Dynamic Memory Management

Manually working with memory requires an attention to detail, and great care in the implementation of the code to avoid issues.

Learning Goals

To complete this task, you need to demonstrate that you can do the following:

- Use memory management function to create and use dynamic memory allocations.
- Illustrate how dynamic memory operations work, showing how values are allocated and manipulated on the heap.
- Incorporate member functions and operators in structs.
- Use and critically evaluate Artificial Intelligence co-pilots to assist with software development tasks.

Focus

As you work through this task, focus on the following aspects of the unit:

- **Programming concepts:** Focus on the heap and how dynamic memory allocations and pointers work together.
- **Programming process**: Focus on how AI co-pilots can assist, and their limitations.
- **Professional characteristics:** Manual memory management requires a focus on developing your attention to detail, and meticulously reviewing code, particularly code generated by AI co-pilots.

Your Task

For this task you will need to submit the following:

- A PDF document containing:
  - Summaries and reflections.
  - Photo of the associated hand execution for your dynamic array and its test program.
  - Screenshots of your programs running.
  - Learning Journey and Resources
- Source code for:
  - The maze game.

1. Complete Learning Activities

Work through these steps to develop and demonstrate your understanding. Aim to demonstrate, to yourself and others, that you have achieved the learning goals.

1. Everything you need is in [Memory Deep Dive](#) and [Member Functions](#).
2. Build a [dynamic array](#) and a small test program to showcase its features. Include the use of the [array access operator](#) to make this look and feel more like a standard array. Use an [AI co-pilot](#) to assist you in this coding.
3. [Hand execute](#) aspects of your code to demonstrate how it works, showing how the operations impact memory on the heap.
4. Build the [Maze Game](#) program, and extend it to include [dynamic rooms and paths](#) based on the [linked list](#).
5. Prepare your summary, making sure to cover all [learning goals](#) and related concepts. Remember that this is a personal summary that demonstrates your understanding of the concepts.
6. Prepare your reflections by responding to the following:
   - How do you know you have achieved the learning goals?

- o What is the most important thing you learned from this and why?
- o How does the content or skills learned here relate to things you already know?

7. Capture your learning journey and collate your evidence of study and practice.

2. Upload Your Submission & Engage with Feedback

Mark the task as **Ready for Feedback** and upload the required files. Make sure to keep copies of these in case you need to resubmit. Then engage with the feedback you receive and get the task Complete!