**Table of figures**

[Figure 2 4](#_Toc48765985)

[Figure 3 4](#_Toc48765986)

[Figure 4 4](#_Toc48765987)

[Figure 5 5](#_Toc48765988)

[Figure 6 5](#_Toc48765989)

[Figure 7 5](#_Toc48765990)

[Figure 8 6](#_Toc48765991)

[Figure 9 7](#_Toc48765992)

[Figure 10 7](#_Toc48765993)

[Figure 11 7](#_Toc48765994)

[Figure 12 8](#_Toc48765995)

[Figure 13 8](#_Toc48765996)

[Figure 14 9](#_Toc48765997)

[Figure 15 11](#_Toc48765998)

[Figure 16 11](#_Toc48765999)

[Figure 17 12](#_Toc48766000)

[Figure 18 12](#_Toc48766001)

[Figure 19 13](#_Toc48766002)

**Table of content**

[Chapter 4: IMPLEMENTATION AND RESULTS 2](#_Toc48765870)

[4.1 C source files in the design program 2](#_Toc48765871)

[4.1.1 Main function 2](#_Toc48765872)

[4.1.2 Input ID function 2](#_Toc48765873)

[4.1.3 Input grades function 3](#_Toc48765874)

[4.1.4 Input the number student and create student list function 3](#_Toc48765875)

[4.1.5 Display all student list 3](#_Toc48765876)

[4.1.6 Display students have highest/lowest grades 4](#_Toc48765877)

[4.2 Plan program activity checks 4](#_Toc48765878)

[4.2.1 Result of case-data 1: (Input valid data) 4](#_Toc48765879)

[4.2.2 Result of case-data2: (Input invalid data) 6](#_Toc48765880)

[4.3 Analyse and testing program 7](#_Toc48765881)

[4.3.1 Table of detail 7](#_Toc48765882)

[4.3.2 The console screen in the testing 8](#_Toc48765883)

[4.4 Review/ Evolution of this program 10](#_Toc48765884)

[4.4.1 Advantages 10](#_Toc48765885)

[4.4.2 Disadvantages 11](#_Toc48765886)

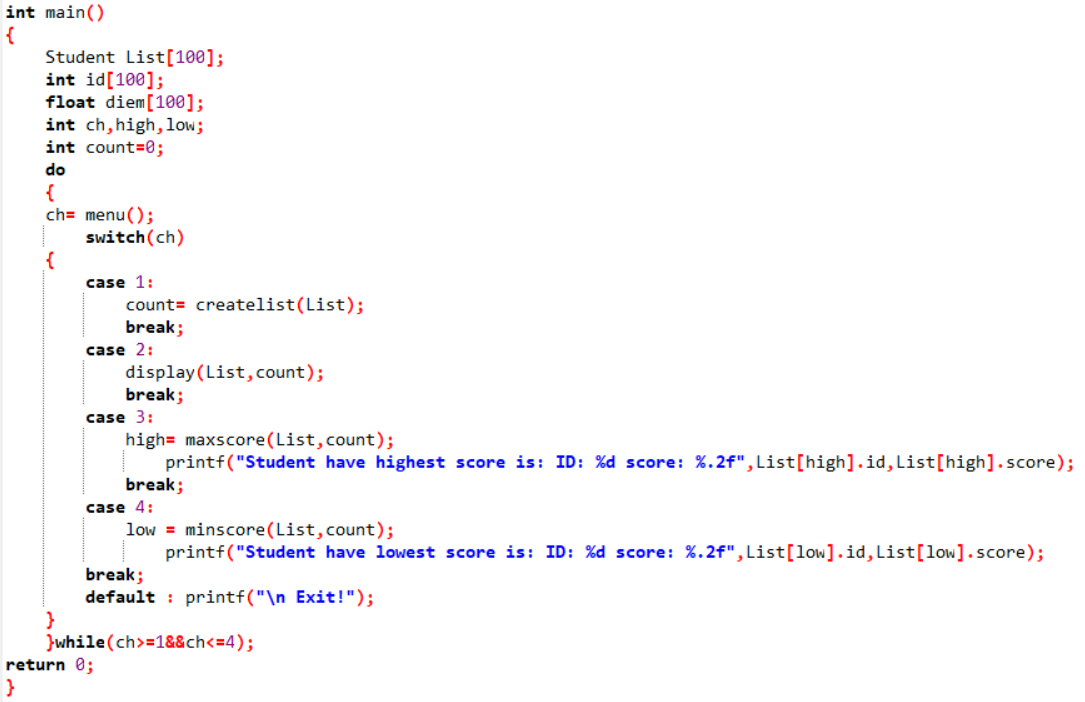
[4.5 Lessons learned after doing the homework 11](#_Toc48765887)

[EPILOGUE 11](#_Toc48765888)

# Chapter 4: IMPLEMENTATION AND RESULTS

## 4.1 C source files in the design program

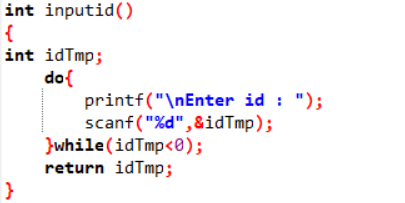
### 4.1.1 Main function



Figure

The main function includes declaring local variables, calling functions, and executing functions on the screen.

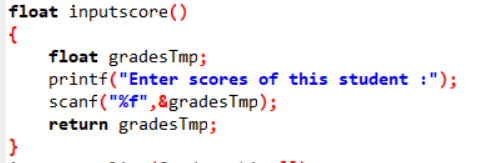
### 4.1.2 Input ID function



Figure

function used to enter data ID

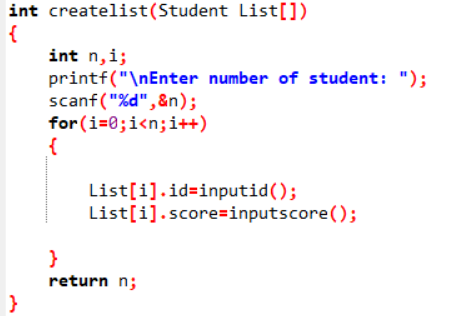
### 4.1.3 Input grades function



Figure

function used to enter grades data.

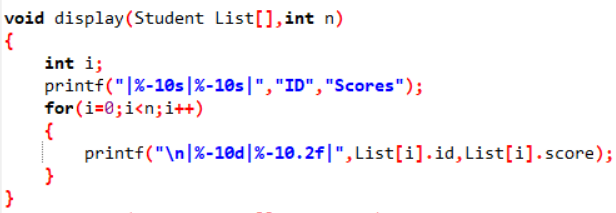
### 4.1.4 Input the number student and create student list function



Figure

The function is used to enter student numbers and create a student list that allows entering student ID and scores.

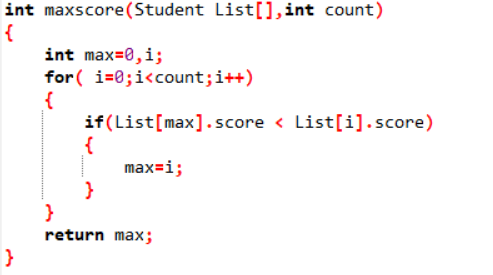
### 4.1.5 Display all student list



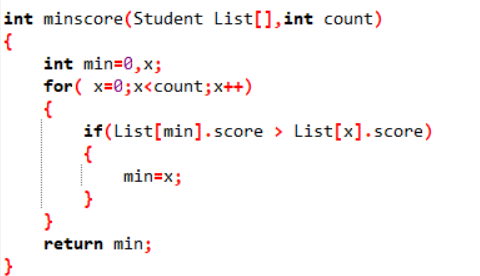
Figure

Function to print to the screen all student lists including ID and score

### 4.1.6 Display students have highest/lowest grades



Figure



Figure

The function finds the student with the highest score and the lowest score

## 4.2 Plan program activity checks

### 4.2.1 Result of case-data 1: (Input valid data)

Press 1:

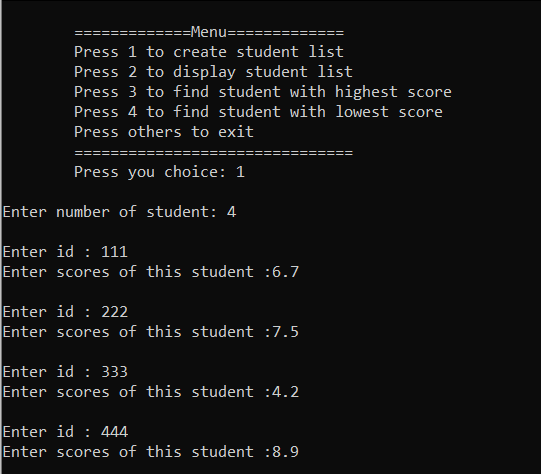
Enter 4 students

1. ID: 111; score: 6.7

2. ID: 222; score: 7.5

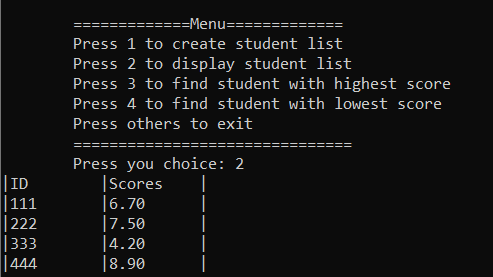
3. ID: 333; score: 4.2

4. ID: 444; score: 8.9



Figure

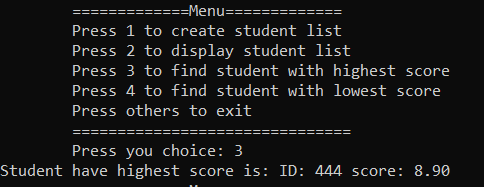
Press 2:



Figure

Display all student list.

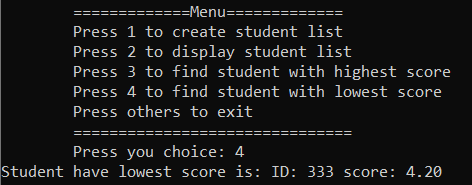
Press 3:



Figure

Display the student have highest score.

Press 4:

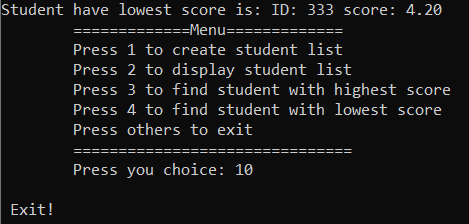


Figure

Display the student have lowest score.

### 4.2.2 Result of case-data2: (Input invalid data)

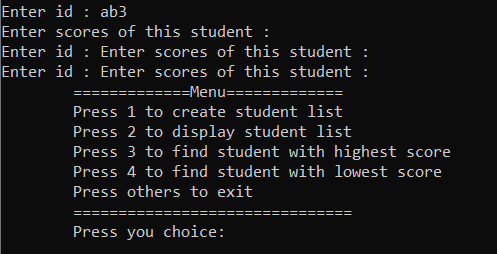
Case 1: Choice 10 on menu



Figure

Announcement in the screen is: “ Exit! ” and exit the program.

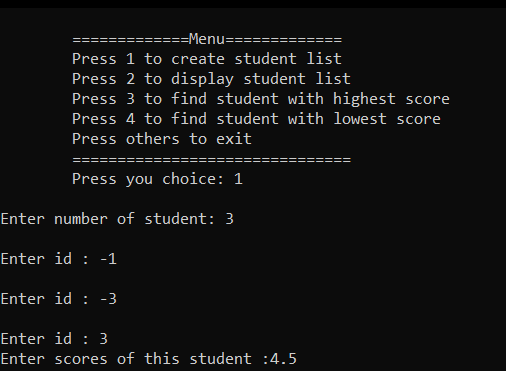
Case 2: Choice 1 but input id is strings.



Figure

The program keeps running but can’t input others id or score.

Case 3: Input id < 0



Figure

The program will force you to re-enter until id > 0

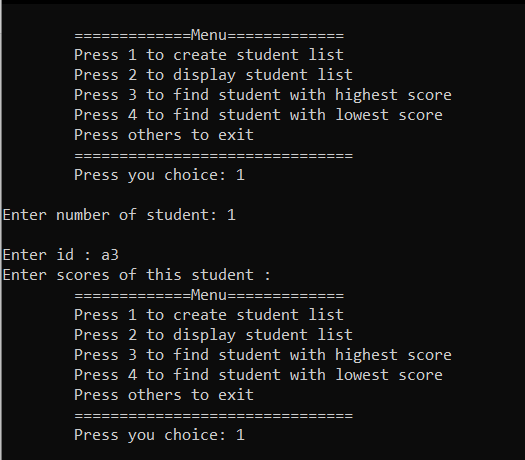
## 4.3 Analyse and testing program

### 4.3.1 Table of detail

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test case Numbe-r** | **Subjects to be inspecte-d** | **Preconditio-n** | **Step to take** | **Input data** | **Expecte-d Output** | **Actual-y output** | **Statu-s** |
| **1** | Input ID | ID is an integer | Press 1  Enter number 1  Enter ID: a3 | a3 | **ERRO** | Keep running but can input others variable | **Fail** |
| **2** | Input number of students | Number is an integer | Press 1  Enter number: z3 | z3 | **ERRO** | After a few seconds the program exits | **Pass** |
| **3** | Input scores | Score is float | Press 1  Enter number: 1  Enter ID: 111  Enter score: s15 | s15 | **ERRO** | Program keep running | **Fail** |
| **4** | Input ID | ID is an integer | Press 1  Enter number: 1  Enter ID: 1234567891011 | 1234567891011 | **ERRO** | Program keep running but display wrong id | **Pass** |

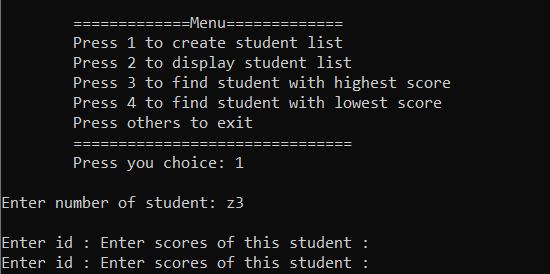
### 4.3.2 The console screen in the testing

Test case1:

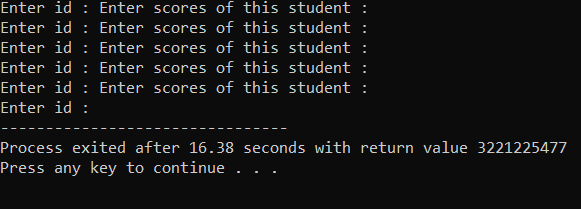


Figure

Test case 2:

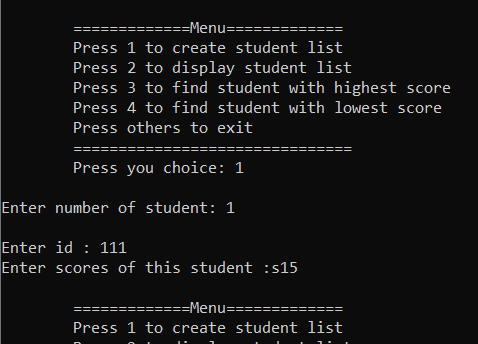


Figure



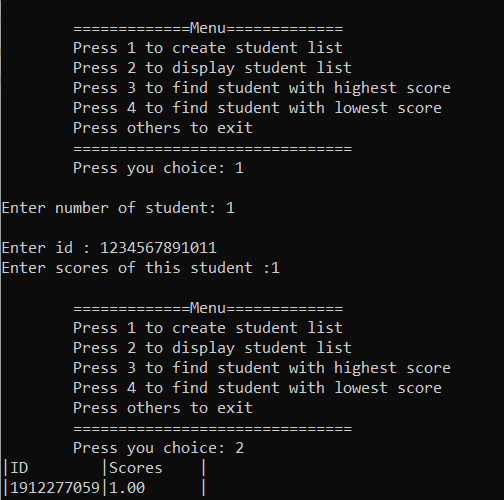
Figure

Test case 3:



Figure

Test case 4:



Figure

## 4.4 Review/ Evolution of this program

### 4.4.1 Advantages

Short, easy to understand and without many errors.

### 4.4.2 Disadvantages

A better layout should be presented, lacking in constraints for rolling input variables.

4.4.3 Directions for future processing and improvement

Add the conditions for the input variables. I have to research hard and try some other ways to come up with the best solution for this situation.

### 4.5 Lessons learned after doing the homework

A program can have many different directions, and I need to find the best path for my program. It takes hard work to improve my programming skills.

## EPILOGUE

Through lessons and practice, I realized the strengths and weaknesses of myself. Thereby improving his programming ability more. Understand how programming languages work and prepare yourself for the next semester.