**UNIVERSITI TUNKU ABDUL RAHMAN**

**LEE KONG CHIAN FACULTY OF ENGINEERING AND SCIENCE**

**UECS1013 INTRODUCTION TO COMPUTER ORGANIZATION AND ARCHITECTURE**

**ASSIGNMENTS**

**February 2025 Trimester**

|  |  |  |  |
| --- | --- | --- | --- |
| **NO.** | **STUDENT NAME** | **STUDENT ID** | **Tutorial/ Practical Group** |
| **1.** | Koh Khai Jeck | 2304740 | **T4/P9** |
| **2.** | Edmund Chan Chee An | 2202241 | **T9/P7** |
| **3.** | Leon Siow Yi Hong | 2204403 | **T9/P7** |
| **4.** | Quak Jing | 2205378 | **T7/P7** |

# **Table of Content**

[**Table of Content** 2](#_Toc197561248)

[**Flowchart** 3](#_Toc197561249)

[**Complete Code** 7](#_Toc197561250)

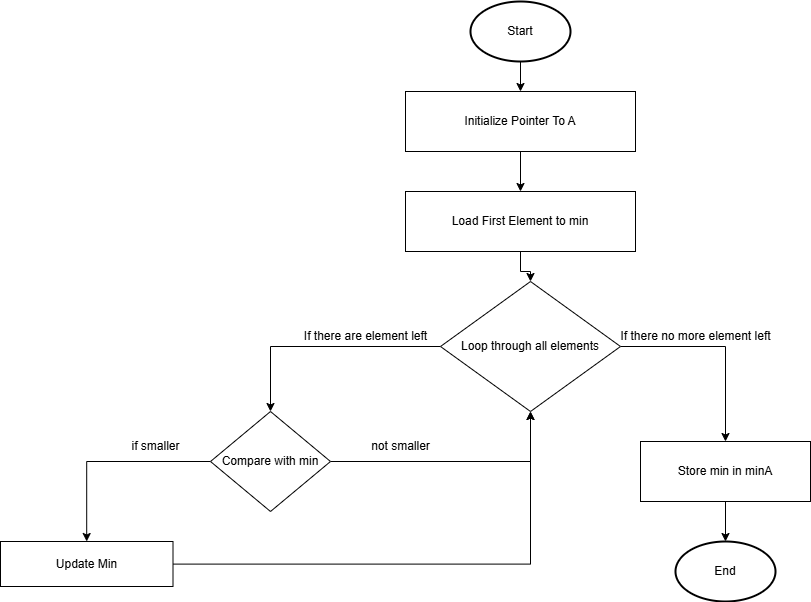
[**Operations and Expected Results** 10](#_Toc197561251)

[**Results and final values** 12](#_Toc197561252)

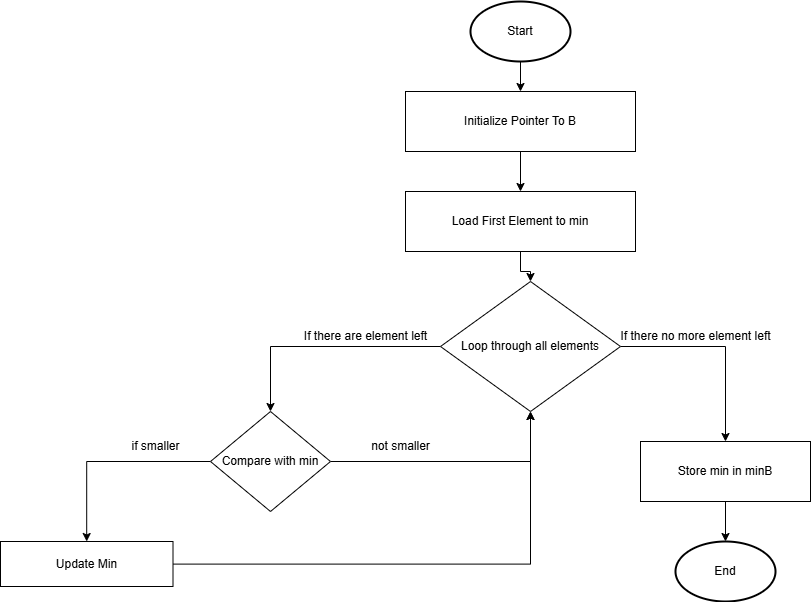
# 

# **Flowchart**

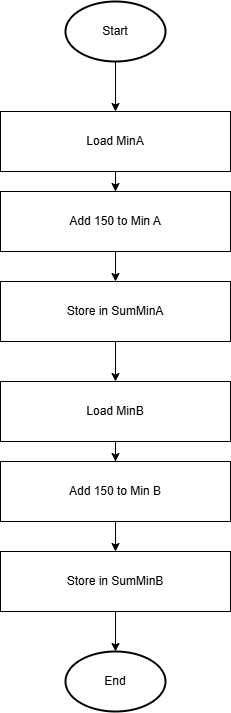
1. Find minimum value in A



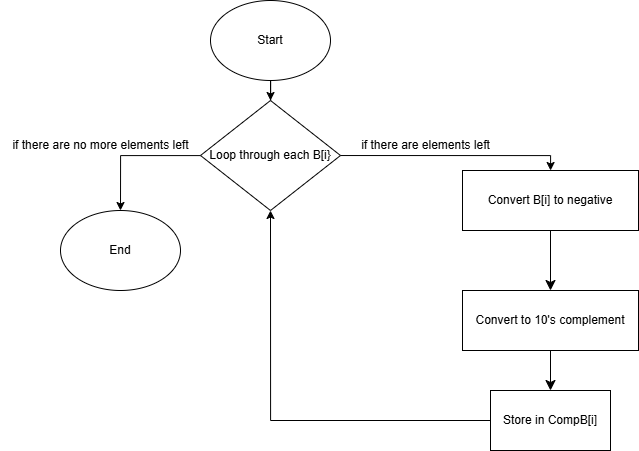
1. Find minimum value in B



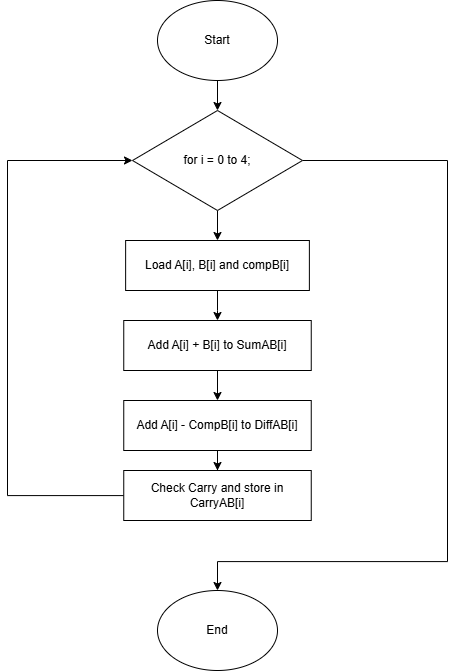
1. Add 150 to minimum A and minimum B



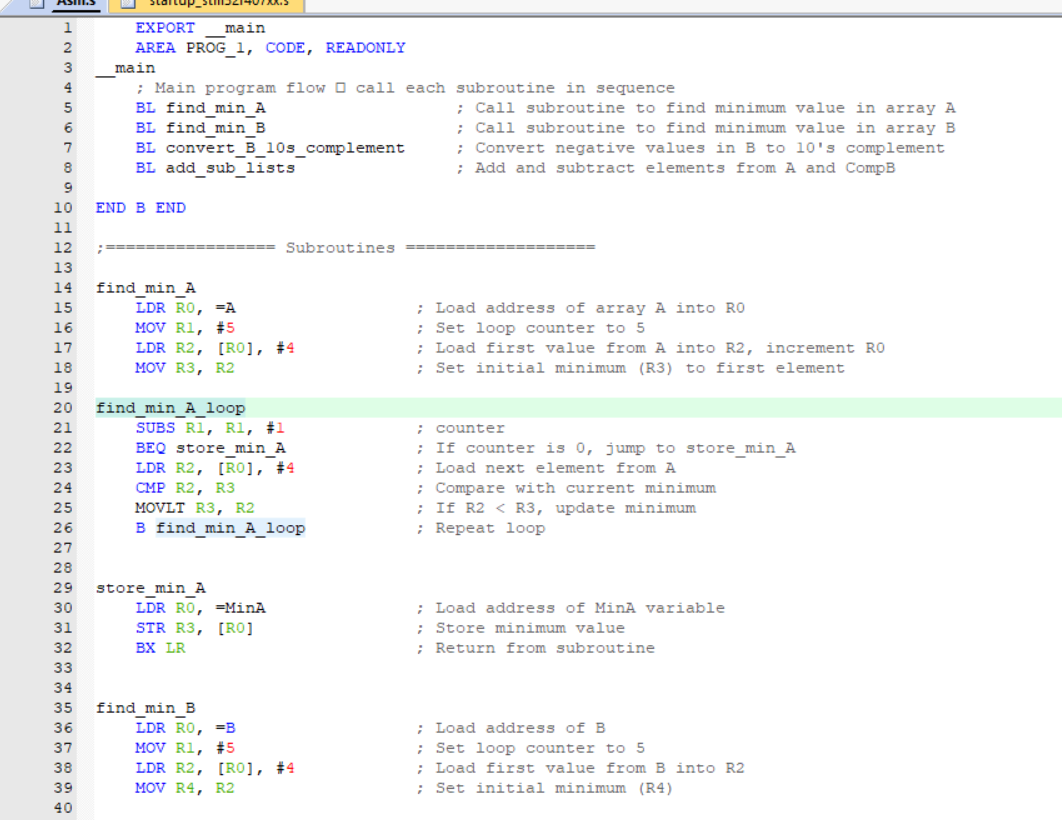
1. Convert List B to 10’s Complement

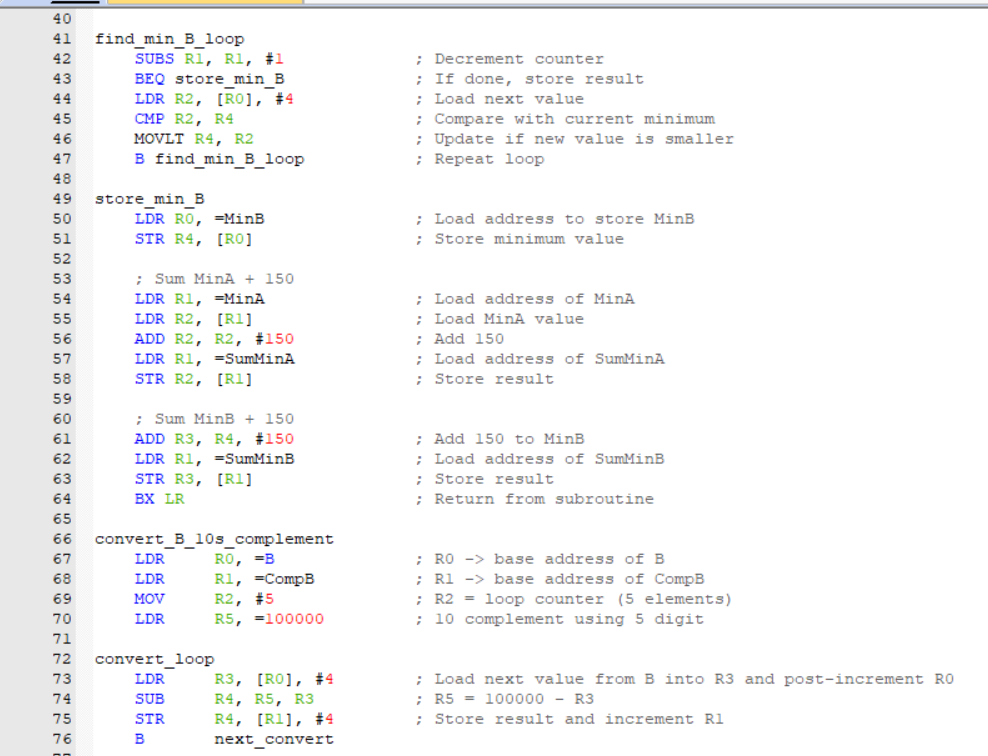
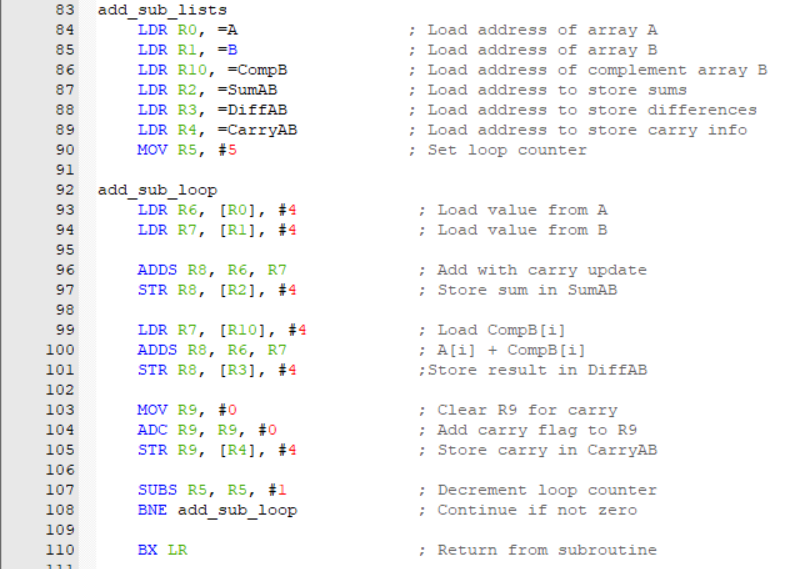


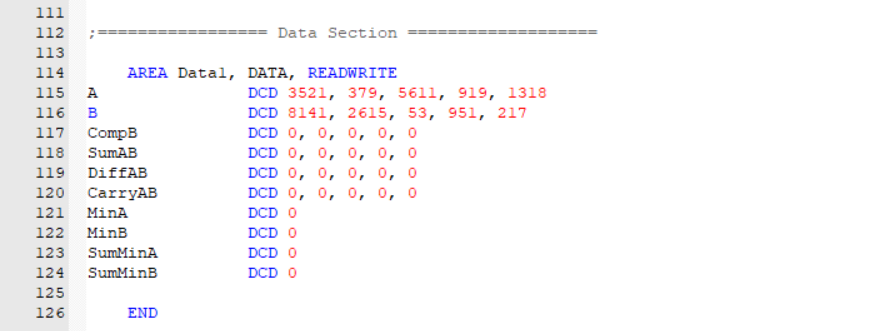
1. Add A[i] + B[i], A[i] - B[i] (10’s complement) and Detect Carry



# **Complete Code**





# **Operations and Expected Results**

1) Find Minimum Value in List A:

* Load the first value of A into R3 as the initial minimum value
* Loop through the remaining elements of A
* Compare each element with the current minimum value
* If smaller value is found, update the minimum to new minimum value
* Store minimum value in MinA

Expected Output:

List A = {3521, 379, 5611, 919, 1318}

MinA = 379

2) Find Minimum Value in List B:

* Load the first value of B into R4 as the initial minimum value
* Loop through the remaining elements of B
* Compare each element with the current minimum value
* If smaller value is found, update the minimum to new minimum value
* Store minimum value in MinB

Expected Output:

List B = {8141, 2615, 53, 951, 217}

MinB = 53

3) Adding 150 to MinA and MinB

* Load MinA and add 150
* Store result in SumMinA
* Load MinB and add150
* Store result in SumMinB

ExpectedOutput:

SumMinA = 379 + 150 = 529

SumMinB = 53 + 150 = 203

4) Converting List B to 10’s Complement

Assume List B = {-8141, -2615, -53, -951, -217}

* Loop through each element in B
* Convert list B to negative value
* Convert to 10’s complement using this formula: 10’s Complement of B[i]=(−B[i])+100000
* Store result in CompB

Expected Output:

CompB = {91859, 97385, 99947, 99049, 99783}

5) Addition Subtraction and Carry detection:

* For each index i from 0 to 4:
* Load A[i], B[i] and CompB[i]
* Perform SumAB[i] = A[i] - B[i]
* Perform DiffAB[i] = A[i] + compB[i]
* Find and store carry for each result

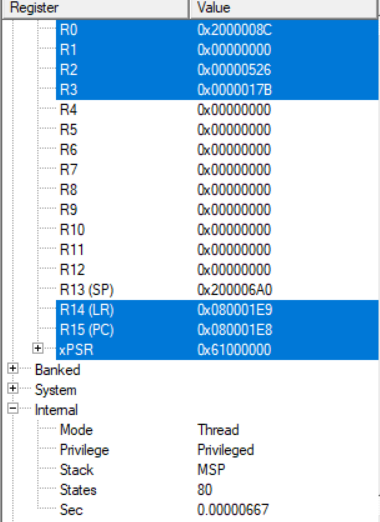
Expected Output

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| i | A[i] | B[i] | CompB[i] | SumAB[i] | DiffAB[i] | Carry |
| 0 | 3521 | 8141 | 91859 | 11662 | 95380 | 0 |
| 1 | 379 | 2615 | 97385 | 2994 | 97764 | 0 |
| 2 | 5611 | 53 | 99947 | 5664 | 10558 | 1 |
| 3 | 919 | 951 | 99049 | 1870 | 99968 | 0 |
| 4 | 1318 | 217 | 99783 | 1535 | 101101 | 1 |

# **Results and final values**

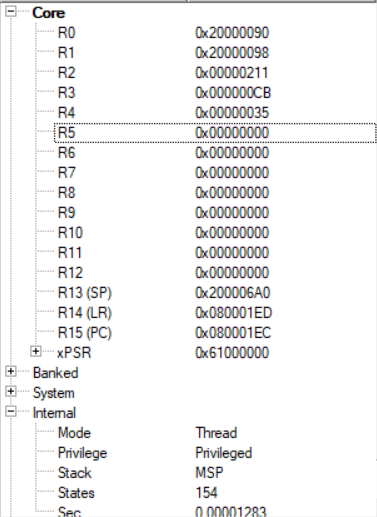
1. Find minimum value in list A

* R0 is the address pointing to MinA
* R1 is the loop counter
* R2 is the last loaded value from the list
* R3 is the final minimum value = 379



1. Find minimum value in list B and add 150 to both minimum value of A and B

* R0 pointing to a variable
* R1 is the address of SumMinA
* R2 is the value of minimum A + 150 = 529
* R3 is the value of minimum B + 150 = 203
* R4 is the minimum value of list B = 53



1. Convert list B to 10’s complement assuming there are negative values

Assume list B = {8141, 2615, -53, 951, -217}

* R1 is pointer to CompB

A screenshot of a computer

AI-generated content may be incorrect.

* CompB = {000266D3, 00017C69, 0001866B, 000182E9, 000185C7}
* CompB = {91859, 97385, 99947, 99049, 99783}

A screenshot of a computer

AI-generated content may be incorrect.

1. Addition Subtraction and Carry detection:

Assume list B = {8141, 2615, -53, 951, -217}

* R2 is address of SumAB
* R3 is address of DiffAB
* R4 is to store info of carry

A screenshot of a computer

AI-generated content may be incorrect.

SumAB = {00002D8E, 00000BB2, 00001620, 0000074E, 000005FF}

= {11662, 2994, 15558, 1870, 11101}

A screenshot of a computer

AI-generated content may be incorrect.

DiffAB = {00017494, 00017DE4, 00019C56, F00018680, 0000018A}

DiffAB = {95380, 97764, 105558, 99968, 101101}

A white rectangular object with numbers

AI-generated content may be incorrect.