Artificial Intelligence and its strengths in coding.

Morgan Huntley

CS131

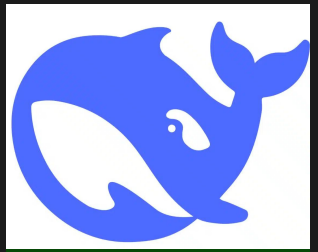
Each AI Used in Testing

ChatGPT
OpenAI

Claude
Anthropic

Gemini
Google

Copilot
Microsoft



DeepSeek
High-Flyer

Coding Tests

Test One: Generating Code

In this test, each AI will be given a prompt to generate code for.

The goal of this test is to see how the AI follows guidelines in generating code. The generated code will then be tested to ensure that it followed instructions, and how well it followed them.

Test Two: Generation Speed

In this test, each AI will be given a complex coding problem.

This will test the speed and accuracy of each AI. They will each have to calculate the result of the code provided. This will test their ability to understand and process the code that they are given. The results will be checked for accuracy.

Test Three: Restricted Coding

In this test, each AI will be given a restricted prompt to generate code for.

The goal of this test is to see how the AI follows the guidelines in generating code, given restrictions on what it can use. This tests how the AI can follow instructions while also following restrictions. The generated code will then be tested to ensure that it has followed instructions, and how well it followed them.

Test One Prompt

You are tasked with creating a Library Management System in C++. The system should manage a collection of books and members, and allow basic operations such as adding books, issuing books to members, returning books, and viewing system status. The system should also store all book names, even if the book is no longer available.

Test One : Result One

```
--- Library Menu ---
1. Add Book
2. Register Member
3. Issue Book
4. Return Book
5. View Status
6. Exit
Choose an option:
```

ChatGPT

Introduce your persona, explaining who they are and where they come from. Mention age and profession.

Pros

- Good user interface
- Quick
- All parameters were followed

Cons

- Generated code missing default constructors.
- Code has infinite loops

Result

The code generated was overall good, but it did need fixing in order for it to work. The code was designed efficiently but unresolved loops could cause memory leaks or information loss if used.

Logo

Test One : Result Two

```
================== LIBRARY MANAGEMENT SYSTEM ==================
1. Add Book
2. Add Member
3. Issue Book
4. Return Book
5. Display All Books
6. Display Available Books
7. Display All Members
8. Search Book by Title
9. Display All Book Titles Ever Added
10. Display System Status
11. Exit
================================================================
Enter your choice:
```

Claude

Pros

- Great interface
- Good error checking
- Met all required criteria.
- Good sample data
- Code ran perfectly without needed any edits.

Cons

- Took over a minute to generate code
- Added aspects that were unasked for.

Result

Claude generated a great program for the given prompt. It created more than asked for which could be seen as both a positive and a negative, but with no errors in the generated code and smooth interface Claude did a good job at following the provided prompt.

Test One : Result Three

Pros

- Code generated quickly
- Comments throughout the code to explain what each part is for.

Cons

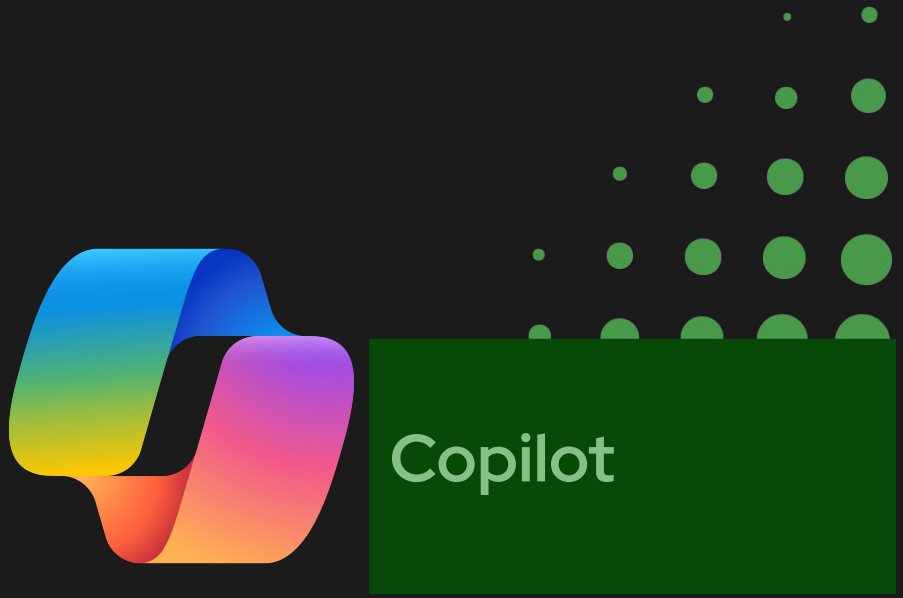
- Initial code had numerous logic and syntax errors preventing it from running
- Brackets were not closed properly
- Certain errors produced infinite loops
- Member declarations were declared incorrectly
- 11 Initial syntax errors.

