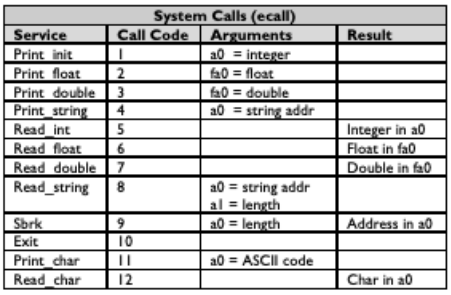
Lab Session 1.
AssemblyPprogramming Exercisses

The objective of this session is to familiarize yourself with the CREATOR simulator, which is a simulator that allows you to run programs that use the RISC-V assembler. During this first session the simulator will be used to execute the programs proposed below. More information about this simulator is available in Aula Global.

Exercise 1. Edit and run the following program in the CREATOR simulator (available in Aula Global).

```
.data
      msg: .string "My first program"

.text

.globl main

main:

    li a7, 1
    la a0, msg
    ecall

    jr ra
```



System Calls (ecall)			
Service	**Call Code**	**Arguments**	**Result**
Print_init	1	a0 = integer	
Print_float	2	fa0 = float	
Print_double	3	fa0 = double	
Print_string	4	a0 = string addr	
Read_int	5		Integer in a0
Read_float	6		Float in fa0
Read_double	7		Double in fa0

Read_string	8	a0 = string addr
		a1 = length	

Sbrk	9	a0 = length	Address in a0
Exit	10		
Print_char	11	a0 = ASCII code	
Read_char	12		Char in a0

Exercise 2. Given the following fragment in a high level language:

```
int a = 6;
int b = 7;
int c = 3;
int d;

d = (a+b) * (a+b);
```

Write a code in RISC-V assembler that allows you to calculate the previous expression. The result must be stored in the t5 register.

Exercise 3. Write a program in assembler RISC-V to calculate the sum of the first 100 natural numbers. The program must leave the result in register a0.

Exercise 4. Write a program that reads two integer numbers A and B and displays if one of them is a multiple of the other.

Exercise 5. Write an assembly program of the RISC-V that reads an N number and displays the following on the screen:

```
1
1 2
1 2 3
1 2 3 4
…..
1 2 3 4 5 …. N
```

Exercise 6. Write an assembly program that reads two integer numbers. The program must print the largest of them.

Exercise 7. Write a program in RISC-V assembler that reads a number and indicates whether the number is odd or even.

Ejercicio 8. Given the following program fragment

```
.data
            a: .word 5
            b: .word 10
.text

main:
            li    t0  1
            lw    t1, a
            lw    t2, b
  label1:   bgt   t0,  t1,   label2
            addi  t2, t2, 2
            addi  t0, t0, 1
            j     label1
  label2:   sw    t0, a
            sw    t2,  b
```

Indicate the value of the records $t0, $t1 and $t2 and the memory locations a and b at the end of program execution.

Exercise 9. Write a program using the MIPS32 assembler, which makes the sum of the squares of a series of numbers entered by the keyboard. To do this, the program will first ask for the quantity of numbers to read. Next, you will read these numbers, calculate the corresponding square, and finally print the resulting sum.