.5), B= (0.1)

A= (0), B= (0)

GPT-4 response: *Yes, you can change the tie reward to benefit player 2 more by assigning a higher reward value to player 2 (p2) compared to player 1 (p1) in case of a tie. Based on the given answer choices, you can set A = 0.1 and B = 0.5, which will give a higher reward to player 2 in case of a tie. The modified reward function would look like this:*

python

Copy code

```

def giveReward(self):

```

```

result = self.winner()
# backpropagate reward
if result == 1:
    self.p1.feedReward(1)
    self.p2.feedReward(0)
elif result == -1:
    self.p1.feedReward(0)
    self.p2.feedReward(1)
else:
    self.p1.feedReward(0.1) # A = 0.1
    self.p2.feedReward(0.5) # B = 0.5

```

This change in the reward function would encourage player 2 to prioritize achieving a tie over losing, while player 1 would still prefer a win over a tie.

Participant 25 prompt: Question 3: Consider a Tic Tac Toe game with the below game tree where X is the maximizing player, and O is the minimizing player. Based on the minimax principle, what move should X make?

The game tree is represented as a dictionary where each key is a state in the game. Each key contains 3x3 array representing the current Tic Tac Toe table (the array's rows start from top to bottom) where 1 = X, 0 = O, and -1 = empty, as well as an array of possible states (which correspond to keys in the game tree dictionary) that you can go to from it- if the array is empty, you have reached the end of the game:

```

game_tree =
{
1: { "table": [[0, -1, 1], [1, -1, -1], [1, 0, 0]], "possible_states": [2, 3, 4] },
2: { "table": [[0, -1, 1], [1, 1, -1], [1, 0, 0]], "possible_states": [] },
3: { "table": [[0, 1, 1], [1, -1, -1], [1, 0, 0]], "possible_states": [5, 6] },
4: { "table": [[0, -1, 1], [1, -1, 1], [1, 0, 0]], "possible_states": [7, 8] },
5: { "table": [[0, 1, 1], [1, 0, -1], [1, 0, 0]], "possible_states": [] },
6: { "table": [[0, 1, 1], [1, -1, 0], [1, 0, 0]], "possible_states": [9] },
7: { "table": [[0, -1, 1], [1, 0, 1], [1, 0, 0]], "possible_states": [] },
8: { "table": [[0, 0, 1], [1, -1, 1], [1, 0, 0]], "possible_states": [10] },

```

```

9: { "table": [[0, 1, 1], [1, 1, 0], [1, 0, 0]], "possible_states": [] },
10: { "table": [[0, 0, 1], [1, 1, 1], [1, 0, 0]], "possible_states": [] }
}

```

Consider the following rewards for each of the end game states: 2: reward of 10, 5: reward of -10, 7: reward of -10, 9: reward of 10, and 10: reward of 10

Here are the possible answers given that you are starting at state 1:

State 2

State 3

State 4

All remaining moves result in a tie

GPT-4 response: *Based on the minimax principle and the rewards given, let's calculate the expected values for each possible state from state 1:*

State 2 (reward of 10) is an end game state, so the value is 10.

State 3 has possible states 5 and 6. State 5 has a reward of -10, and state 6 leads to state 9, which has a reward of 10. Since O is the minimizing player, it will choose the minimum value between these two states, which is -10. So, the value for state 3 is -10.

State 4 has possible states 7 and 8. State 7 has a reward of -10, and state 8 leads to state 10, which has a reward of 10. Since O is the minimizing player, it will choose the minimum value between these two states, which is -10. So, the value for state 4 is -10.

Now, we need to decide which move X should make. X is the maximizing player and will choose the move with the highest value. The values for states 2, 3, and 4 are 10, -10, and -10, respectively.

Based on the minimax principle, X should choose the move leading to State 2, as it has the highest value (10).

