Formatting the Printed Board

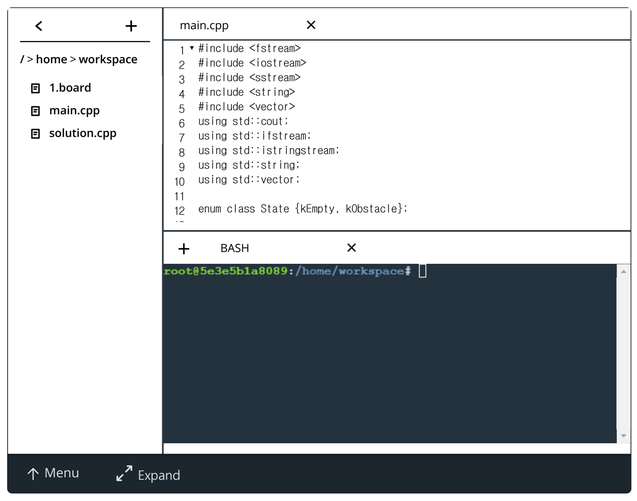
```
0  ⛬  0  0  0  0
0  ⛬  0  0  0  0
0  ⛬  0  0  0  0
0  ⛬  0  0  0  0
0  0  0  0  ⛬  0
```

The board will eventually have more than two cell states as the program becomes more complicated, and it would be nice to add formatting to the printed output of the board to ensure readability as the number of board states increases.

To accommodate more board states and facilitate print formatting, we have provided the `State` enum with enumerator values `kEmpty` and `kObstacle`. In this exercise, you will write a `CellString` function which converts each `State` to an appropriate string. In the next exercise, we will update the program to use the `enum` values and `CellString` function.

To Complete This Exercise:

1. Write a function `CellString` which accepts a `State` as input and returns the following string:
   - `"⛬ "` if the input equals a `kObstacle`
   - `"0 "` otherwise.



NEXT