# Arquitetura de Computadores

A arquitetura de um computador refere-se às propriedades que têm impacto direto sobre a execução lógica de um software. Os computadores são usados para **guardar, processar e tornar acessível informação**. O computador processa dados automaticamente e é programável. Um computador é constituído por **hardware** (componente física) e por **software** (componente lógica). O

## Os principais **componentes de hardware** são:

* **Dispositivos de Entrada e de Saída**

- **Dispositivos de Entrada:** São dispositivos físicos que recolhem os dados a serem processados, por exemplo, o teclado;

- **Dispositivos de Saída:** São dispositivos físicos que mostram os dados já processados, ou seja, mostram os resultados, por exemplo, os monitores;

-**Dispositivos de Entrada e de Saída (Híbridos):** São dispositivos físicos com funções de Entrada e de Saída de dados.

* **Memória-** Podem ser **voláteis ou não voláteis**. As memórias voláteis são as que requerem energia para manter a informação armazenada, já as não voláteis são aquelas que guardam todas as informações mesmo quando não estiverem a receber alimentação. Existem **vários tipos de memoria**:

**-** **RAM (Random Access Memory):** É uma **memória Primária**, ou seja, o computador não pode funcionar sem ela. A função da RAM é de **Leitura e Escrita,** é usada pelo processador para armazenar os arquivos e programas que estão a ser processados, quando é utilizada em conjunto com o Disco Rígido aumenta o desempenho do computador. A RAM identifica a capacidade de acesso a qualquer posição e qualquer momento, tem **acesso aleatório**. É **volátil** e existem diferentes tipos de RAM (DIP, DDR, SIPP, entre outros).

**-** **ROM (Read-Only Memory):** Ao contrário da RAM, a ROM apenas permite **leitura**, tem também **acesso aleatório** e é **não volátil**. Faz parte de um conjunto de instruções operacionais programadas diretamente no hardware do computador, o **firmware**. A **BIOS** (Sistema Básico de Entrada/Saída) é um firmware responsável pelo suporte básico de acesso ao hardware, bem como iniciar a carga do sistema operacional, a relação com a ROM é que a BIOS fica gravada na ROM, impedindo assim de ser desinstalada.

**-Caches:** As caches são memórias que se situam entre o núcleo do processador e a memória principal, no sentido de tornar o acesso mais rápido à memória externa do processador. Baseiam-se no principio da **Localidade.** Pode ser **Localidade Temporal** (Este principio refere que se um endereço foi utilizado recentemente, é provável que seja novamente acedido num curto espaço de tempo) ou **Localidade Espacial** (Se um endereço foi acedido então os que estão próximos do mesmo têm mais probabilidades de serem acedidos, mais do que os mais afastados).

* **CPU** (Central Processing Unit / Unidade Central de Processamento) - Interpreta e executa as instruções do Software, é o cérebro do computador. A CPU é dividida por:

**-** **ALU – Arithmetic Logic Unit / Unidade Lógica e Aritmética**- Executa Operações Aritméticas e Lógicas.

**-Control Unit / Unidade de Controle** -Troca informação com a memória e Dispositivos de Entrada e Saída

**-Registradores-** Pequenas Unidades de Memórias Internas do Processador

**-Motherboard** (Placa Mãe) - Interliga os vários componentes do computador.

**Software-** É o conjunto dos programas e dos meios não materiais que possibilitam o

funcionamento do computador, na execução de diversas tarefas. É composto pelos Sistemas Operativos, linguagens de programação, entre outros. Existem dois tipos de linguagens programação:

* **Alto Nível-** Próximo á linguagem humana
* **Baixo Nível-**Próximo á linguagem do computador

## **COMPUTADOR VS HUMANOS**

O computador tem várias características:

**Automatização –** Executa tarefas repetitivas indefinidamente “sem se cansar”.

**Capacidade –** de Memória e Processamento.

**Rapidez –** Executa tarefas repetitivas indefinidamente “sem se cansar”.

**Custo –** Trabalha continuamente “sem pedir ordenado ou aumento”.

Os humanos também têm diversas características como a inteligência, a criatividade e o trabalho físico especializado.

Mas afinal o que é que um computador é capaz de fazer? Que tipo de problemas consegue resolver? Todos problemas que envolvam processamento de informação. Um humano consegue fazer aquilo que o computador faz, no entanto, o computador consegue fazê-lo de uma forma mais rápida, mais fiável e com maiores capacidades de processamento e de memória. Portanto, o processamento de informação pressupõe a existência prévia de um algoritmo que num computador é implementado por um programa.

A Geração dos computadores é constituída por **5 gerações.**

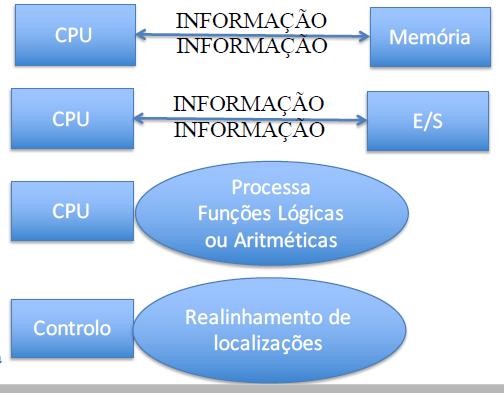
## **Lei de Moore:** Resumidamente a lei defende que o numero de transistores num processador duplicava a cada dois anos mantendo o mesmo custo e o mesmo espaço (ou menos). Mais tarde a lei alterou no numero de meses de 24 meses para 18 meses e mantêm-se ainda hoje.

## **Funções de um computador:** A função básica é executar programas, mas mais concretamente as funções são **PROCESSAR, ARMAZENAR, TRANSFERIR** e **CONTROLAR**.

Este programa é um conjunto de instruções que são processadas pela CPU pela seguinte ordem:

1. **Ciclos de Extração**- O processador **lê/procura** as instruções uma a uma. As instruções extraídas são carregadas num registo do processador conhecido por **Registo de Instrução** (IR).
2. **Execução-** O processador **executa** cada uma das instruções lidas/procuradas
3. **Repetição do 1. e 2.**

Cada instrução, que se encontra em código binário, especifica a ação a ser tomada pelo processador. O processador pode executar o programa de uma das seguintes situações:



## **Arquitetura JOHN VON NEUMANN**

Caracteriza-se pela possibilidade de um computador armazenar os seus programas e dados no mesmo espaço de memória, tipo 2 em 1.

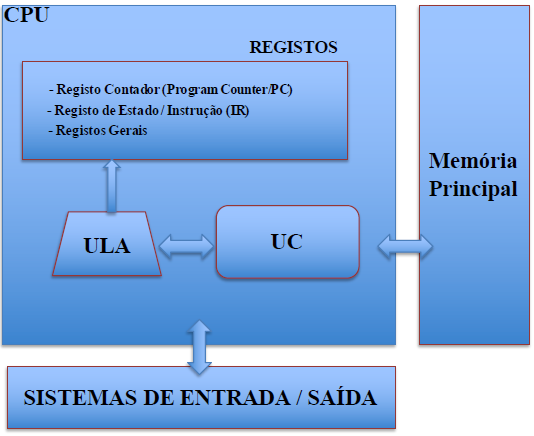
A máquina proposta por Von Neumann reúne os seguintes componentes:

1. **Uma unidade central de processamento (CPU)** que vai ver os **registos** (informações do programa)**.** Os registos podem ser de **endereçamento** (os que vai guardar o endereço das instruções dum programa), de **instrução** (que serve para guardar/registar o estado de execução de um programa) ou os registos **gerais**. A CPU ainda está subdividida por:

**-Uma Unidade de Controlo (UC)**, cuja função é encontrar um programa na memória e executá-lo, instrução por instrução.

**-Uma unidade lógica e aritmética (ULA),** cuja função é efetuar operações aritméticas e lógicas. Depois disponibiliza essas operações á execução dos programas.

1. **Uma memória**
2. **Sistema de Entrada/ Saída**



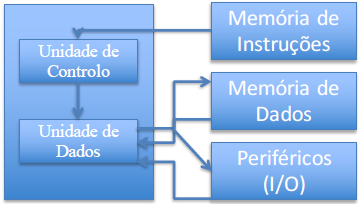
## **Arquitetura HARVARD**

A grande vantagem da arquitetura de Harvard face à arquitetura de von Neumann, é que permite ao processador aceder ao mesmo tempo à memória de dados e à memória de instruções, consequentemente os programas são executados de forma mais rápida.

É composto por:

1. **Unidade de dados-** Vai executar as operações que o processador precisa de executar
2. **Unidade de controlo-** Vai ler e interpretar as instruções e dá ordens á unidade de dados para as executar, vai também coordenar a comunicação com o exterior.

Resumidamente o processador vai ler a instrução, ler os dados, executar as instruções e armazenados os dados, criando-se um ciclo.



|  |  |
| --- | --- |
| **ARQUITETURA DE JOHN VON NEUMANN** | **ARQUITETURA DE HARVARD** |
| **Uma memória** para guardar os dados e programas | **Duas memórias**, uma para os dados e outra para os programas. |
| O Processador precisa de **2 ciclos** “clock” para completar uma instrução | O Processador precisa de **1 ciclos** “clock” para completar uma instrução |

## **BARRAMENTO (BUS)**

É um meio físico que liga os diversos dispositivos de um computador, se um computador não tivesse isto seria apenas uma ligação ponto a ponto. Os barramentos são compostos por:

* **Barramento de endereços-**Permite indicar a que dispositivos o processador necessita aceder, assim como a que parte especifica desses dispositivos (células individuais)
* **Barramento de dados-** O processador usa este barramento para ler e escrever dados, fazendo um barramento bidirecional
* **Barramento de controlo-**Formado pelos sinais que controlam o acesso do processador à memória e periféricos.

## **Avaliação do Desempenho do Processador**

Quanto mais rápido for a execução de um programa maior desempenho tem o computador. O desempenho é derivado dos seguintes componentes:

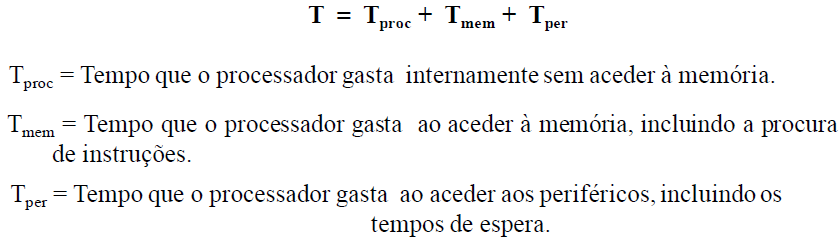
1) **Processador-** O mais rápido. O objetivo é aumentar a frequência do relógio e diminuir o número de ciclos por instrução.

2) **Memória**- Mais lenta que o processador. Diminuir o tempo médio necessário para aceder às memórias.

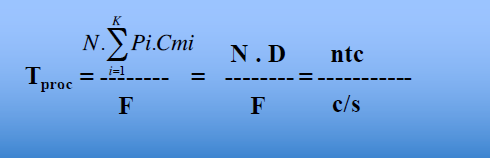
3). **Periféricos**- Tipicamente o componente com menor desempenho. Por isso o que pode contribuir mais para fracos resultados de desempenho.

4) **Compilador** “Mau” código pode produzir pouco desempenho. Privilegia-se assim as boas práticas e código bem direcionado para os objetivos.

A fórmula para calcular o tempo que um programa demora a ser executa por um computador:



Se pretendermos melhorar o tempo do processador temos de dividir o Tproc por o tempo que se quer melhorar (dobro, triplo…).



N-Numero total de instruções

F-Total dos ciclos por segundo

K-Número de classes (ultima classe)

i=1 ou o índice inicial (primeira classe)

Pi-Percentagem de ocorrência de cada classe

Cmi- Numero de ciclos por instrução

## **DISCOS MAGNÉTICOS**

Discos magnéticos, Hard Drive ou **Discos Rígidos**, são dispositivos que servem para guardar dados de forma persistente de um computador, é um dos principais componentes de um computador. Comunicam com o computador através de interfaces. Os tipos mais comuns são chamados de: IDE (ATA), EIDE, SCSI, SCSI-2, SATA, entre outros.

### **Os discos rígidos funcionam da seguinte forma:**

Em cada superfície desloca-se uma cabeça, com a função de ler ou escrever dados. As cabeças movimentam-se no raio de cada disco. Cada superfície (do disco) é composta por um conjunto de pistas concêntricas (Tracks) e cada pista é dividida em sectores radiais. O Cilindro é o conjunto das pistas de todas as superfícies na mesma perpendicular.

**Paralelismo-** Um ou mais discos a funcionar conjuntamente no sentido do mesmo fim.

Existem duas fórmulas que nos podem ajudar.

**Total de Sectores = Nº Cilindros x Cabeças (ou superfícies) x Nº Sectores por pista**

**Capacidade do Disco = Total de Sectores x Comprimento do Sector (em bytes)**

Existem ainda dois tipos de formatação de disco:

**Formatação de Baixo Nível** – Criação de Pistas e Setores

**Formatação de Baixo Nível** – Criação do sistema de Ficheiros, ext2, VFAT, NTFS, etc...

Os discos Rígidos podem ser baseado em **Software** ou **Hardware**

|  |  |
| --- | --- |
| **Software** | **Hardware** |
| Depende do S.O. | Independente do S.O. |
| Configuração mais complexa | Maior Performance |
| Custo mais reduzido | Custo mais elevado |
| Mais probabilidade de Overhead (processamento ou armazenamento em excesso) | Normalmente com controlador dedicado |
| Requer mais CPU | Hot-Swapping (permite que a troca de um disco defeituoso possa ser feita com o sistema operativo em execução.) |

## **Interfaces (Discos Rigidos)**

1. **ATA (Advanced Technology Attachment),** padrão para interligar dispositivos de armazenamento, nomeadamente, discos rígidos, cd-roms, etc. Transmitem os dados a partir de um barramento de 40 ou 80 ligações paralelas.
2. **(E)IDE (Extended Integrated Drive Electronics) -** o controlador está integrado à unidade de disco rígido, livrando a board dessa função.
3. **ATAPI (Advanced Technology Attachement Packet Interface) –** Inicialmente apenas desenhada para discos rígidos, adaptada mais tarde para outros dispositivos.
4. **UDMA (Ultra DMA-Direct Memory Access) –** Transfere os dados entre disco e memória sem interferência do CPU.
5. **SATA (Serial Advanced Technology Attachment),** transmitem dados em série através de uma ligação ponto por ponto. Os campos SATA são formados por dois pares de fios, um para a transmissão e outro para a receção) e mais três condutores para a ligação terra. Isto permite uma maior rapidez, hot swapping,cabos mais estreitos e autorreconhecimento dos dispositivos imediato.
6. **SCSI (Small Computer Systems Interface),** usado em computadores de médio e alto desempenho. Taxas de transferência mais elevadas que o ATA. Usa menos recursos da CPU. Mais rápidos no acesso e na escrita dos dados, menor encobrimento, maiores buffers.
7. **SAS (Serial Attached Small Computer Systems Interface),** em 2003 o padrão SCSI evoluiu também para uma “versão” serie. Melhor relação custo-benefício. Admite cabos mais longos, pode trabalhar com discos SATA II (Se bem que o contrário não é possível). Podem-se juntar na mesma ligação vários discos ao mesmo tempo (num único barramento).
8. **FC (Fibe Channel),** tecnologia de rede, com taxas na ordem dos gigabits por segundo. Usado na construção de redes de armazenamento (SAS). Tem 3 ligações:

a) **Ponto por Ponto (FC-P2p)**: permite a ligação entre dois dispositivos;

b) **Arbitrated Loop (FC-AL**): permite a ligação entre dispositivos em forma de anel de rede, a adição de dispositivos ou interrupção do anel causa a paragem de todo o anel.

3) **Ligação via switch (FC-SW)**: maior desempenho, permite múltiplas ligações em simultâneo. Falhas numa porta não prejudicam a outra... Maior Custo.

## **DAS (Direct Attached Storage)**

Os dispositivos de armazenamento encontram-se ligados diretamente a um computador por interface SCSI (Small Computer System Interface). Estão limitados sob o ponto de vista da escalabilidade, pois dependem do limite que cada um dos servidores tem relativamente à quantidade de dispositivos permitida. Também, sempre que se necessite de intervenção nos servidores a atividade de armazenamento ficará interrompida. Fraco desempenho face a tecnologias mais recentes. Abastecidos de “Block-level” interface. (A gestão do sistema de ficheiros é gerida do lado do host, assim como os protocolos de comunicação).

## **NAS (Network Attached Storage)**

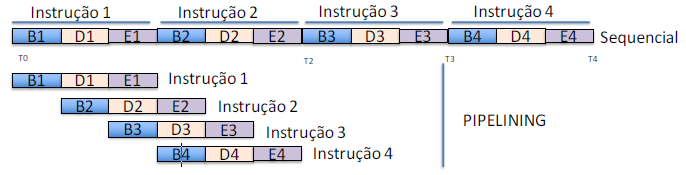
Unidade de Armazenamento que se encontram ligados a uma rede de forma autónoma, através de um sistema de ficheiros próprio (mas uniformizado). Os “clients” solicitam os ficheiros ao dispositivo a partir de RPC (Remote Procedure Calls). Munidos de “File-level” interface. (A gestão do sistema de ficheiros é gerida do lado do dispositivo, assim como os protocolos de comunicação).

## **SAN (Storage Area Network)**

Comporta grande trafego de dados forma mais rápida. Rede usualmente de fibra ótica. Normalmente não são acessíveis a partir da rede local. Munidos de “Block-level” interface. (A gestão do sistema de ficheiros é gerida do lado do host, assim como os protocolos de comunicação).

## **Pipelining**

Tipo de Processamento (processamento em estágios) muito parecido com uma linha de montagem. Uma tarefa é dividida em sub-tarefas, executadas em diferentes fases. O Processamento total de uma instrução é dividido em tarefas mais pequenas (estágios), executadas por unidades especializadas numa só tarefa (operação). Ao conjunto destas unidades chamam-se cadeia de estágios.



## **Filosofia RISC**

Arquitetura de processadores que priveligia uma preocupação base sobre o **HARDWARE.** **Não têm microprogramação**, não têm instruções executadas diretamente pelo hardware.

**Quanto mais funcionalidades o hardware tivesse:**

**1)** Mais fácil seria para o compilador gerar código

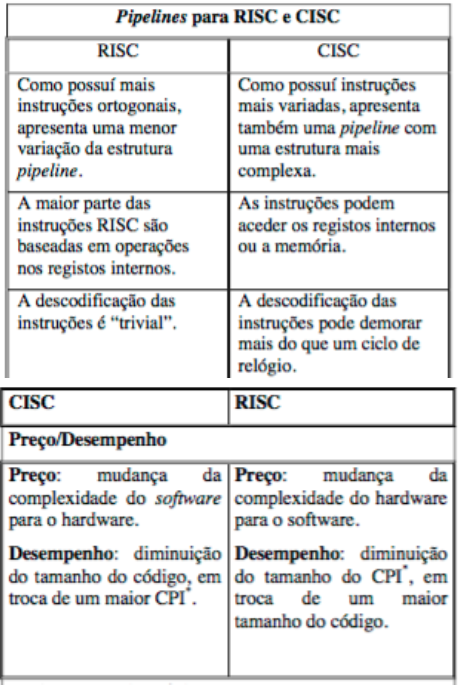
**2)** Maior desempenho o computador teria.

