Swinburne University of Technology Sarawak
COS10009
Introduction to Programming
Semester 1, 2018

Custom Program

Due Date: 5 PM on Friday of week 12 (25 May 2018).

Submission Guidelines: Your program and report must be submitted as part of your portfolio in an A4 paper envelope with a facing sheet attached to the front of envelope. Late penalties will apply as described in the unit outline.

Deliverables: (X means required)

1	Facing sheet with your signature	X
2	All source files, data files, extra libraries, project file, and the executable file. The source C or project PRJ files (if any) ***must*** be named as *studentID_surname.C* or *.PRJ*	X
3	Printed design report, including user manual, description of program design and flowchart.	X
4	Screen capture of the program output. (optional)	-
5	Print-out of the source code.	X

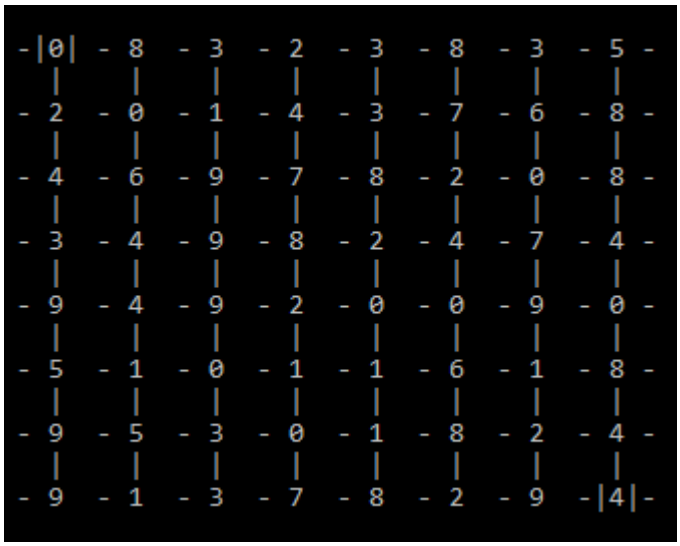
In the design report,** write a short description of program design and brief user manual. Also, submit a **Flowchart** showing the logic in the "main(void)" function.*

Topic: The Manhattan Tourist Problem

Background:
Dynamic programming is often illustrated by using Manhattan Tourist Problem, which is also known as Longest Path Problem. It is an effort to navigate for a path from a given source to a given destination in a Manhattan-like city grid that maximizes the number of attractions. Since there are many attractions along every street in the city, the tourists will try to visit as many attractions as possible, and they will only move either to the south or to the east. The city map in this case can be represented as a grid with the numbers next to each line (called weights) showing the number attractions on very block. We will refer to this kind of construct as a graph, the intersections of streets are called vertices, and the streets themselves will be edges with weight associated to them.

Video:
https://www.youtube.com/watch?v=wmIMeyWDZI0

Specification & Requirement:



- Develop a game with C that will generate 64 integers (eg:0-9) on a 8 by 8 grid (2-dimensional array).

- You are expected to design your own mechanism of generating the integers to start the game. It should be documented in your design report to elaborate the details of your mechanism.

- You are allowed to design your own rules of playing the game, such as the size of the board, ways to generate the integers, routing direction, randomize starting point and destination, provide hints or suggestions to players, winning condition etc.

- The performance of the player may be measured by the total attractions visited throughout the entire tour. (You may introduce other ways of measuring player's performance).

- Each element of array contains the following ***suggested*** attributes :
    - a variable of integer / character
    - and any other attributes you think appropriate to your program

  Thus, the element can be described as "struct" of array in the C programming language.

- The record of previous players is stored in a plain text data file. The information of all the previous players should be displayed at the beginning of the game. User / player will also be prompted to key in their information, and that information will then be stored in the same text file.

- Input to the system is through a DOS command window.

- This program must first prompt the user (in the command window) for the next action to be carried out. (Do provide appropriate menu of actions). The user should be able to terminate the program at any time.

- Output should be printed on DOS window to show the latest layout of the board.

- The program codes MUST be in functions. DO NOT write all the code in the "main()" function.

- Must NOT use global variable and goto in the program.

- Graphical window interface can be used as an alternative (optional)

- This C program is recommended to be written using Quincy 2005. Build options must include strict ANSI/ISO compliance, C99 support.

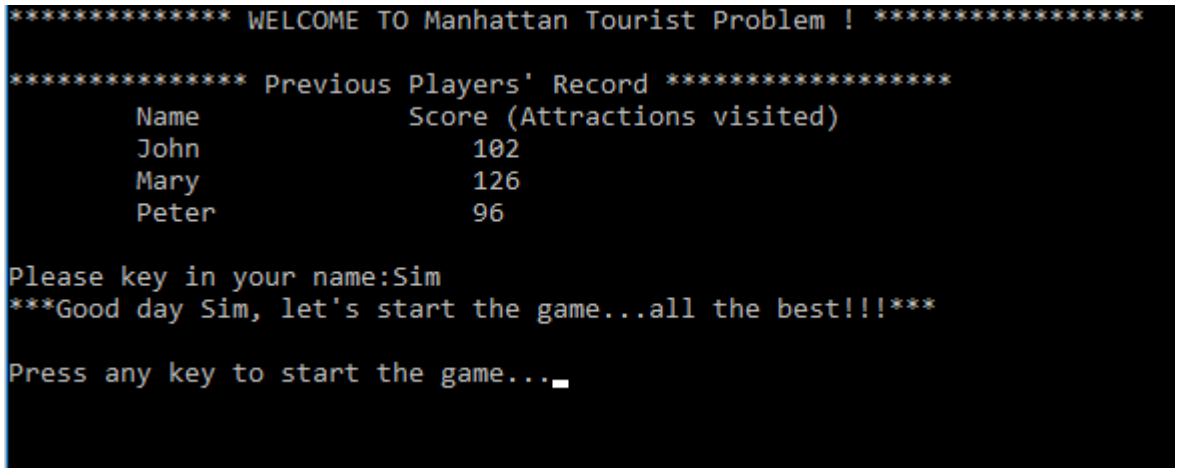
Procedure of Program Development

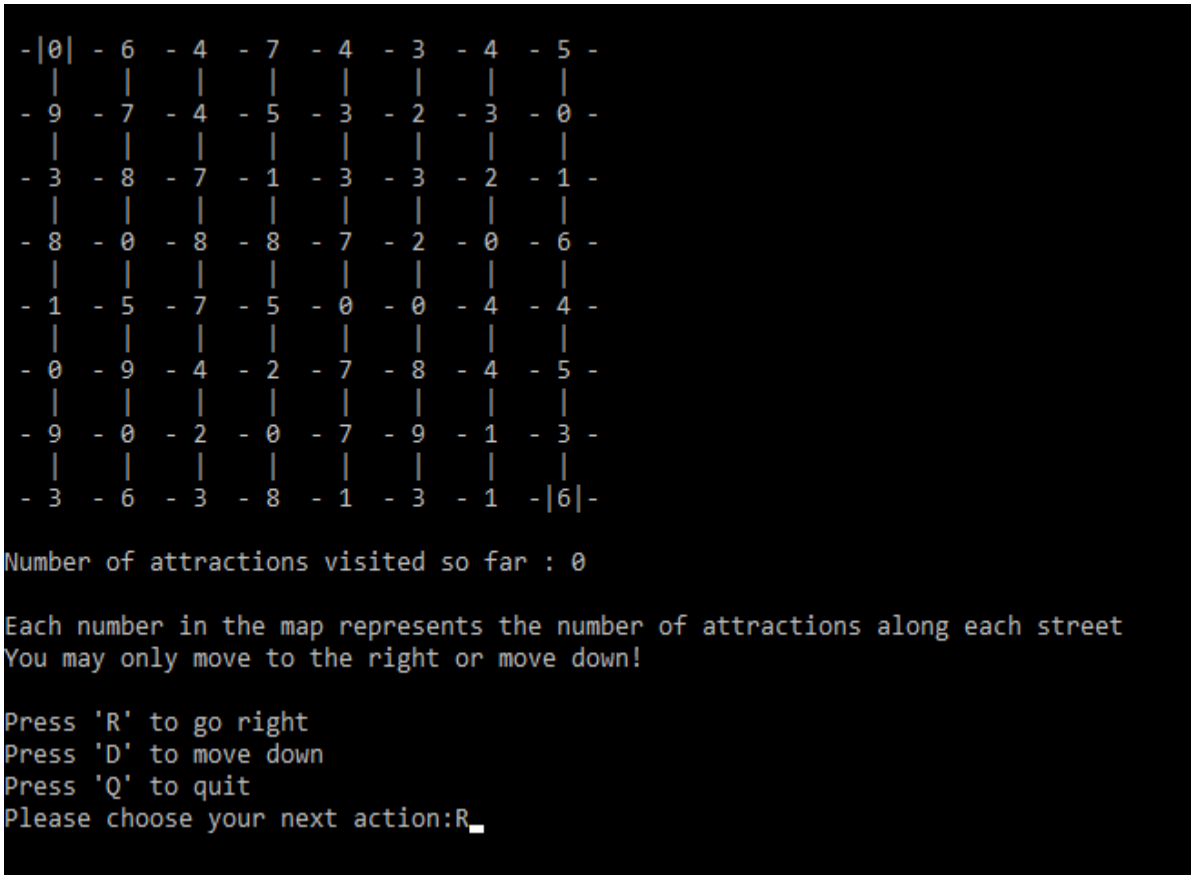
Use an iterative process (spiral or prototype SDLC) to design and implement a solution to your program. That means

1. write a small program to do a little bit of the problem
    a. compile and run the small program
2. next, add a function or two
    a. compile and run the improved program
3. repeat step 2 until you are finished

(Kindly refer the last page of this document for the assessment details)

Snapshots:


********** WELCOME TO Manhattan Tourist Problem ! **********
********** Previous Players' Record **********
Name Score (Attractions visited)
John 102
Mary 126
Peter 96
Please key in your name:Sim
Good day Sim, let's start the game...all the best!!!
Press any key to start the game...


-|0| - 6 - 4 - 7 - 4 - 3 - 4 - 5 -
- 9 - 7 - 4 - 5 - 3 - 2 - 3 - 0 -
- 3 - 8 - 7 - 1 - 3 - 3 - 2 - 1 -
- 8 - 0 - 8 - 8 - 7 - 2 - 0 - 6 -
- 1 - 5 - 7 - 5 - 0 - 0 - 4 - 4 -
- 0 - 9 - 4 - 2 - 7 - 8 - 4 - 5 -
- 9 - 0 - 2 - 0 - 7 - 9 - 1 - 3 -
- 3 - 6 - 3 - 8 - 1 - 3 - 1 -|6|-
Number of attractions visited so far : 0
Each number in the map represents the number of attractions along each street
You may only move to the right or move down!
Press 'R' to go right
Press 'D' to move down
Press 'Q' to quit
Please choose your next action:R

```
 -|0| -|6| - 4  - 7  - 4  - 3  - 4  - 5 -
   |    |    |    |    |    |    |    |
 - 9  - 7  - 4  - 5  - 3  - 2  - 3  - 0 -
   |    |    |    |    |    |    |    |
 - 3  - 8  - 7  - 1  - 3  - 3  - 2  - 1 -
   |    |    |    |    |    |    |    |
 - 8  - 0  - 8  - 8  - 7  - 2  - 0  - 6 -
   |    |    |    |    |    |    |    |
 - 1  - 5  - 7  - 5  - 0  - 0  - 4  - 4 -
   |    |    |    |    |    |    |    |
 - 0  - 9  - 4  - 2  - 7  - 8  - 4  - 5 -
   |    |    |    |    |    |    |    |
 - 9  - 0  - 2  - 0  - 7  - 9  - 1  - 3 -
   |    |    |    |    |    |    |    |
 - 3  - 6  - 3  - 8  - 1  - 3  - 1  -|6|-

Number of attractions visited so far : 6

Each number in the map represents the number of attractions along each street
You may only move to the right or move down!

Press 'R' to go right
Press 'D' to move down
Press 'Q' to quit
Please choose your next action:D
```

```
 -|0| -|6| - 4  - 7  - 4  - 3  - 4  - 5 -
   |    |    |    |    |    |    |    |
 - 9  -|7| - 4  - 5  - 3  - 2  - 3  - 0 -
   |    |    |    |    |    |    |    |
 - 3  - 8  - 7  - 1  - 3  - 3  - 2  - 1 -
   |    |    |    |    |    |    |    |
 - 8  - 0  - 8  - 8  - 7  - 2  - 0  - 6 -
   |    |    |    |    |    |    |    |
 - 1  - 5  - 7  - 5  - 0  - 0  - 4  - 4 -
   |    |    |    |    |    |    |    |
 - 0  - 9  - 4  - 2  - 7  - 8  - 4  - 5 -
   |    |    |    |    |    |    |    |
 - 9  - 0  - 2  - 0  - 7  - 9  - 1  - 3 -
   |    |    |    |    |    |    |    |
 - 3  - 6  - 3  - 8  - 1  - 3  - 1  -|6|-

Number of attractions visited so far : 13

Each number in the map represents the number of attractions along each street
