



FIE 2024

WASHINGTON, D.C., USA

OCTOBER 13-16, 2024

Integrating Generative AI with Data Structures and Algorithm Analysis Course Homework

Cong Pu

Department of Computer Science

Oklahoma State University

Stillwater, OK, United States

Outline

- Introduction
- DS2 Course at Oklahoma State University
- Redesigned Course Homework
- Evaluation Survey
- Lessons Learned
- Conclusion



Introduction

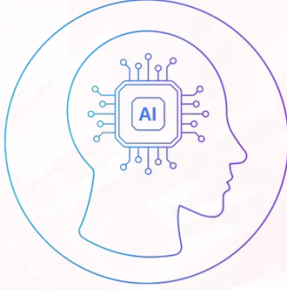
Learning CS2 concepts is **challenging** due to

- Abstract nature of concepts
- Conceptual misunderstanding
- The use of inappropriate data structures and algorithms

- Data Structures and Algorithm Analysis (**CS2**)
 - Fundamental and important knowledge and skills
 - Computing-related degrees (e.g., computer science, ..., etc.)
 - **Rationale:** building block for
 - improving problem solving skills
 - realizing the importance of resource and performance optimization
 - elevating code quality and efficiency
 - facilitating program scalability and flexibility
 - developing robust and reliable software



Introduction (cont.)

- Generative artificial intelligence (AI)  is no longer a fad/hype
 - Become a subject of concern and interest for different sectors
 - Industry and government embracing the AI revolution
 - Higher education institutions facing “disruption”
 - Provided sample statements regarding the use of generative AI in the classroom:
 - Do not allow
 - Allow in some circumstances or with explicit permission
 - Allow full use in the learning environment



Introduction (cont.)

- Generative AI is totally different
 - Unlike electronic devices or traditional online resources (e.g., Wikipedia) which can be prohibited/avoided/detected.
 - College students and faculty have already been using generative AI which are embedded in other application programs
 - Grammarly, Google Docs, Microsoft Word, etc.
 - ACM permits authors to use generative AI to produce manuscript content as long as they acknowledge its usage prominently

If there is no proper way to ban generative AI in the classroom, we might need to think about how to use them in the careers students are preparing to enter.



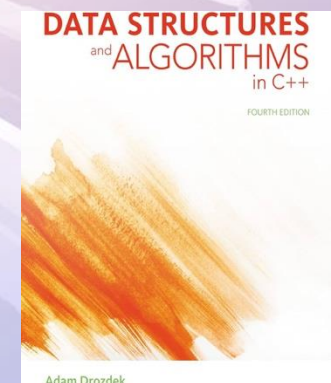
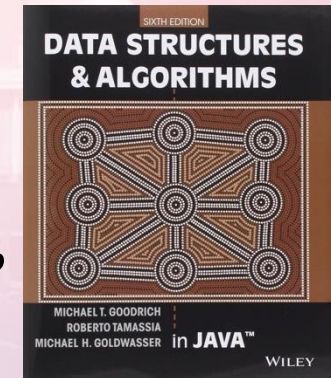
This Paper Summarizes



- How we redesigned course homework
 - Provide students with opportunities to use generative AI to support their learning in the CS2 course
 - Measure the effectiveness of utilizing generative AI to improve student learning outcomes in the CS2 course

CS2 Course at Oklahoma State University

- The CS2 course covers *complexity analysis of algorithms, linked list, recursion, stacks, queues, trees, priority queues, heaps, hash tables, graphs, as well as searching and sorting algorithms.*
- Recommended textbooks (not required)
 - Data Structures and Algorithms in Java, M.T. Goodrich, R. Tamassia and M. H. Goldwasser, 6th Edition, Wiley
 - Data Structures and Algorithms in C++, A. Drozdek, 4th Edition, Cengage Learning



CS2 Course at Oklahoma State University

- Learning outcomes:

1. Understand basic data structures and abstract data types including stacks, queues, lists, sets, maps and graphs.
2. Use recursion as a powerful problem solving technique in design and development of data structures and understand when it is not appropriate to use.
3. Gain an appreciation of the variety, theoretical nature, and practical uses of data structures.
4. Analyze the efficiency of data structures and select the most appropriate data structure for applications.
5. Build data structures and use them as building blocks to form more complex and advanced data structures in a hierarchical manner.



Course Homework Prior Redesign



- Six homework throughout the entire semester:
 - Homework #1: algorithm complexity analysis
 - Homework #2: singly and doubly linked lists and their operations.
 - Homework #3: stacks and queues.
 - Homework #4: recursion.
 - Homework #5: binary tree and various tree traversal algorithms.
 - Homework #6: graphs and searching and sorting algorithms.
- The instructions and requirements of each course homework were straightforward and prescriptive to students.
 - they knew explicitly what data structures and algorithms were required to complete the course homework.



Course Homework Prior Redesign (cont.)

- Potential “issue” of course homework prior redesign:

The course homework did not include any other coding related activities, students only got implementation practice on the pre-determined data structures and algorithms, and other essential programming skills could not be improved.



Redesign Course Homework

- The redesigned course homework was composed of three components:
 - 80% data structures and algorithms implementation
 - 10% program testing
 - 10% exploiting generative AI
- Each homework
 - 2 weeks to complete
 - the required knowledge, skills, and abilities are introduced before releasing homework



Redesign Course Homework #2

- The objective of homework #2 is to help students understand and master the concepts and implementation of linked lists and their operations
- Homework Description – Part One:

Description: Part One: Write a program that can insert, delete, search, and print nodes using singly linked lists and doubly linked lists. Your program should be menu driven and execute the chosen operation. If you type 12, then exit the program. Here, IH (Insert Head), IT (Insert Tail), DH (Delete Head), DT (Delete Tail), SD (Search & Delete), PS (Print Single Linked List), and PD (Print Double Linked List). Display an “error” message when a node to be searched or deleted does not exist. Or try to delete a node in the empty list.



Redesign Course Homework #2 (cont.)

- The objective of homework #2 is to help students understand and master the concepts and implementation of linked lists and their operations
- Homework Description – Part One:

M E N U

SLL: IH(0), IT(1), DH(2), DT(3), SD(4), PS(5)

DLL: IH(6), IT(7), DH(8), DT(9), SD(10), PD(11)

Exit Program (12)



Redesign Course Homework #2 (cont.)

- The objective of homework #2 is to help students understand and master the concepts and implementation of linked lists and their operations
- Homework Description – Part Two:

*Use a generative AI, such as ChatGPT, Bing AI, or Bard, to **debug one programming problem** (e.g., error message from compiler, unexpected behavior, etc.) you encounter while programming, obtain a valid response from generative AI, and fix the problem accordingly.*



Redesign Course Homework

- Generative AI component in course homework:
 - Homework #1: Use a generative AI, such as ChatGPT, Bing AI, or Bard, to **generate a “solution”** for the program in the above question. Compare your solution with the “solution” generated by generative AI, and summarize your findings.
 - Homework #3: Use a generative AI, such as ChatGPT, Bing AI, or Bard, to **generate a stack implemented by a linked list**. Compare your implementation with the “program” generated by generative AI and summarize your findings.



Redesign Course Homework

- Generative AI component in course homework:
 - Homework #4: Use a generative AI, such as ChatGPT, Bing AI, or Bard, to **optimize program, function, or code segment**. Summarize your findings.
 - Homework #5: Use a generative AI, such as ChatGPT, Bing AI, or Bard, to **generate the idea of creating binary tree, Breadth-First Traversal, and Depth-First Traversal** (pre-order, in-order, and post-order), verify/ validate the correctness of ideas, and summarize your findings.



Evaluation Survey

- The perception survey was organized and coordinated by Oklahoma State University Institute for Teaching and Learning Excellence (ITLE).
 - Students were provided 30 minutes of class time to complete the survey without feeling rushed.
 - Response rate: **77%** (47 out of 61 students submitted the perception survey)
 - Six (6) questions: five (5) likert scale questions and one (1) open-ended question.
 - Open-ended question:

If you were the instructor, what changes would you like to make for the usage of generative AI to improve programming skills as well as design and implement efficient algorithms and data structures in this class?



Evaluation Survey (cont.)

- The likert scale questions and their corresponding responses are shown in Table

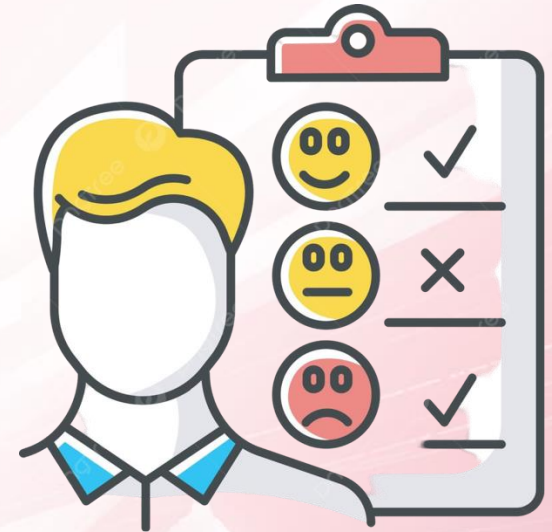


TABLE I
STUDENT PERCEPTION SURVEY QUESTIONS AND RESPONSES.

Question: The generative AI tools helped me	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Total
understand testing and debugging better.	2	0	5	23	17	47
improve coding skills and code quality.	2	3	6	19	17	47
design and implement efficient algorithms and data structures.	2	1	9	23	12	47
select appropriate algorithms and data structures.	1	3	13	21	9	47
understand the importance of designing and implementing efficient algorithms and data structures.	3	2	9	22	11	47

Lessons Learned

- The lessons learned after adopting the redesigned course homework with the integration of generative AI



Communicate expectation of the use of generative AI to students



Help students understand the rules round generative AI in the classroom



Use generative AI to apply CS2 concepts rather than simply recall definitions

Conclusion

- In Fall 2023 semester, we redesigned all course homework in the CS2 course at Oklahoma State University.
- The redesigned course homework were integrated with generative AI so that students had opportunities to use this emerging advanced technology to support and improve learning outcomes in the CS2 course.
- Based on the survey responses, a majority of respondents perceived that generative AI helped them understand fundamental and important knowledge and skills of CS2.





Thank you!

