# CIS11 Course Project Part 1: Documenting the Project

Fill in the following areas (purple).

Team 2 – Musamim Mubtakir and Joshua Sepulveda

**Introduction**

* 1. **Purpose**

The purpose of this project is to create a LC-3 program that displays the minimum, maximum, and average grade of 5 test scores and display the letter grade associated with the test scores.

* 1. **Intended Audience and Users**

Students/professors who want the minimum, maximum, and average grade of 5 test scores and display the letter grade associated with the test scores

* 1. **Product Scope**

What is the intention of this program?

The intention of this project is to test 5 different scores and give a letter grade.

* 1. **Reference**

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

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.
13. Testing Test the program using the below values

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

What other documents should be reviewed with this document?

1. CIS Course Project Part 1
2. CIS Course Project Part 2
3. CIS Course Project Part 3

**2. Overall Description**

**2.1 Product Perspective**

Primary program objectives

To find minimum, maximum, and average grade of 5 test scores and display the letter grade associated with the test scores

* 1. **Product Functions**

**The overall description of functionality:**

Highlight the program functionality: Identify tasks and subtasks of the program in summary.

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.
13. Testing Test the program using the below values
    1. **User Classes and Characteristics**

**Who are involved in this development process? Include business and technical personnel and their tasks.**

**Musamim Mubtakir: LC-3 Coding, pseudocode, I/O, Debugging**

**Joshua Sepulveda: LC-3 Coding, pseudocode, I/O, Documentation**

* 1. **Operating Environment**

What type of system will the application be operated on? Operating system? System types? Development platform?

No Specific Operating environment, LC-3 simulator is sufficient.

* 1. **Design and Implementation Constraints**

Note any constraints or limitation to the application.

Depending on the operating system, access to the web is required since there is no LC-3 simulator for Mac OS.

* 1. **Assumptions and Dependencies**

Note any dependencies

None

***3*. External Interface Requirements**

* 1. **User Interfaces**

How will the user interface with your program? Menus? Access prompt? Links? Icons?

Menus and prompts to help the user navigate the LC-3 Program.

PROMPT .STRINGZ "Enter 5 test scores [0 – 99]: "

* 1. **Hardware Interfaces**

Any computer for the online LC-3 Simulator. No restrictions.

* 1. **Software Interfaces**

Specify additional software interface – if any. What type of software will the application require to run?

Web browser.

Does your application require web, Internet or network connectivity? If so, which browser? What type of network connection?

Access to a web browser and the internet is required. Notepad app also required to save unfinished LC-3 code.

**4. Detailed Description of Functional requirements**

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

**What are the functions? Their purposes? Inputs? Outputs? Data? Where is the data stored (internal or external to the application)?**

**Input:** User is prompt to input the test scores.

**Output:** Display **maximum, minimum, average scores** **and letter grade equivalence** (0 – 50 = F, 60 – 69 = D, 70 – 79 = C, 80 – 89 = B, 90 – 100 = A) on the console.

**4.2 Performance requirements  
 What is the expected performance level of the program?**

High level with no bugs.

**4.3 Flow Chart and Pseudocode.**

The program assigns a grade to a correlated score

An example of the pseudocode uses is seen below for A

A\_GRADE LD R0, A\_NUM ; LOAD NUMBER VALUE

LD R1, A\_LET ; LOAD SYMBOL VALUE

ADD R2, R3, R0 ; COMPARE INPUT TO VALUE OF GRADE

BRzp STR\_GRADE ; IF POS OR ZERO STORE GRADE

|  |
| --- |
|  |
|  |  |

If score is invalid the program will let the user know and error has occurred

User inputs score

If score is valid user will get a grade of associated with the score. The grade the user gets is an A, B, C D, or F

Program also calculates the average, above average, and minimum score among all students an example for max is below:

; CALCULATE MAX

CALCULATE\_MAX

LD R1, NUM\_TESTS ; R1 HOLDS THE TOTAL NUMBER OF TEST

LEA R2, GRADES ; R2 HOLDS THE STARTING ADDRESS OF GRADES

LD R4, GRADES

ST R4, MAX\_GRADE

ADD R2, R2, #1

LOOP1 LDR R5, R2, #0 ; ACCESS POINTER VALUE IN GRADES

NOT R4, R4

ADD R4, R4, #1

ADD R5, R5, R4

BRp NEXT1

LEA R0, MAX

PUTS

LD R3, MAX\_GRADE

AND R1, R1, #0

JSR BREAK\_INT

LD R0, SPACE

OUT

LD R0, NEWLINE

OUT

JSR CLEAR\_REG

|  |
| --- |
|  |
|  |  |