You may only move to the right or move down!

Press 'R' to go right
Press 'D' to move down
Press 'Q' to quit
Please choose your next action:D
```

```
 -|0| -|6| - 4  - 7  - 4  - 3  - 4  - 5 -
   |    |    |    |    |    |    |    |
 - 9  -|7| - 4  - 5  - 3  - 2  - 3  - 0 -
   |    |    |    |    |    |    |    |
 - 3  -|8| - 7  - 1  - 3  - 3  - 2  - 1 -
   |    |    |    |    |    |    |    |
 - 8  -|0| -|8| -|8| - 7  - 2  - 0  - 6 -
   |    |    |    |    |    |    |    |
 - 1  - 5  - 7  -|5| - 0  - 0  - 4  - 4 -
   |    |    |    |    |    |    |    |
 - 0  - 9  - 4  -|2| -|7| -|8| - 4  - 5 -
   |    |    |    |    |    |    |    |
 - 9  - 0  - 2  - 0  - 7  -|9| - 1  - 3 -
   |    |    |    |    |    |    |    |
 - 3  - 6  - 3  - 8  - 1  -|3| -|1| -|6|-

Number of attractions visited so far : 72

Each number in the map represents the number of attractions along each street
You may only move to the right or move down!

Press 'R' to go right
Press 'D' to move down
Press 'Q' to quit
Please choose your next action:R

Congratulation! You have completed your tour!

Total attractions you have visited = 78

Thanks for playing!!! See you again...
```

COS10009
Semester 1, 2018
Marking Scheme

Student ID: _____ Student Name: _____

Assessment of this custom program will be part of the portfolio interview. Each interview will last about 10-15 minutes.

Pass			**Credit**			**Distinction**			**High Distinction**			
50	55	59	60	65	69	70	75	79	80	85	90	100
▪ Design report consists of required components ▪ Flowchart is correct, corresponds to logic of main() function ▪ Use functions and parameters ▪ Adequate comments to explain the code ▪ C code compiles and links, executable runs, does not crash			In addition to including the criteria required for Pass, the custom program must: ▪ Use functions, parameters, pointers and struct, ▪ Meet most of the requirements ▪ Read and write external text/ data file ▪ Program handles errors gracefully, allowing the user to continue.			In addition to including the criteria required for Credit, the design report must: ▪ Have good descriptioin of core program functionality and how it works The custom program must: ▪ Meet all the requirements ▪ Code written with proper indentation that helps document program structure ▪ Bugs free			In addition to including the criteria required for Distinction, the software solution must have: ▪ Code with extensively comments that provide meaningful insights into the code being documented ▪ Additional feature other than the specified requirements. ▪ The additional feature must be thoroughly documented in the design report ▪ The design report must demonstrate good communication skills, and present a well thought out program design			

Evidence of plagiarism: