Lab 3 - Registers

Dr. Donald Davendra
CS311 - Computer Architecture 1

October 19, 2024

The third laboratory exercise requires you to assign the contents of two arrays in **nasm** and calculate the result based on the following equation (1):

$$\sum_{i=0}^{N-1} ((-3 + a_i) + (b_i - 14)) \tag{1}$$

where N is the size of the arrays a and b.

Create a file named `array.asm` in ebe.

Question 1 - `.data` section.

You are required to assign two array's in the `.data` segment as the following:

- label - `a`, `b`
- contents - `a` = {-4, 22, 144} and `b` = {-3, -16, 12}
- size - `a` (**word**) and `b` (**byte**)

The `segment .data` is given as:

```
      segment .data
a           dw   -4, 22, 144      ; array of 3 values
b           db   -3, -16, 12      ; array of 3 values
result      dq    0               ; memory to result
```

You can declare other variables as you deem necessary to solve this assignment.

Question 2 - `.text` section.

Start the text segment as the following:

```
      segment .text
      global  main
main:
```

Question 3 - `global main` section.

The task in the main section is to **explicitly** follow the equation and iteratively add -3 to the indexed value in array ***a***, subtract the indexed value in array ***b*** by 14 and finally add these two parts and store the resulting value in memory location ***result***.

You are allowed to use a maximum of **three general purpose registers** in this lab. You are NOT allowed to change any values in the memory locations of ***a*** and ***b***. Some of the opcodes of use in this lab are:

- `mov` - moving data from register-register, register-variable etc
- `lea` - loading effective address of a variable to a register.
- `add` - adding two values in registers or in variables.
- `sub` - subtract two values in registers or in variables

Upon completion of the task, zero out **all** used registers and return. This following can be taken as an example:

```
      ...                    ; your code
      xor     rax, rax       ; zero out rax
      ret
```

Submission

All submitted files **MUST** have the **student name**, **student CWU ID** and the **honor code**.

The file must be submitted through Canvas before **5pm October 25, 2024**. The grading rubric is given in Table 1.

Table 1: Grading rubric

File	Aspects	Points
`array.asm`	Compiles	5
	Correct equation interpretation	40
	Correct use of registers	15
	Correct use of memory offsets/addressing	25
	Documentation/commenting	15