**CS 2060 Programming with C - Fall 2017**

**Assignment #5**

Due Date: Oct 4, 2017 at 9:25am (MW class), Oct 5, 2017 at 9:25am (TR class)

Purpose: Learn to use functions

Effort: Individual

Points: 100

Deliverables: Upload the .c source code file to Blackboard by due date.

Please include pseudocode in the comments at the beginning of your code.

**Please hand in a hardcopy version of your code at beginning of class.**

**Assignment Description**

Write a program to sell concert tickets for a venue that hosts 3 groups. You can pick any three groups of your choice. You **must** use the ticket prices I’ve included in the specifications below. The program must allow the customer to do the following:

* View a menu of the three groups playing at the venue
* Select one of the three groups
* View the venue sections and the prices associated with each section
* Select the section to purchase seats in
* Select the number of seats to purchase
* View a summary of the ticket order
* Decide within 30 seconds if they want to purchase the tickets

**Specifications**

1. Create a C project called **Assignment5 (please use this exact name)**
2. Follow "CS2060 Programming Assignments Policy"
3. Concert Details:
   1. The concert venue has the following sections. Seat price is based on desired section.

|  |  |  |
| --- | --- | --- |
| **Section** | **Section Numbers** | **Seat Price** |
| Floor | A – F | $400.00 |
| 100 | 102-148 | $325.00 |
| 200 | 202-260 | $250.00 |
| 300 | 302-380 | $100.00 |

1. **Each ticket** will also be charged a “service fee and taxes” of $23.50.
2. Customer can purchase only 1 to 6 seats.
3. When displaying the order information, the **section number**, the **row**, and **seat** must be randomly generated values.
   1. Section numbers:
      1. Make sure the random **section number** fits into the range of numbers shown in above table. For example, if customer selected section 100, then the random section number must be between 102 and 148.
   2. Row number - value between 1 to 30
   3. Seat number – value between 1 to 15
4. When purchasing more than one seat, make sure the seats are together in the same row! For example, if 3 seats are desired and the random number generator gives you seat 14, saying the seats are 14-16 doesn’t work because there isn’t a seat 16. You need seats 13,14,15.

Quantity Section Row Seats Seat Price Total

-------------------------------------------------------------------

4 A 11 11-14 $400.00 $1600.00

Service Fee & Taxes**(x4)** $94.00

-------------------------------------------------------------------

Total: $1694.00

1. After displaying ticket order information, call the function **purchaseTickets**. The first task this function must do is get a start time. Next, it will ask the customer if they want to purchase the tickets. Only after the user answers the purchase question should you get the stop time. If the difference between the **stop time** and **start time** is more than 30 seconds, the order is timed out and void. This is a “very” simple way to “kind of” simulate a stopwatch. Here are some pieces you need to get started.

* #include <time.h>
* time\_t startTime = time(NULL);
* Use the **difftime** function in time.h to determine elapsed time
  + Open time.h file to view the function’s prototype (see lecture #8 notes)

1. Your code is a one-time run through main. No loop is needed to repeatedly ask customer about purchasing . Customer selects options and either buys the tickets or not. Main point is to learn to use functions properly.
2. Write code that:
   1. Does **NOT** use global variables; you can and **SHOULD** use global constants.
      1. A variable declared **before** main is called a global variable.
   2. Gracefully handles invalid input
      1. Allow customer to repeatedly enter **concert**, **section**, **number of seats**, and **purchase decision** until a valid value is entered.
   3. Uses proper data types and constants.
      1. What would happen if at 9:15 the morning your assignment is due I say, “ticket prices have all gone up and need to be changed to some new amount.” Does your code allow you to make this change quickly and efficiently?
   4. Uses proper control structures
      1. Think about what control structures should be used to solve the different patterns the problem presents.
   5. Is clean!
      1. Be mindful where you are repeating code and rewrite those sections.
      2. We will be grading on redundant code on this assignment.
3. Design your program to use the following functions:

You may add more, but as a minimum write code for these functions.

// Displays the concert menu.

// Prompts customer for desired concert and returns selected concert (group).

**int** displayConcertMenu (**void**);

// Displays the sections and prices associated with those sections.

// Prompts customer for desired section for selected concert (group).

// Returns selected section.

**int** getSection (**int** whichConcert);

// Prompts customer for the number of seats they want for the selected concert.

// The customer can purchase between 1 and 6 seats.

**int** getNumberOfSeats (**int** whichConcert);

// Generate a random section number and returns that section number.

**int** generateRandomSectionNumber (**int** section)

// Generate a random row number and returns that row.

**int** generateRandomRowNumber (**void**);

// Generates a random seat number and returns that seat.

**int** generateRandomSeatNumber (**void**);

// Calculate the total cost and display order information.

**float** ticketCost (**int** section, **int** numSeats, **int** sectionNumber, **int** row, **int** seat);

// Prompts customer to purchase tickets. Starts a timer. If customer doesn't purchase

// tickets within 30 seconds, then void the order, otherwise charge credit card.

bool purchaseTickets(**float** totalCost);

// Print tickets for each seat purchased

**void** printTickets (**int** whichConcert**, int** section, **int** numberSeats,

**int** sectionNumber, **int** row, **int** seat);

**Tips**

* Write code incrementally
  + Step 1: write code for each function
  + Step 2: test each function to make sure the function produces correct results
  + Step 3: build code in main around the functions
* If you write the main program correctly, it will essentially be calls to all the functions.

**Output**

Your output may look similar to the following:

**Output - Example #1**

------------------------------

1: Rolling Stones

2: Imagine Dragons

3: U2

------------------------------

Which concert would you like to attend? **2**

-------------------------------------------

Section Price

-------------------------------------------

0: Floor A - F $400.00

1: Section 102 - 148 $325.00

2: Section 202 - 260 $250.00

3: Section 302 - 380 $100.00

-------------------------------------------

In what section would you like to view Imagine Dragons? **0**

How many seats for the Imagine Dragons? **3**

Quantity Section Row Seats Seat Price Total

-------------------------------------------------------------------

3 E 10 10-12 $400.00 $1200.00

Service Fee & Taxes(x3) $70.50

-------------------------------------------------------------------

Total: $1270.50

You have 30 seconds to decide if you want to purchase these tickets (Y/N)? **y**

Thank you for your purchase. Your credit card will be charged $1270.50

Imagine Dragons

-------------------------

Section | Row | Seat |

-------------------------

E 10 10

-------------------------

Imagine Dragons

-------------------------

Section | Row | Seat |

-------------------------

E 10 11

-------------------------

Imagine Dragons

-------------------------

Section | Row | Seat |

-------------------------

E 10 12

-------------------------

**Error Handling Output - Example 1**

------------------------------

1: Rolling Stones

2: Imagine Dragons

3: U2

------------------------------

Which concert would you like to attend? **4**

4 is not a valid option. Please try again. **0**

0 is not a valid option. Please try again. **1**

-------------------------------------------

Section Price

-------------------------------------------

0: Floor A - F $400.00

1: Section 102 - 148 $325.00

2: Section 202 - 260 $250.00

3: Section 302 - 380 $100.00

-------------------------------------------

In what section would you like to view the Rolling Stones? **4**

4 is not a valid section. Please try again. **6**

6 is not a valid section. Please try again. **1**

How many seats for the Rolling Stones? **7**

**7 seats is not valid. Please select between** 1 and 6 seats. **0**

0 seats is not valid. Please select between 1 and 6 seats. **1**

Quantity Section Row Seats Seat Price Total

-------------------------------------------------------------------

1 121 12 12 $325.00 $325.00

Service Fee & Taxes(x1) $23.50

-------------------------------------------------------------------

Total: $348.50

You have 30 seconds to decide if you want to purchase these tickets (Y/N)? **y**

p is not valid. Please enter Y or N: **t**

t is not valid. Please enter Y or N: **y**

Sorry, this order expired, please try again