

TALLER PARA PRIMER PARCIAL ARQUITECTURA DE COMPUTADORES

1. Explique las ocho grandes ideas en Arquitectura de Computadores (Ver capítulo 1 en el texto guía).
2. Defina las siguientes siglas y explique que son:

a. DIMM	f. SIMM	k. RAID
b. DDR	g. LCD	l. HLL
c. GPP	h. PCI	m. LLL
d. GPU	i. PCIe	n. HDL
e. FPGA	j. USB	

3. Implemente una función **orgvector** en lenguaje de alto nivel C, lenguaje de bajo nivel RISC-V y lenguaje de máquina RISC-V que desarrolle un algoritmo de ordenamiento de las **n** primeras posiciones de un vector de enteros de 32 bits.
4. Implemente una función **mul** en lenguaje de alto nivel, lenguaje de bajo nivel RISC-V y lenguaje de máquina RISC-V que realice la multiplicación de dos enteros de 32 bits usando solo sumas.
5. Implemente la función **pot** en lenguaje de alto nivel, lenguaje de bajo nivel RISC-V y lenguaje de máquina RISC-V que realice la operación potencia de dos números enteros de 32 bits realizando llamados a la función desarrollada en el punto 6.
6. Implemente una función **pol** en lenguaje de alto nivel, lenguaje de bajo nivel RISC-V y lenguaje de máquina RISC-V que calcule un polinomio de grado **n**, en donde los coeficientes del polinomio se encuentran almacenados en un vector **k** de **n+1** posiciones, de la siguiente forma:

$$\text{pol}(x, k, n) = \sum_{i=0}^n k_i x^i$$

En donde el k_i se encuentra en la posición **k[i]** del vector **k**.

7. Implemente una función **fact** en lenguaje de alto nivel, lenguaje de bajo nivel RISC-V y lenguaje de máquina RISC-V que calcule el factorial de un número entero de 32 bits.
 - a. Realice esta implementación con recursividad y sin recursividad, y muestre los pros y los contras de cada una de las implementaciones.
 - b. Calcule los tiempos de ejecución de cada una de las implementaciones suponiendo que se ejecuta una instrucción por ciclo y que el reloj es de 1GHz, para un valor de 10 en la entrada.
8. Implemente una función **divint** en lenguaje de alto nivel, lenguaje de bajo nivel RISC-V y lenguaje de máquina RISC-V que calcule la división entera entre dos números enteros de 32 bits.
9. Implemente una función **modint** en lenguaje de alto nivel, lenguaje de bajo nivel RISC-V y lenguaje de máquina RISC-V que calcule el modulo entre dos números enteros de 32 bits.

10. Implemente una función `void primos(int n, int p[])` en lenguaje de alto nivel, lenguaje de bajo nivel RISC-V y lenguaje de máquina RISC-V que calcule los primeros n números primos y que los retorne en un vector p .
11. Implemente una función `int ReLU(int n)` en lenguaje de alto nivel, lenguaje de bajo nivel RISC-V y lenguaje de máquina RISC-V que calcule la función de activación *Rectified Linear Unit* usada en redes neuronales. Esta función retorna el valor máximo entre 0 y la entrada de la función, es decir, si la entrada es un valor negativo, la salida es 0, y si la entrada es un valor positivo, la salida es el mismo valor de la entrada.
12. Implemente una función `neurona` en lenguaje de alto nivel, lenguaje de bajo nivel RISC-V y lenguaje de máquina RISC-V. La función debe recibir cuatro argumentos: un apuntador a un vector de enteros con signo \mathbf{x} , que son las entradas de la neurona, un apuntador a un vector de enteros con signo \mathbf{w} , que corresponde a los pesos, un entero con signo valor umbral b y un entero n que corresponde al tamaño de los vectores \mathbf{x} y \mathbf{w} . La función de activación es la **ReLU** implementada en el punto anterior. La función debe calcular y retornar la salida de la neurona a partir de la siguiente descripción:

