

Práctica 04

DOCENTE	CARRERA	CURSO
MSc. Vicente Enrique Machaca Arceda	Escuela Profesional de Ingeniería de Software	Fundamentos de Lenguajes de Programación

PRÁCTICA	TEMA	DURACIÓN
04	Ensamblador	3 horas

1. Datos de los estudiantes

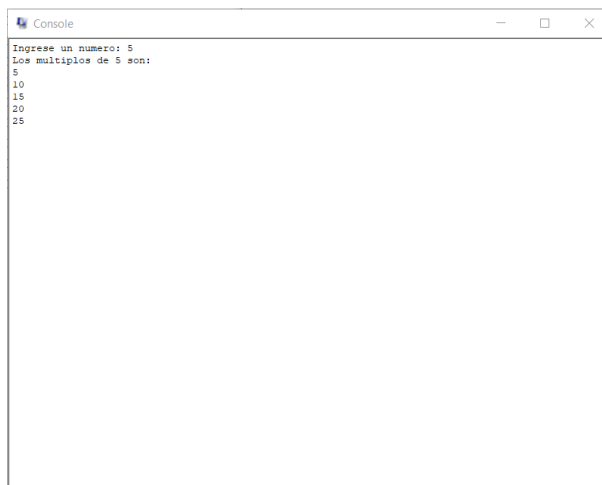
- GIT: GIT-Repo
- Integrantes:
 - Roberto Heredia Garland

2. Ejercicios

1. Cree un array de 5 elementos con múltiplos de un número ingresado por teclado, luego debe mostrar dicho array por pantalla.

```
1
2 ..data
3     txt1: .asciiz "Ingrese un numero: "
4     txt2: .asciiz "Los multiples de "
5     txt3: .asciiz " son:"
6     arr: .word 1, 2, 3, 4, 5
7     txt4: .asciiz "\n"
8 .text
9 main:
10     li $v0,4
11     la $a0, txt1
12     syscall
13
14     li $v0,5
15     syscall
16
17     move $t3, $v0
18
19
20     li $v0,4
21     la $a0, txt2
22     syscall
23
24     move $a0, $t3
25
26     li $v0,1
```

```
27     syscall
28
29     li $v0,4
30     la $a0, txt3
31     syscall
32
33     li $t0, 0
34     la $t1, arr
35
36     loop1:
37         bge      $t0, 5, exit
38
39         lw       $t2, 0($t1)
40         add      $t1, $t1, 4
41
42         mul      $t6, $t2, $t3
43         li $v0,4
44         la $a0, txt4
45         syscall
46
47         li      $v0, 1
48         move    $a0, $t6
49         syscall
50
51         li      $a0, 32
52         li      $v0, 11
53         syscall
54
55         add     $t0, $t0, 1
56         j      loop1
57     exit:
58
59     jr $ra
```



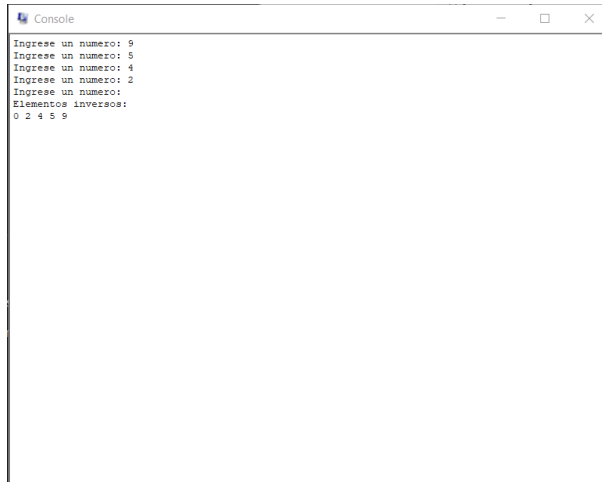
```
Console
Ingrese un numero: 5
Los multiplos de 5 son:
5
10
15
20
25
```

2. Cree un programa que solicite 5 elementos por teclado y los almacene en un Array. Luego se debe mostrar estos elementos en orden inverso.

```
1  .data
2      txt1: .asciiz "Ingrese un numero: "
3      txt2: .asciiz "Elementos inversos:\n"
4      arr: .word 1:5
5
6  .text
7  main:
8      li $t0, 0
9      li $t2, 0
10     la $t1, arr
11  loop1:
12     bge $t0, 5, end_loop1
13
14     li $v0, 4
15     la $a0, txt1
16     syscall
17
18     li $v0, 5
19     syscall
20     move $t2, $v0
21
22     sw $t2, 0($t1)
23     add $t1, $t1, 4
24     add $t0, $t0, 1
25     j loop1
26
27  end_loop1:
28
29     sub $t1, $t1, 4
30     li $t0, 0
31
32     li $v0, 4
33     la $a0, txt2
34     syscall
35
36  loop2:
37     bge $t0, 5, end_loop2
38     lw $t2, 0($t1)
39     sub $t1, $t1, 4
40
41     li $v0, 1
42     move $a0, $t2
43     syscall
44
45     li $a0, 32
46
47     li $v0, 11
48     syscall
49     add $t0, $t0, 1 #i++
50
51     j loop2
52
53  end_loop2:
54
```

55

jr \$ra



```
Console
Ingrese un numero: 9
Ingrese un numero: 5
Ingrese un numero: 4
Ingrese un numero: 2
Ingrese un numero:
Elementos inversos:
0 2 4 5 9
```