Result

Unfortunately, there were many errors in the generated code. Most of the functions were unusable, and certain aspects of the code would need to be completely remade to fix certain errors.

Logo

Test One : Result Four



Pros

- Tested errors well
- Generated good tests in main

Cons

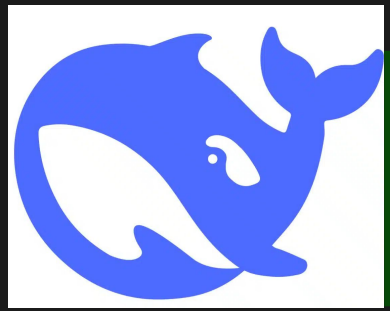
- Code generated unusable
- Code generated missing <algorithm>
- Didn't add an interactive menu.

Result

Unfortunately, the code generated by Copilot was unusable unless almost entirely re-written. Numerous errors were present when testing the code, and unresolved issues persisted preventing the code from being run.

Test One : Result Five

```
Library Management System
1. Add Book
2. Add Member
3. Issue Book
4. Return Book
5. Display Status
6. Exit
Enter your choice:
```



Deepseek

Pros

- Good user interface
- Data stored properly, and easy to access
- Followed instructions properly

Cons

- Infinite loop if invalid choice in menu.
- Took almost 5 minutes to generate code.

Result

Deepseek did a good job following the prompt with its code generation and created a very simple but effective interface. Though there was an infinite loop bug in the menu, the data storage and outputs were all accurate and it did a good job with its code generation.

Test Two

For test two, I provided each AI with code containing a complex mathematical function.
f(x)=e^−x^2 · sin(x)+ln(x+2)

The AI is tasked with running the code containing this function and presenting the result as timely as possible. The AI is timed when it computes this, and will be compared with others based on time spent and accuracy.

Test Two Results

	Time Spent	Result (7.6595130658 is correct answer)	Thoughts
ChatGPT	18.4s	6.5242005085	Unfortunately, ChatGPT is unable to run the code directly, and wasn't able to correctly simulate the code and reason out the answer. The issue with its reasoning was it didn't include ln(x+2) into its reasoning.
Claude	48.3s	7.6595130658	Though Claude took almost a minute to generate it's answer, it was able to produce the correct result and show how it got there. It was able to run the code, and explained the output when it was reached.
Gemini	5.3s	8.878516592	Gemini was unfortunately unable to run the code provided, so it tried to reason the answer. I was unable to determine where it made an error, but the produced result is incorrect.
Copilot	None	None	Unfortunately, Copilot is unable to run code directly, and wouldn't reason the answer. This wouldn't be a good choice when trying to work with complex math.
Deepseek	423s	7.655080155	Not a good option for solving complex problems in code. Deepseek would constantly double back on the answer it was generating, and in the end took over 5 minutes to produce the answer.

Test Three Prompt

Create a simple C++ game called BotAttack where robots fight around a grid until one wins.
Requirements: C++, standard libraries, single .cpp file.
Rules: Only struct and functions, no classes.
- No new, delete, or any dynamic memory.
- Only use vector and array.
- Only #include and constants.

Game Features:
- Runs in console.
- 5x5 grid for arena.
- 3 bots which fight each other.
- Each bot has health, a name, random movement.

Main() should remain under 30 lines.

ChatGPT

The Game

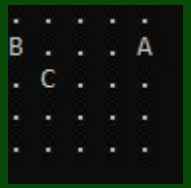
The game by ChatGPT consists of a 5x5 grid with the letters A, B, and C representing the bots. After each turn, the letters move randomly in a direction. If two of the same letters are on the same space, each one takes damage. Eventually, there will be one letter left which is the winner.

How well did it follow restrictions?

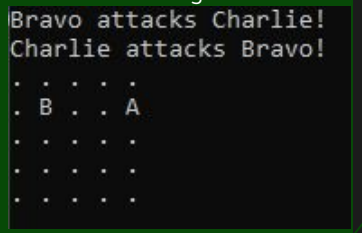
Overall, the code generated and the results are very accurate to what was requested. Each game feature and rule was followed.

UI

Starting Screen



Ending Screen



Thoughts

I like what ChatGPT did for this code. I think that the interactivity with the interactions between each bot being displayed is good, and makes the program easier to understand. I think that ChatGPT followed the instructions well and the code generated sufficiently followed the rules.

Notes

- Game only progressed by user interaction. If you didn't interact with the game by not pressing the continue button it would never end.

