Reading the Board from a File

CODE: Read the Board from a File

Until now, the board has been declared and initialized in the main() function. As discussed in the previous notebook, you will need a function to read the board in from another file in order to make the program a little more flexible and user-friendly.

The first step in this process will be to write a ReadBoardFile function that reads in the file and prints each line to cout. The output should look like the 1.board file, which can be opened in the editor below:

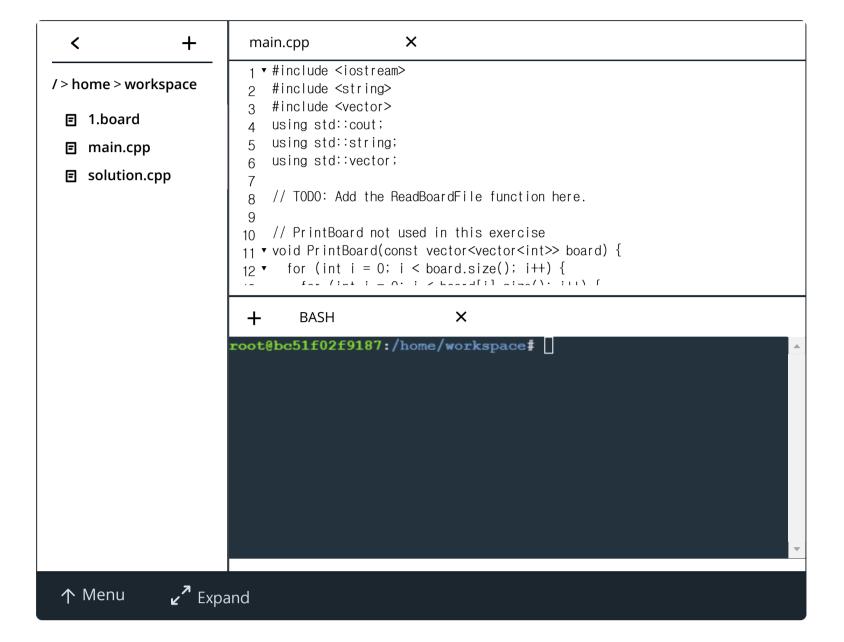
```
0,1,0,0,0,0,
0,1,0,0,0,0,
0,1,0,0,0,0,
0,1,0,0,0,0,
0,0,0,0,1,0,
```

## To Complete This Exercise:

1. Write a function void ReadBoardFile which reads lines from a file. Write each line to cout, followed by a newline character "\n". The function should accept a string argument, which represents the path to the file. For now, since ReadBoardFile does not need to return

anything, you can use a void return type. 2. Call ReadBoardFile from main using the argument "1.board".

Note: you will need to include the fstream class, and you may want to have using std::ifstream to use the ifstream object without having to write std:: in the rest of your code.



- ✓ 12. Using Auto

11. C++ Comments

- 13. CODE: Store a Grid in Your Progr...
- 14. Getting Ready for Printing
- 15. Working with Vectors
- ✓ 16. For Loops
- 17. Functions
- ☑ 18. CODE: Print the Board
- 19. If Statements and While Loops
- 20. Reading from a File
- 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

- 26. Formatting the Printed Board
- 27. CODE: Formatting the Printed Bo...
- 28. CODE: Store the Board using the ...
- ≤ 29. Great Work!

SEND FEEDBACK