Practical 03

Assembly Language

THIS IS A PROCTORED PRACTICAL

YOU MUST SHARE YOUR SCREEN SO YOUR PARTICIPATION IN THIS PRACTICAL CAN FULLY INVIGILATED

1. Create a Github repository "Assembly_and_C"
2. Create a sub directory PRACTICAL_##
3. Add Github link to CA Spreadsheet
   e.g https://STUDENTID.github.com/Assembly_and_c/PRACTICAL_##
4. Invite Lab Supervisors including **MuddyGames** as a collaborators
5. Go to designated group to complete practical
6. Upload completed Practical files to Github repository

NOTE: Use of EASy68K editor and emulator allowed, use of internet allowed, use of slide deck(s) allowed. Installer located here http://www.easy68k.com/

Create a unique file ***e.g. practical_##_part#.X68*** for each practical section below.

Practical 03

Assembly Language

Objective Understand and utilise Address Registers**:**

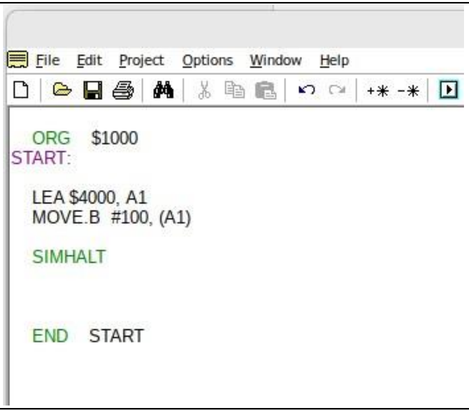
1

Create a new 68K project and name the file ***practical_03_part1.X68***

Edit compile and execute the code across and observe while debugging and contents of memory.

Examine and note contents of address registers and memory.

Review questions, what do the address register mean and what is stored in memory and why?



```
  ORG   $1000
START:

  LEA $4000, A1
  MOVE.B  #100, (A1)

  SIMHALT


  END   START
```

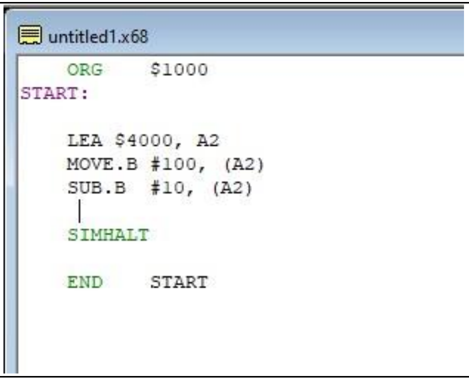
2

Create a new 68K project and name the file ***practical_03_part2.X68***

Edit compile and execute the code across and observe while debugging and contents of memory.

Examine and note contents of address registers and memory.

Review questions, what do the address register mean and what is stored in memory and why?



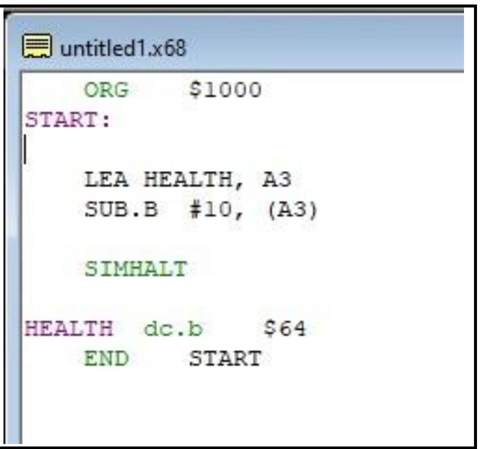
```
    ORG     $1000
START:

    LEA $4000, A2
    MOVE.B #100,  (A2)
    SUB.B  #10,  (A2)

    SIMHALT

    END     START
```

practical_03_part3.X68

Edit compile and execute the code across and observe while debugging and contents of memory.

Examine and note contents of address registers and memory.

Review questions, what does the Health declaration mean and what is stored in memory and why?



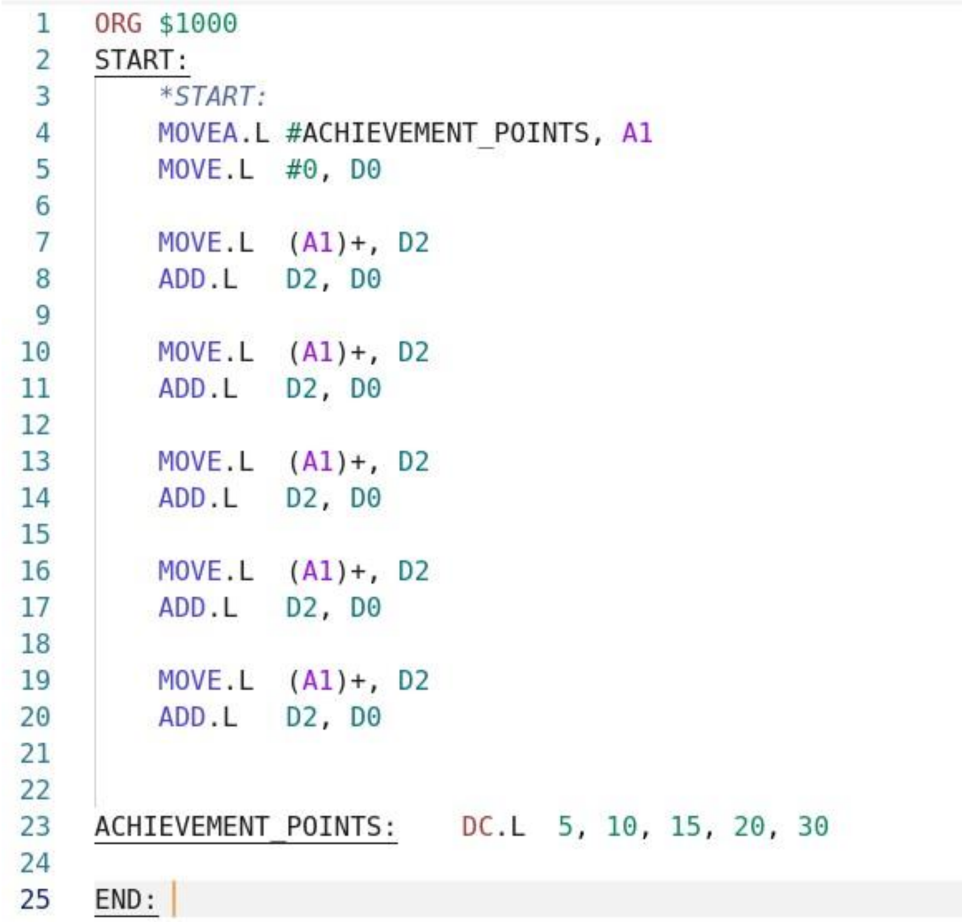
```
    ORG     $1000
START:

    LEA HEALTH, A3
    SUB.B  #10,  (A3)

    SIMHALT

HEALTH  dc.b    $64
    END     START
```

3 Create a new 68K project and name the file

4	Create a new 68K project and designate the file as ***practical_03_part4.X68***. Perform tasks such as editing, compiling, and executing the code according to the **Specification**. During debugging, closely monitor the contents of the memory. Inspect the stored values in memory and adjust values and their locations within the memory.	**4 Specification**: Complete the following data to memory operations using Address Registers. MOVE.L MOVE.B MOVE.W Declaring initial data such as Player Health at 100% or $64 (Hex equivalent) Use your own examples, such as typical game data; • Player Points • Player Health • Player X and Y Position • Boss Health • Boss X and Y Position Your solution should include at least **12 examples** for the above.
5	Create a new 68K project and designate the file as ***practical_03_part5.X68***. This is an example of an **Array** and ***traversing an Array***.	

Practical 03

Assembly Language

	
	ORG $1000
	START:
	    *START:
	    MOVEA.L #ACHIEVEMENT_POINTS, A1
	    MOVE.L #0, D0
	
	    MOVE.L (A1)+, D2
	    ADD.L D2, D0
	
	    MOVE.L (A1)+, D2
	    ADD.L D2, D0
	
	    MOVE.L (A1)+, D2
	    ADD.L D2, D0
	
	    MOVE.L (A1)+, D2
	    ADD.L D2, D0
	
	    MOVE.L (A1)+, D2
	    ADD.L D2, D0
	
	ACHIEVEMENT_POINTS: DC.L 5, 10, 15, 20, 30
	
	END:

6	Complete Practical Quiz which will be provided by Lab Supervisor

Demonstrate completed assembly files at the end of the LAB and ensure it has been checked

Student Name	**Brandon Jaroszczak**	**Student Number**	**C00296052**
Date	**27/1/2025**	**Checked**	