Claude

The Game

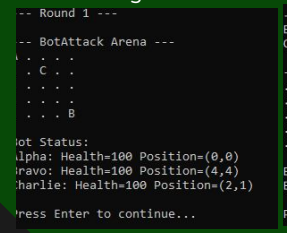
The game generated by Claude consists of a 5x5 grid. When the code is run, the menu pops up with a round screen, the 5x5 arena grid, and the health amounts along with the position of each bot. After each turn, the bots move randomly and if they hit each other, they deal a random amount of damage to each other. The last bot remaining wins after all other bots lose their health.

How well did it follow restrictions?

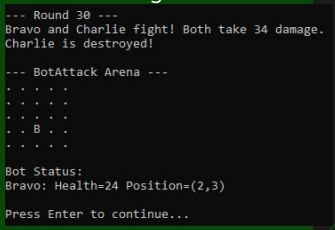
Overall, the code generated and the results are very accurate to what was requested. Each game feature and rule was followed.

UI

Starting Screen



Ending Screen



Thoughts

I think that Claude did an excellent job when generating its program. Not only did it it follow the guidelines, but it also went beyond the prompt by also showing the position of each bot and counting each round that occured.

Notes

- I like how the game shows the position of each bot on the grid, and as the bot is eliminated its health and position disappear.

Gemini

The Game

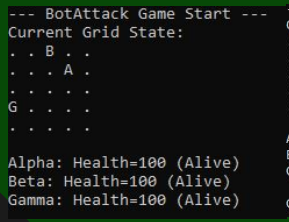
The game generated by Gemini consists of a 5x5 grid. When the game is run, an interface appears showing bot A, B, and G on the grid. This grid is surrounded by an interface showing the bots name and health. After each bots move, the bots position and health is displayed along with who the bot attacked. When all but one bots are eliminated, the final bots name is displayed along with its remaining health.

How well did it follow restrictions?

Overall, the code generated and the results are very accurate to what was requested. Each game feature and rule was followed.

UI

Starting Screen



Ending Screen

Thoughts

I think that Gemini did a good job creating and running the program while also ensuring that it followed the required guidelines. I like how similar to Claude, the program also shows the positions of each bot and how the program is run without user interaction, because sometimes it can take many turns for the game to complete.

Notes

- Unlike the other AI's, the program created by Gemini ran entirely on its own and all turns were automatically completed without user interaction.

Copilot

The Game

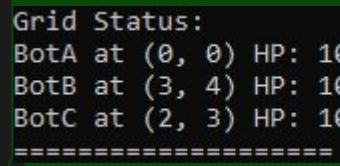
The game generated by Copilot consists of a information screen showing the bots position and health. Each turn is run automatically, and by the end of the game the last bot with positive health wins.

How well did it follow restrictions?

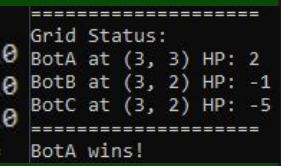
Overall, the code generated and the results are very accurate to what was requested, though no 5x5 grid was displayed. This could be because though the arena is 5x5, no representation of it was requested. Each game feature and rule was followed though bots with negative health are still in the game.

UI

Starting Screen



Ending Screen



Thoughts

I think that Copilot followed the instructions but didn't do anything else other than what was given. Unlike the other AI, Copilots game only consists of text information with no 5x5 grid showing each bots positions.

Notes

- Bots with negative HP are still counted in the game and can take more damage and deal damage to other bots.

Deepseek

The Game

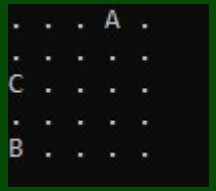
The game generated by Deepseek consists of a 5x5 grid. When the code is run, bots A, B, and C start randomly on this grid. The bots move randomly throughout until two of the bots collide, leaving one bot remaining or a winner.

How well did it follow restrictions?

Overall, this code followed almost every restriction other than each bot having health. Otherwise, it did a good job and followed the guidelines and rules accurately.

UI

Starting Screen



Ending Screen

Thoughts

I think that Deepseek did a great job of following the instructions and producing working code fit the guidelines. The game was interesting to watch and its process of randomization and how it interpreted the rules in the game it made were interesting, like how a bot can only win if the other two come in contact with each other.

Notes

- Because bots didn't have health, if the game didn't last as long as the others.

Results

After these tests, each AI demonstrated different strengths and weaknesses. Depending on what you are trying to do, you may want to use Claude as an assistant, but in different circumstance Copilot might be right for you. It's important to note though that some of the code generated by these

Thank you

Thank you for viewing my presentation.

Morgan Huntley

morgahuntl@gmail.com