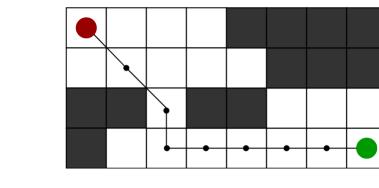
CODE: Store a Grid in Your Program

Introduction to the C++ Language

Store a Grid in Your Program



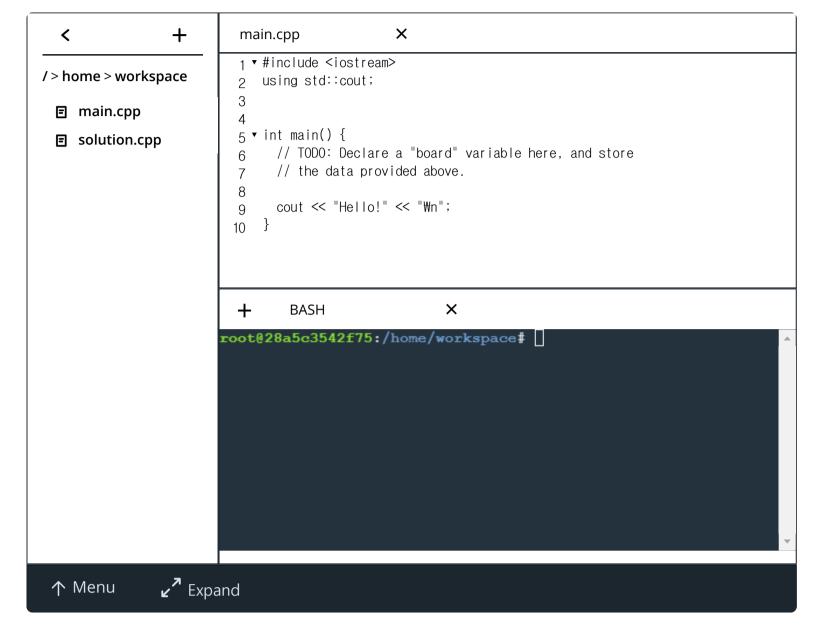
In order to write the A* search algorithm, you will need a grid or "board" to search through. We'll be working with this board throughout the remaining exercises, and we'll start by storing a hard-coded board in the main function. In later exercises, you will write code to read the board from a file.

To Complete This Exercise:

```
    In the main function, declare a variable board as a vector of vectors of ints:
        vector<vector<int>>>.
    Assign this data to the board variable:
        {{0, 1, 0, 0, 0, 0}, {0, 1, 0, 0, 0, 0}, {0, 1, 0, 0, 0, 0}, {0, 1, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 1, 0}}
```

Note: you will need to include the vector library, just as iostream is included. You will also need to use the namespace std::vector if you want to write vector rather than std::vector in your code.

This exercise will be ungraded, but if you get stuck, you can find the solution in **solution.cpp**. Finally, if you feel a little crowded in the editor below and need more space to work, you can click the "Expand" button in the lower left corner.



☑ 1. Intro

CONCEPTS

SEARCH

V 1. III. 0

2. CODE: Write and Run Your First C...

☑ 3. Compiled Languages vs Scripted L...

4. C++ Output and Language Basics

✓ 5. CODE: Send Output to the Console

☑ 7. Bjarne Introduces C++ Types

8. Primitive Variable Types

☑ 9. What is a Vector?

✓ 10. C++ Vectors

✓ 12. Using Auto

13. CODE: Store a Grid in Your Prog...

14. Getting Ready for Printing

15. Working with Vectors

18. CODE: Print the Board

19. If Statements and While Loops

21. CODE: Read the Board from a File

22. Processing Strings

23. Adding Data to a Vector

24. CODE: Parse Lines from the File

25. CODE: Use the ParseLine Function

27. CODE: Formatting the Printed Bo...

28. CODE: Store the Board using the ...

☑ 29. Great Work!