1. Descreva a finalidade dos seguintes componentes:

* ULA – é responsável por executar as principais operações lógicas e aritméticas do computador.
* UC – Controla as ações a serem realizadas pelo computador, comandando todos os demais componentes de sua arquitetura. Garante a correta execução dos programas e a utilização dos dados corretos nas operações que as manipulam.
* Registradores – são circuitos digitais capazes de armazenar e deslocar informações binárias, e são tipicamente usados como um dispositivo de armazenamento temporário. São utilizados na execução de programas de computadores, disponibilizando um local para armazenar dados.
* MMU – é a unidade de gerenciamento de memória virtual, manipulação e proteção de memória.

1. Qual a diferença entre memória primária e memória secundária?

Principal: o acesso é mais rápido, mas a capacidade é mais restrita. Armazena informações temporariamente durante um processo realizado pela UCP.

Secundária: de acesso mais lento, mas de capacidade bem maior

1. Realize a conversão dos seguintes números decimais para binário.

* 178: 10110010
* 1022: 1111111110
* 5: 101
* 92: 1011100
* 2033: 11111110001
* 1024: 10000000000
* 32: 100000
* 64: 1000000
* 73: 1001001
* 12: 1100
* 3000: 101110111000
* 42: 101010
* 80: 1010000
* 7: 111
* 256: 100000000

1. Realize a conversão dos seguintes binários para decimal:

* 1110 1111 0011 1110: 61246
* 1111: 15
* 1110 1110: 238
* 1010 1010: 170
* 0101 0111: 87
* 1010 1010 1010: 2730
* 1111 1111: 255
* 1011: 11
* 1011 1110: 190
* 1011 0001: 177

1. Faça a conversão entre as unidades de medidas:
2. 1B para bit = 8 bit
3. 13kB para B = 13312B
4. 2000MB para GB = 1,953125GB
5. 3GB para bi = 24000000000
6. Converta o seu nome para binário.

01110110 01101001 01110100 01101111 01110010 01101001 01100001

1. Converta as seguintes frases para binário:
2. Hello World!

01001000 01100101 01101100 01101100 01101111 00100000 01010111 01101111 01110010 01101100 01100100 00100001

1. Um inseto no Mark II

01010101 01101101 00100000 01101001 01101110 01110011 01100101 01110100 01101111 00100000 01101110 01101111 00100000 01001101 01100001 01110010 01101011 00100000 01001001 01001001

1. Os líderes do futuro são os que empoderam os outros

01001111 01110011 00100000 01101100 11101101 01100100 01100101 01110010 01100101 01110011 00100000 01100100 01101111 00100000 01100110 01110101 01110100 01110101 01110010 01101111 00100000 01110011 11100011 01101111 00100000 01101111 01110011 00100000 01110001 01110101 01100101 00100000 01100101 01101101 01110000 01101111 01100100 01100101 01110010 01100001 01101101 00100000 01101111 01110011 00100000 01101111 01110101 01110100 01110010 01101111 01110011

1. Encontre e copie um algoritmo desenvolvido em uma linguagem de baixo nível e um desenvolvido em uma linguagem de alto nível.

- Assembly:

section .text

 global \_start ; informa o linker (ld) qual é o ponto de entrada

\_start: ; ponto de entrada

mov ebx, 0 ; valor de retorno para o SO(Sistema Operacional)

mov eax, 1 ; valor da syscall exit()

int 0x80 ; chamando o kernel

- Python:

>>> def cumprimentar(s):

... s = "Olá, " + s # s na tabela de símbolos local é reatribuída.

... print(s)

...

>>> pessoa = "João"

>>> cumprimentar(pessoa)

Olá, João

>>> pessoa # pessoa usado para chamar continua vinculado ao objeto original, ou seja, 'João'.

'João'