SECJ1013-01&05 & SCSJ1013-01&05

20192020-01

Assignment 2 (10%)

Write a program that can be used by a small theater to sell tickets for performances. The theater's auditorium has 15 rows of seats, with 30 seats in each row. The program should display a screen that shows which seats are available and which are taken. For example, the following screen shows a chart depicting each seat in the theater. Seats that are taken are represented by an * symbol, and seats that are available are represented by a # symbol:

```
                         Seats
              123456789012345678901234567890
ROW 1         ***###***###*########*****####
ROW 2         ####*************####*******##
ROW 3         **###**********########****###
ROW 4         **######**************##******
ROW 5         ********#####*********########
ROW 6         ##############************####
ROW 7         #######************###########
ROW 8         ************##****############
ROW 9         #########*****############****
ROW 10        #####*************############
ROW 11        #**********#################**
ROW 12        #############********########*
ROW 13        ###***********########**######
ROW 14        ##############################
ROW 15        ##############################
```

Here is a list of tasks this program must perform:

- When the program begins, it should ask the user to enter the seat prices for each row. The prices can be stored in a separate array.
- Once the prices are entered, the program should display a seating chart similar to the one shown above. The user may enter the row and seat numbers for tickets being sold. Every time a ticket or group of tickets is purchased, the program should display the total ticket prices and update the seating chart.

- The program should keep a total of all ticket sales. The user should be given an option of viewing this amount.
- The program should also give the user an option to see a list of how many seats have been sold, how many seats are available in each row, and how many seats are available in the entire auditorium.

Input Validation: When tickets are being sold, do not accept row or seat numbers that do not exist. When someone requests a particular seat, the program should make sure that seat is available before it is sold.

Other Requirements: Applying proper styles, e.g. indentation and comments. Using an appropriate structure for the program (e.g. all required header files are included, the function main, functions definition, function prototypes are properly written, etc.)

Output: Capture the screen outputs for every given task and convert them to a pdf file.

Submission: You are required to submit the program/code and the output by 11.59pm on 5th December 2019. The submission files should be *.cpp (program/code) and *.pdf (output). Example CodeA2.cpp and OutputA2.pdf.