# Worksheet 4 Assembly language

**Task 1**

The instruction set for the Little Man Computer (LMC) is shown in the Table below.

|  |  |  |  |
| --- | --- | --- | --- |
| **Mnemonic code** | **Instruction** | **Numeric code** | **Description** |
| ADD | ADD | 1xx | Add the contents of the memory address to the Accumulator |
| SUB | SUBTRACT | 2xx | Subtract the contents of the memory address from the Accumulator |
| STA | STORE | 3xx | Store the value in the Accumulator in the memory address given. |
| LDA | LOAD | 5xx | Load the Accumulator with the contents of the memory address given |
| BRA | BRANCH (unconditional) | 6xx | Branch - use the address given as the address of the next instruction |
| BRZ | BRANCH IF ZERO (conditional) | 7xx | Branch to the address given if the Accumulator is zero |
| BRP | BRANCH IF POSITIVE (conditional) | 8xx | Branch to the address given if the Accumulator is zero or positive |
| INP | INPUT | 901 | Input into the accumulator |
| OUT | OUTPUT | 902 | Output contents of accumulator |
| HLT | Halt | 0 | Stops the execution of the program. |
| DAT | DATA |  | Used to indicate a location that contains data. |

1. The assembly language program below is written using the LMC instruction set.

INP

STA x

INP

STA y

INP

ADD x

SUB y

OUT

X DAT

y DAT

State what the output is when the user inputs 5, 7 and 13

11

2. The assembly language program below is written using the LMC instruction set.

INP input n=1 to the accumulator

STA n store value in n

INP input 6 to accumulator

STA x store in x

STA z store in z

INP input 4 to the accumulator

SUB n subtract 1 leaving 3 in accumulator

SUB n subtract 1 leaving 2 in accumulator

STA y store 2 in y

loop LDA x load x into accumulator

ADD z add 6 to accumulator

STA x store in x

LDA y load y

SUB n subtract n

STA y store in y

BRP loop branch if accumulator is positive or zero to loop

LDA x load x

OUT output contents of accumulator

n DAT

x DAT

y DAT

z DAT

HLT

(a) State what the output is when the user inputs 1, 6, 4.

Use the trace table below to help you.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **acc** | **n** | **x** | **z** | **y** |
| 1 | 1 |  |  |  |
| 6 |  | 6 | 6 |  |
| 4 | 1 | 6 | 6 |  |
| 3 | 1 | 6 | 6 |  |
| 2 | 1 | 6 | 6 |  |
| 2 | 1 | 6 | 6 | 2 |
| 6 | 1 | 6 | 6 | 2 |
| 12 | 1 | 6 | 6 | 2 |
| 12 | 1 | 12 | 6 | 2 |
| 6 | 1 | 12 | 6 | 2 |
| 5 | 1 | 12 | 6 | 2 |
| 12 | 1 | 12 | 6 | 2 |
| 18 | 1 | 12 | 6 | 2 |
| 18 | 1 | 18 | 6 | 2 |
| 2 | 1 | 18 | 6 | 2 |
| 1 | 1 | 18 | 6 | 2 |
| 1 | 1 | 18 | 6 | 1 |
| 18 |  |  |  |  |
| 24 |  |  |  |  |

(b) State the purpose of the program.

3. Write an assembly code program to input two numbers x and y and output the maximum.

**Extension Task**

Run all the programs in Task 1 on the LMC computer at <http://peterhigginson.co.uk/LMC/>

**Task 2**

4. The contents of memory cells 51-58 are as shown in the table below.

|  |  |
| --- | --- |
| **Memory location** | **Contents** |
| 51 | 2 |
| 52 | 55 |
| 53 | 51 |
| 54 | 14 |
| 55 | 20 |
| 56 | 3 |
| 57 | 52 |
| 58 | 53 |

*Table 1*

State what will be the contents of the accumulator after each of the following operations:

(a) load immediate 53 Answer:

(b) load direct 53 Answer:

(c) load indirect 53 Answer:

(d) load indexed R0 (Assume R0 contains 3 and the Index register contains 50)

Answer: