FF-MakeUp: STACK

Oct 5, 2023

Create a video explaining and implementing the code for the following instructions

P1: Design a *Stack* Class

You may not use std::stack. You may use std::Queue
Implement a stack class using only queues.
Your implementation must include these following member functions:

push(x)

pop()

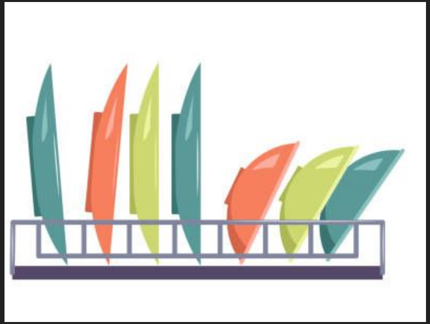
top()

size()

getMin() → Returns the minimum amount in the stack

Explain normal stack functionality and how you recreated it with queues.

P2: Plate Sorting



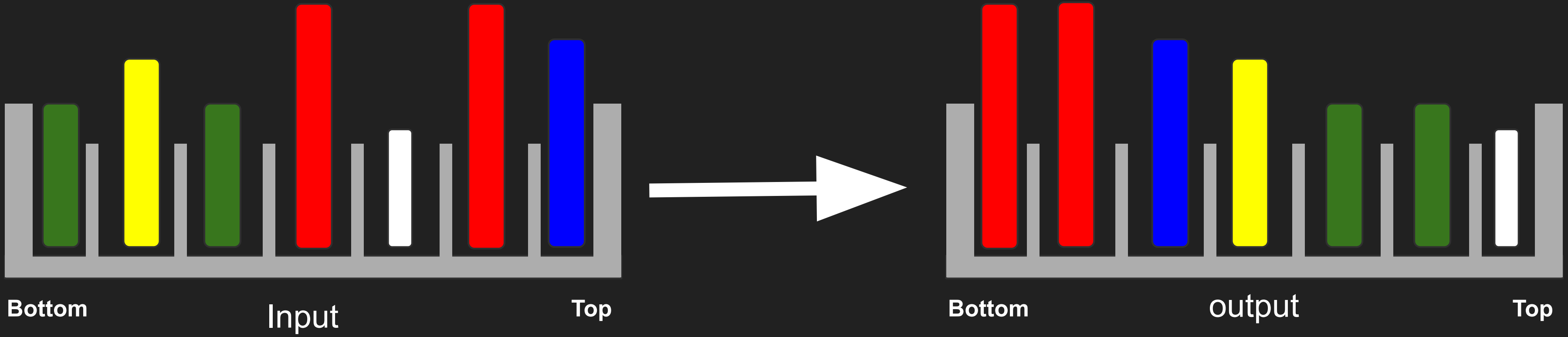
You may use std::stack

Design a program that sorts a plate rack using only one other empty plate rack.

You are given a stack (plate_st) representing a plate rack. Sort it, using only std::stack functions and one other stack, so that the stack is in order (Biggest at bottom, smallest at top). At the end of the algorithm the elements must be sorted in its original stack (plate_st). Print the sorted stack.

“Stack Functions” found here under “Member Functions” → https://cplusplus.com/reference/stack/stack/

P2: Input → Output example



Input:
45
55
5
55
25
35
25

key

🟥 55
🟨 35
🟦 45
🟩 25
⬜ 5

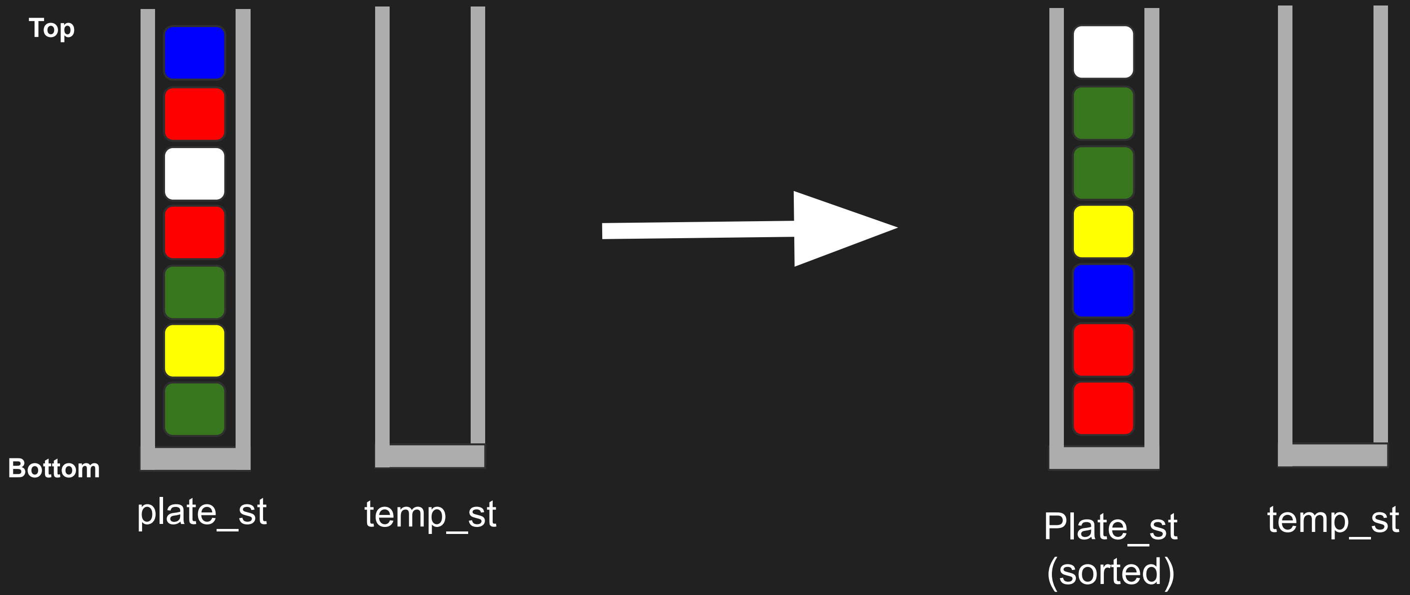
Output:
5
25
25
35
45
55
55

P2: Visualized

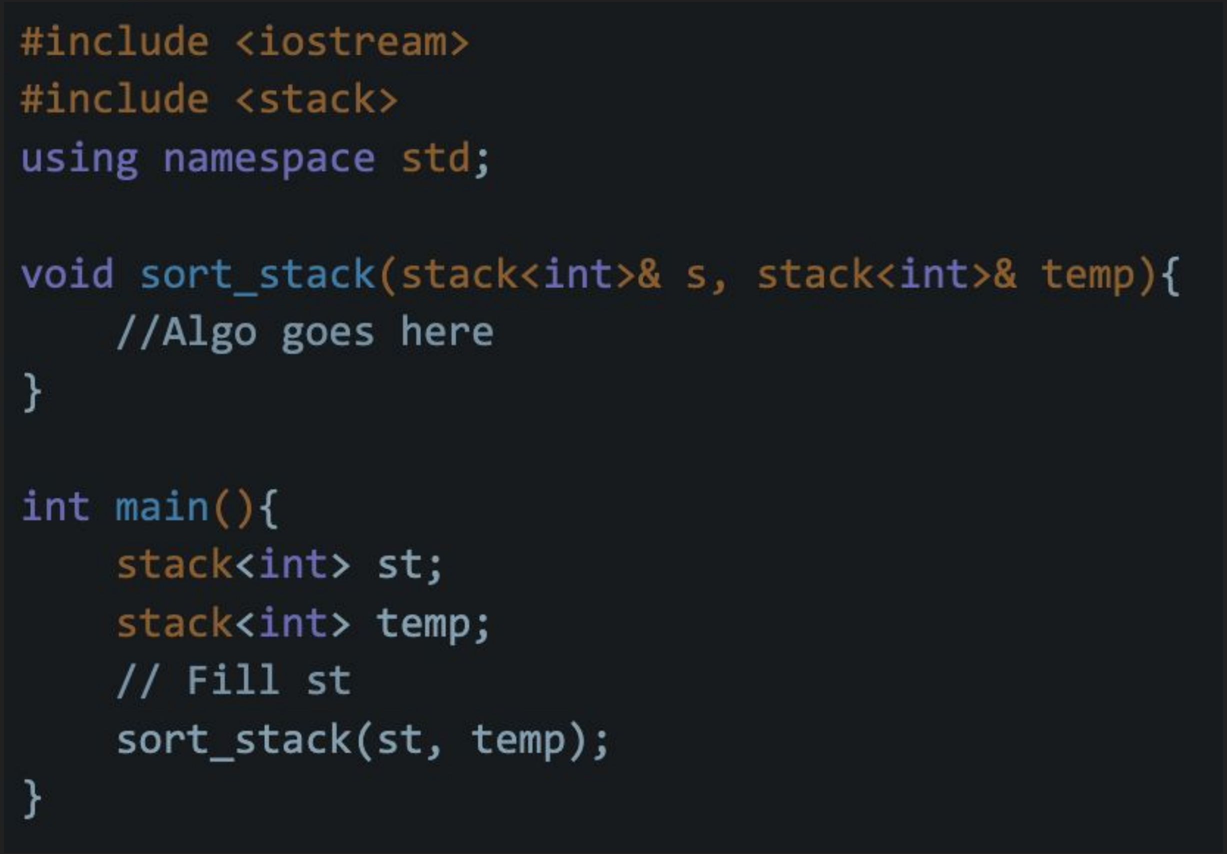
key

- 🟥 55
- 🟦 45
- 🟨 35
- 🟩 25
- ⬜ 5

You may only use std::stack functions and one additional stack (temp_st)



P2: Sample Starter Code



```cpp
#include <iostream>
#include <stack>
using namespace std;

void sort_stack(stack<int>& s, stack<int>& temp){
    //Algo goes here
}

int main(){
    stack<int> st;
    stack<int> temp;
    // Fill st
    sort_stack(st, temp);
}
```

Temp stack must remain empty until sort_stack() function begins

Algo Example:
https://docs.google.com/presentation/d/1I7EcLhBp3OP-et_qsMtNfMchwe8Oz6vwrIicdUaEqEk/edit?usp=sharing

SUBMIT

Upload video to coogTube or any other video sharing platform (youtube)

FF → Coogtube

Upload code to repl.it or any other code sharing platform

Fill out https://forms.gle/8GJ4SXoFK5njm8sV9

SUBMIT **BEFORE** 10 PM

Contact me if there are any upload/submission errors