Amtul Syeda      Sarah Welch      Demitrius De La Corte  
  
Professor Nguyen   
  
CIS-11-22530: Assembly Programming  
  
2022 May 29

**CIS-11 Course - Project Part 1: Documenting the Project**

1. **Introduction**

**1.1  Purpose**

The purpose of this program is to create an accessible way for the minimum, maximum, and average of five test scores to be calculated by students that are curious or anxious about their grades. The way the program will function is by providing a prompt to obtain inputs from the user, perform the necessary calculations, and display outputs on the console that are the minimum, maximum, and average of the five provided test scores (and corresponding letter grades).

The following requirements for the program were provided by Professor Nguyen:

1.       Contain appropriate addresses: origination, fill, array, input and output.

2.       Display minimum, max, average values/grades in console.

3.       Use appropriate labels and comments.

4.       Contain appropriate instructions for arithmetic, data movement and conditional operations.

5.       Comprise of 2 or more subroutines and implement subroutine calls.

6.       Use branching for control: conditional and iterative.

7.       Manage overflow and storage allocation.

8.       Manage stack: include PUSH-POP operation on stack.

9.     Include save-restore operations.

10.   Include pointer.

11.   Implement ASCII conversion operations

12.   Use appropriate system call directives.

**1.2  Intended Audience and Users**

The primary audience/users for this program is intended to be students that wish to calculate their test scores.

**1.3  Product Scope**

The intention of this program is to provide an easy way for students to calculate their test scores while also displaying the range and use of the LC-3 programming language.

**1.4  Reference**

**Source Documents for the Program Requirements and Specification**

1)      Team Sonikuu’s CIS-11 Final project (Used as a reference)

<https://github.com/Sonikuu/CIS11-Final/blob/master/Final.asm>

2)      Team KISMET-INC’s CIS-11 Final project (Used as a reference)

<https://github.com/KISMET-INC/Test_Score_Calculator>

**Companion Application Requirements Documents (If applicable)**

    None.

**2. Overall Description**

**2.1 Product Perspective**

 This program provides:

       An accessible calculator for students that includes:

- A way to calculate the maximum, minimum, and average grade of 5 test scores, which displays

as a corresponding letter grade.

- A way to showcase the function and utility of the LC-3 language to those that may be

interested.

- A potential resource that may be built upon and used to create future LC-3 programs at a later

date.

**2.2 Product Functions**

**The overall description of functionality:**

1. After the LC-3 simulator is downloaded, the user can utilize it along with Github to help run the program, as well as make any changes and share it as needed.
2. The program itself can provide an easy way to ease a student’s stress by allowing them to calculate the average, maximum, and minimum of five of their test scores and gain an understanding of what their final grade may be.

**Technical functionality:**

A configurable toolkit of functions including:

* The ability to calculate the average, maximum, and minimum of five test scores for a user.
* The ability to provide corresponding letter grades for the user’s given test scores.
* Provide a unique and accessible way to perform unique calculations in an LC-3 program.
* Provide a widely available potential resource to build upon to create future LC-3 programs.

**2.3  User Classes and Characteristics**

**Team: “Grade B” (Sarah Welch, Amtul Syeda, Demitrius De La Corte)**

- Responsible for the overall research, documentation, creation, implementation, testing, and

publication of the program.

- Must manage and schedule regular time for research, documentation, creation, implementation,

and testing of the program.

- Potentially use the program as a resource to build upon and create future LC-3 programs.

**Professor Kasey Nguyen**

- Responsible for review and grading of the submitted documentation for the program.

- Responsible for review and grading of the submitted program.

**Students**

- Use the program to obtain an accurate average, minimum, and maximum of 5 test scores.

- Use the program to obtain corresponding letter grades for their test scores.

- Potentially use the program as a resource to build upon and create future LC-3 programs.

**2.4 Operating Environment**

* The program is developed for use on the LC-3 simulator for Windows and Unix found at<https://highered.mheducation.com/sites/0072467509/student_view0/lc-3_simulator.html>
* This program is also developed for sharing purposes on Github (https://github.com/)

**2.5 Design and Implementation Constraints**

Use of LC-3 Simulator is required. Users can only input 5 test scores and the program can only give the closest letter grade equivalent to the score. One way this program could be improved is to exponentially increase the amount of test scores a user could take. There will also likely be areas within the program that could be later optimized to improve speed.

**2.6 Assumptions and Dependencies**

It is assumed that the user is familiar with how to use a keyboard, mouse, and monitor.

Due to the fact a simulator will have to be downloaded from the internet or used directly from an internet browser, it is also assumed that the user is familiar with how to use and interact with an internet browser, as well as have a decent connection to the internet.

**3. External Interface Requirements**

**3.1  User Interfaces**

The user will interface with the program by running it in an LC-3 simulator and interacting with the console. They can click using their mouse and input data using their keyboard in the required fields. Once all the information has been entered, the program will generate an accurate minimum, maximum, and average grade for the user as well as corresponding letter grades for the given input.

**3.2  Hardware Interfaces**

The user will only require access to any computer that has the necessary software and capability to download and run the LC-3 simulator. Devices such as laptops, desktops, and some tablets will likely work, while smartphones will likely be unable to run the LC-3 simulator.

**3.3  Software Interfaces**

The application will require Java 1.4 or newer (which is available for Windows, Linux, and Mac OS X), as that is the software required for the LC-3 simulator.

**3.4  Communications Interface**

The application will require access to the internet in order to download the LC-3 simulator, which is used to run the program. Once the simulator is downloaded, internet connection is no longer required to run the program. Any browser and network connection that allows access to the LC-3 simulator is fine.

**4. Detailed Description of Functional requirements**

**4.1 Type of Requirement (summarize from Section 2.2)**

Purpose: Provides a maximum, minimum, and average grade of 5 test scores provided by the user. Also provides a corresponding letter grade for each given test score.

Inputs: Inputs are provided by the user via mouse clicks and typing using a keyboard.

Processing: Inputs are validated after being inputted by the user into the LC-3 simulator console. The program will run and perform a variety of calculations to obtain the desired outputs. Once the calculations are finished, the program will display letter grades for each input as well as a maximum, minimum, and average of the five inputted test scores. If the inputs return as invalid, the program will instead display an error message.

Outputs: Displays the user’s provided test scores alongside corresponding letter grades. Will also display a maximum, minimum, and average of all given test scores. The console will show an error message if invalid input is detected.

Data: Final outputs are available to be saved by the user if desired.

**4.2 Performance requirements**

1. The application should be available to all users able to access Github and the downloadable LC-3 simulator.
2. Because the program is designed to display prompts and receive input, the time taken for the program to show prompts and perform the needed calculations will be dependent on the inputs that the program is given and the hardware running the simulator.
3. The program is expected to manage overflow and storage allocation.
4. The program should be capable of validating input through value comparison and know when to perform calculations and when to display an error message in the console.
5. The application performance is dependent on the number of inputs.

**4.3 Flow Chart and Pseudocode.**

Flow Chart can be found on the next page. Pseudocode can be found on the page after the Flow Chart.

**Diagram, Teams

Description automatically generated**

**Pseudocode:**

Display prompt

Input test score

Check if value is >100-90. If true, then grade corresponds to A

Check if value is 89-80. If true, then grade corresponds to B

Check if value is 79-70. If true, then grade corresponds to C

Check if value is 69-60. If true, then grade corresponds to D

Check if value is 59-0. If true, then grade corresponds to F

Check if value is negative or not a number. If true, print error message and halt the program

increment count by one and repeat if count is less than 5, or continue if count is equal to 5

Calculate minimum (Find lowest grade number)

Calculate maximum (Find highest grade number)

Calculate average ([score 1 + score 2 + score 3 + score 4 + score 5] / 5)

Display grade scores 1-5 with corresponding letter grades

Display Minimum, Maximum, and Average of the five test scores

**CIS-11 Team Project Schedule:**

**Week 1: Wednesday, May 11th to Sunday**

First meeting with team to create a schedule and pick project

Utilize trello and google doc to complete part 1

Make sure to delegate tasks among team members

-  
**Week 2: Monday, May 12th to Sunday, May 22nd**

Research information to add onto our documentation

Set aside time to play with code from the projects we found online

-

**Week 3: Monday, May 23rd to Sunday, May 29th**

Finalize our documentation

2 meetings at least this week

Ask questions about documentation during lab hours

Submit Part 1: Planning and Documenting Project

-  
**Week 4: Monday, May 30th to Tuesday, June 7th**

Make sure to delegate tasks among team members

Create a Github account

Ask questions during lab hours if needed

Meet at least 3 times this week to work on LC-3 program

Create a Github repo for our program

Submit Part 2: Building LC-3 program

Submit Part 3: Self, Team Evaluation and Github