Victor Gutierrez, Luis Garibay, Jose Morales

CIS 11

Project Documentation

**1 Introduction**

**1.1 Purpose**

* A program that is able to display the minimum, maximum, and average grade of five test scores. Requires the user to input five different test scores, these test scores will then be evaluated and sorted into minimum and maximum.

**1.2 Intended Audience and Users**

* This program could be useful for educational purposes of recording class average test scores of only five scores.

**1.3 Product Scope**

* The total average of the five test scores will be accounted for by adding all their values and then dividing them by five (the number of test scores). This should provide the user with the output of a Min, Max, Average, and letter grade for the inputs.

**1.4 Reference**

**2 Overall Description**

**2.1 Product Perspective**

* **Primary Program objectives**
  + Collect five test scores from the user.
  + Find min, max, and average value.
  + Display letter grade for average score.

**2.2 Product Functions**

* **Tasks**
  + Asks user for five test scores
  + Output min, max, average, and grade
* **Subtasks**
  + Store users inputs
  + For loop for min
  + For loop for max
  + Formula for average

**2.3 User Classes and Characteristics**

* **Students**
  + Luis Garibay, Victor Gutierrez, Jose Morales

**2.4 Operating Environment**

* This program should run on either Windows, Mac OS, or Linux
* As long as the operating system has the appropriate LC3 simulator to run the code.

**2.5 Design and Implementation Constraints**

* Access to the web is required if the user does not have an offline LC3 simulator.
* Preferably this program is designed for users with a novice experience in LC3.

**2.6 Assumptions and Dependencies**

* The program has not been finalized, everything is subject to change.
* User input may not input characters or special characters, only numeric characters may be entered.

**3 External Interface Requirements**

**3.1 User Interfaces**

* Users will interact with a basic font format console.
* Console will display to the user what input is needed.
* Output will provide the user with Max, MIn, and Average.

**3.2 Hardware interfaces**

* A PC or laptop could be used for the program.
  + Mac, Windows, or Linux

**3.3 Software Interfaces**

* For Windows and Linux:
  + LC3 simulator program
* For Mac and other OS operating systems:
  + Online LC3 simulator program

**3.4 Communications Interface**

* The program will require access to the web if the user plans on using an online LC3 compiler.
* Type of network connection does not matter, nor does the type of browser.

**4. Detailed Description of Functional Requirements**

**Pseudocode**

**Main:**

**1. Prompt user to enter a score value**

1. Set up a string to prompt “Enter the score value: “

2. Loop this five times until User has provided input for 5 Tests.

**2. Store the user input**

* Store the user test scores in an array.

**3. Go to subroutine for max value**

1. Find max value by for loop
   1. Loop array 5 times to cycle through array
   2. if : array[i]  >  max
   3. Then our max = array[i]
2. Return max value to print

**3. Display max score**

* Call a puts subroutine to print to screen

**4. Go to subroutine for min value**

1. Find max value by for loop
2. Loop array 5 times to cycle through array
3. if : array[i]  <  min
4. Then our min= array[i]

* Return min value to print

**5. Display min score**

* Call a puts subroutine to print to screen

**6. Go to subroutine for average value**

* Average = 5 / (array[0] + array[1] +array[2] +array[3] +array[4])
  + Return average score

**7. Display average test score**

* Call a puts subroutine to print to screen

**8. Go to subroutine for grade score**

* Pass average score to Grade Score subroutine
  + Return get grade score Letter

**9. Display grade score**

* Call a puts subroutine to print to screen

**11. Display end message**

* Call end message to end program

**12. .End**

**Subroutines**

* **Div X = X/Y \* Y + X (mod Y)**
  + Collect numerator (X) and denominator (Y)
* **GetScore**
  + Create an array that stores the users inputs for 5 test scores
* **AvgSc Total = 5 / (array[i++])**
  + Add all the values of the five test scores and divide by 5.
  + Return average score.
* **Print**
  + Puts
* **Prompt**
  + Print “ This program collects five test scores and provides you with the smallest, largest, and average score.\n”
  + “ provide a test score: ”
* **MinSub**
  + Collect iteration, counter amount, test scores of users
    - For (i= 0; i < tests; i++) {

If  ( array[i] < small) {

Small = array[i]; }}

* Return small
* **MaxSub**
  + Collect iteration, counter amount, test scores of users
    - For (i= 0; i < tests; i++) {

If  ( array[i] > large) {

Large = array[i]; }}

* Return large
* **LetGrade**
  + Collect the average score
  + Use branching to compare the average score with 90, 80, 70, 60.
  + Return letter grade.
* **TerminateMes**
  + Print “Goodbye”

**Flow Chart**

