

# Ejercitación Risc-V

Sistemas Digitales

6 de junio de 2024

1. Dado vector de enteros *arreglo* y su *Longitud*, escribir un programa que encuentre el valor máximo en *arreglo*.

Ejemplo:

- Entrada: Arreglo = [3, 1, 4, 1, 5, 9, 2, 6], Longitud = 8
- Salida: 9

2. Sean dos vectores *s* y *q* tal que las direcciones de inicio vienen en *a0* y *a1* respectivamente, a su vez *a2* contiene el tamaño de *s* y *q*. Se pide copiar la información de *q* a *s*.
3. Sean dos vectores *s* y *q* tal que las direcciones de inicio vienen en *a0* y *a1* respectivamente, además se cuenta con *a2* que contiene el tamaño de *s* y *q*. Se pide copiar los elementos pares de *q* a *s* y si no 0.
4. Dado un vector *array* que contiene *n* números enteros ordenados de menor a mayor y un *target* que es un valor entero, se pide devolver el índice de *array* adonde se encuentra *target*, usando búsqueda binaria.

#BINARY SEARCH

```
.data
array: .word 1, 3, 5, 7, 9, 11, 13, 15, 17, 19    # Sorted array
target: .word 1                                  # Target value to find
n: .word 10                                       # Number of elements in the array
```

5. Hacer un programa que calcule `Fibonacci(5)`

```
# Fibonacci(5) = Fibonacci(4) + Fibonacci(3)
# ...
# Fibonacci(1) = 1
# Fibonacci(0) = 0
```

6. a. Escribir la función `multiplicacion`, que dados dos números *a0* y *a1* y devuelva el resultado de *a0* x *a1* en *a0*.  
b. Utilizar la función anterior para armar la función `factorial` (La implementación de `factorial` tiene que ser recursiva manteniendo las convenciones de llamados en riscv)  
c. ¿Qué problema presentan estas implementaciones al devolver el resultado en un único registro? ¿Con qué entradas falla?  
d. ¿Qué tamaño de memoria necesitaría nuestra máquina para correr el programa `factorial(3)` asumiendo que el programa ocupa *X* bytes?  
e. Si la implementación fuera iterativa, ¿requerirá más o menos memoria?
7. Sumar los *n* primeros números naturales

```
# n_sum(n) = sum_{i=0}^n i

# n_sum(n) = n + n_sum(n-1)
# n_sum(n) = n + n-1 + n_sum(n-2)
# n_sum(n) = n + n-1 + ... + 0

# Ejemplo:
# n_sum(5) = 5 + 4 + 3 + 2 + 1 + 0 = 15
```