Participant 25 prompt: Question 4: In your Tic Tac Toe code it appears that the AI agent repeatedly chooses the same option, even though there are other options available. Upon examining the code, you discover that the play() method fails to update the AI agent's Q-values correctly. What is the most likely cause of the bug, given the scenario and the code highlighted in red below?

```

def play():
    board = Board()
    players = [Player("X"), Player("O")]
    index = random.choice([0, 1])
    activePlayer = players[index]

    while not board.is_game_over():
        availablePositions = board.availablePositions()
        action = activePlayer.chooseAction(availablePositions, board.board)
        board.place(activePlayer.symbol, action)
        activePlayer.updateQvalues(board.getState(), action, board.getReward(activePlayer))
        index = (index + 1) % 2
        activePlayer = players[index]

    return board.getWinner()

reward = play()
print("Player" + reward + "wins the game")

```

Here are the possible answers:

The Board() class is not implemented correctly.

The Agent() class is missing.

There is an indentation error in the play() method.

GPT-4 response: *The provided code snippet does not show any syntax errors or indentation issues. However, it seems like the Q-values for the losing player are not being updated. This could be the reason for the AI agent repeatedly selecting the same course of action. To fix the issue, you should update the Q-values for both the winning and losing players. You can do this by adding another line of code after the game is over, to update the Q-values for the losing player. Modify the play() function as follows:*

python

Copy code

```

def play():
    board = Board()
    players = [Player("X"), Player("O")]
    index = random.choice([0, 1])
    activePlayer = players[index]

    while not board.is_game_over():
        availablePositions = board.availablePositions()
        action = activePlayer.chooseAction(availablePositions, board.board)
        board.place(activePlayer.symbol, action)
        activePlayer.updateQvalues(board.getState(), action, board.getReward(activePlayer))
        index = (index + 1) % 2
        activePlayer = players[index]

    winner = board.getWinner()
    if winner is not None:
        loser = players[0] if players[0] != winner else players[1]
        loser.updateQvalues(board.getState(), action, -board.getReward(winner))

    return winner

reward = play()
print("Player" + reward + "wins the game")

```

This should help the AI agent explore different courses of action and not get stuck in a loop of selecting the same moves.

Participant 25 prompt: So what would be your choice from the list of possible answers?

GPT-4 response: *Apologies for not addressing the options provided earlier. Given the list of possible answers and the code snippet, the most likely cause for the bug is:*

“The Agent() class is missing.”

The code provided uses a Player class, which seems to be the agent for the game. However, the question refers to an Agent() class, which is not present in the code snippet. It’s possible

that the *Player* class is missing the necessary logic to update the *Q*-values correctly, as we discussed in the previous response.

Participant 25 prompt: Question 5: Refer to the Alpha-Beta pruning algorithm and prune the game tree below, assuming child nodes are visited from left to right. The final alpha and beta values calculated at each internal node visited are also provided. The game tree will be represented as a dictionary where each key represents a node in the tree and contains an array of children (which correspond to keys in the game tree dictionary) that are ordered from left-most to right-most, node value, alpha value, and beta value- if a node doesn't have a certain value defined, it will be assigned to an empty array:

```
game_tree =
{
  1: { "children": [2, 3, 4] , "value": 11, "alpha": [-inf, 11], "beta": [inf] },
  2: { "children": [5, 6, 7] , "value": 11, "alpha": [-inf], "beta": [inf, 13, 11] },
  3: { "children": [8, 9] , "value": 9, "alpha": [11], "beta": [inf] },
  4: { "children": [10, 11] , "value": 10, "alpha": [11], "beta": [inf] },
  5: { "children": [12, 13] , "value": 13, "alpha": [-inf, 4, 13], "beta": [inf] },
  6: { "children": [14, 15] , "value": 11, "alpha": [-inf, 5, 11], "beta": [13] },
  7: { "children": [] , "value": 16, "alpha": [], "beta": [] },
  8: { "children": [16, 17, 18] , "value": 9, "alpha": [11], "beta": [inf] },
  9: { "children": [] , "value": 12, "alpha": [], "beta": [] },
  10: { "children": [19, 20, 21] , "value": 10, "alpha": [11], "beta": [inf] },
  11: { "children": [22, 23] , "value": 7, "alpha": [], "beta": [] },
  12: { "children": [] , "value": 4, "alpha": [], "beta": [] },
  13: { "children": [] , "value": 13, "alpha": [], "beta": [] },
  14: { "children": [24, 25] , "value": 5, "alpha": [-inf], "beta": [13, 5] },
  15: { "children": [] , "value": 11, "alpha": [], "beta": [] },
  16: { "children": [26, 27] , "value": 1, "alpha": [11], "beta": [inf] },
  17: { "children": [] , "value": 9, "alpha": [], "beta": [] },
  18: { "children": [28, 29] , "value": 6, "alpha": [11], "beta": [inf] },
  19: { "children": [] , "value": 10, "alpha": [], "beta": [] },
  20: { "children": [] , "value": 8, "alpha": [], "beta": [] },
  21: { "children": [30, 31, 32] , "value": 2, "alpha": [11], "beta": [inf] },
```



```

22: { "children": [], "value": 7, "alpha": [], "beta": [] },
23: { "children": [], "value": 4, "alpha": [], "beta": [] },
24: { "children": [], "value": 5, "alpha": [], "beta": [] },
25: { "children": [], "value": 10, "alpha": [], "beta": [] },
26: { "children": [], "value": 1, "alpha": [], "beta": [] },
27: { "children": [], "value": 8, "alpha": [], "beta": [] },
28: { "children": [], "value": 6, "alpha": [], "beta": [] },
29: { "children": [], "value": 12, "alpha": [], "beta": [] },
30: { "children": [], "value": 2, "alpha": [], "beta": [] },
31: { "children": [], "value": 5, "alpha": [], "beta": [] },
32: { "children": [], "value": 7, "alpha": [], "beta": [] }
}

```

The possible branches you can prune are represented as a dictionary where each key contains a 1x2 array where the first value is the top-most node and the second value is the bottom-most node that correspond to keys in the game tree dictionary:

branches =

```

{
  "B1": [3, 9],
  "B2": [4, 10],
  "B3": [4, 11],
  "B4": [10, 19],
  "B5": [10, 20],
  "B6": [10, 21],
  "B7": [11, 22],
  "B8": [11, 23],
  "B9": [16, 26],
  "B10": [16, 27],
  "B11": [18, 28],
  "B12": [18, 29],
  "B13": [21, 30],
  "B14": [21, 31],
  "B15": [21, 32]
}

```

}

GPT-4 response: *Based on the Alpha-Beta pruning algorithm and the provided game tree, the branches that would be pruned are:*

B1, B3, B7, B8, B10, B12, B14, B15

Here's the explanation of the pruned branches:

B1: Pruned because after visiting node 8, $\alpha = 11$ and $\beta = 9$ at node 3. Since $\alpha \geq \beta$, we can prune the branch from node 3 to node 9.

B3: Pruned because after visiting node 10, $\alpha = 11$ and $\beta = 10$ at node 4. Since $\alpha \geq \beta$, we can prune the branch from node 4 to node 11.

B7: Pruned because after visiting node 22, $\alpha = 11$ and $\beta = 7$ at node 11. Since $\alpha \geq \beta$, we can prune the branch from node 11 to node 23.

B8: Pruned because after visiting node 24, $\alpha = 11$ and $\beta = 5$ at node 14. Since $\alpha \geq \beta$, we can prune the branch from node 14 to node 25.

B10: Pruned because after visiting node 26, $\alpha = 11$ and $\beta = 1$ at node 16. Since $\alpha \geq \beta$, we can prune the branch from node 16 to node 27.

B12: Pruned because after visiting node 28, $\alpha = 11$ and $\beta = 6$ at node 18. Since $\alpha \geq \beta$, we can prune the branch from node 18 to node 29.

B14: Pruned because after visiting node 30, $\alpha = 11$ and $\beta = 2$ at node 21. Since $\alpha \geq \beta$, we can prune the branch from node 21 to node 31.

B15: Pruned because after visiting node 31, $\alpha = 11$ and $\beta = 5$ at node 21. Since $\alpha \geq \beta$, we can prune the branch from node 21 to node 32.