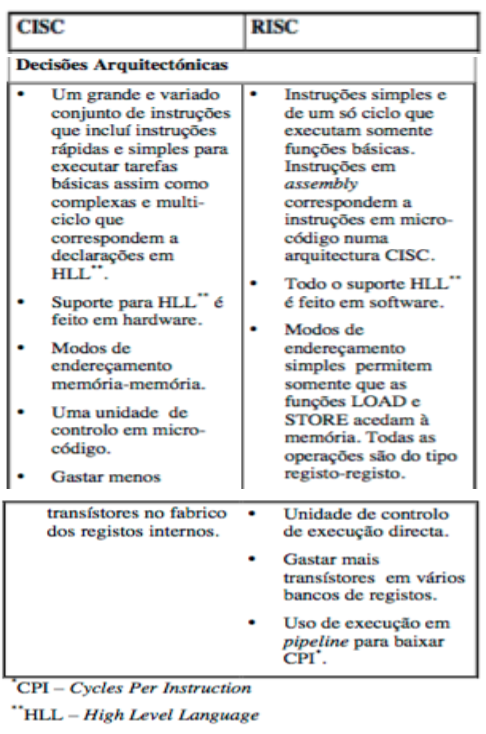
Esta arquitetura durou até ao computador VAX da Digital Equipment, 1977. Exigia circuitos muito complexos, o que impedia a frequência do relógio elevar-se Utilização de Pipelining (intensamente). Têm instruções de um só ciclo, mas têm ganhos de performance ao nível da execução de cada instrução.

## **FILOSOFIA CISC**

Oposto ao RISC, ou seja, têm o hardware simples e têm poucas instruções- Usam a microprogramação e o microcódigo. É MULTI-CLOCK, isto é, grande conjunto de instruções. Tem poucos registos e a pipeline tem uma estrutura mais complexa.

## **RISC/CISC**





## **BASES NUMÉRICAS**

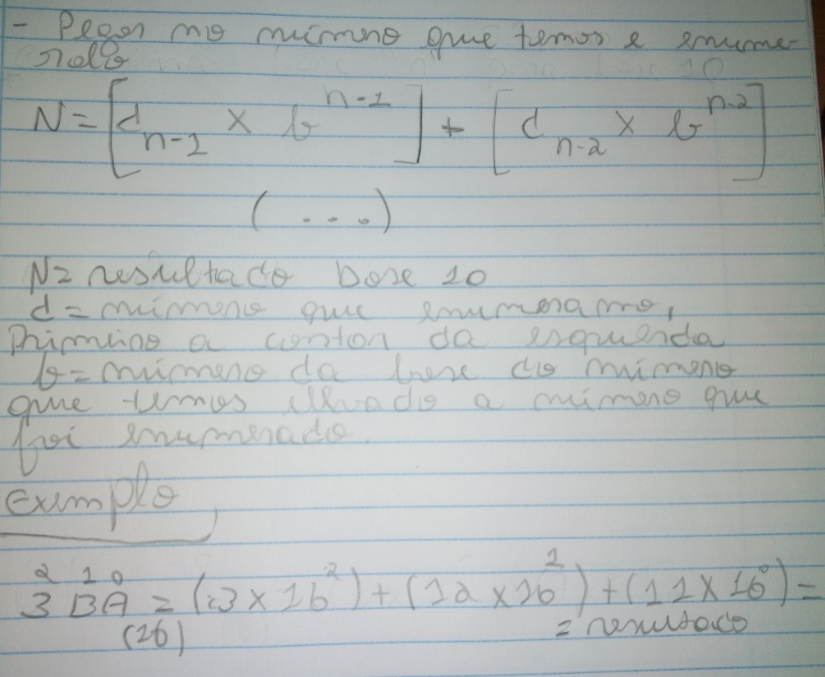
Decimal -> Base 10 ->0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Binário -> Base 2 ->0, 1

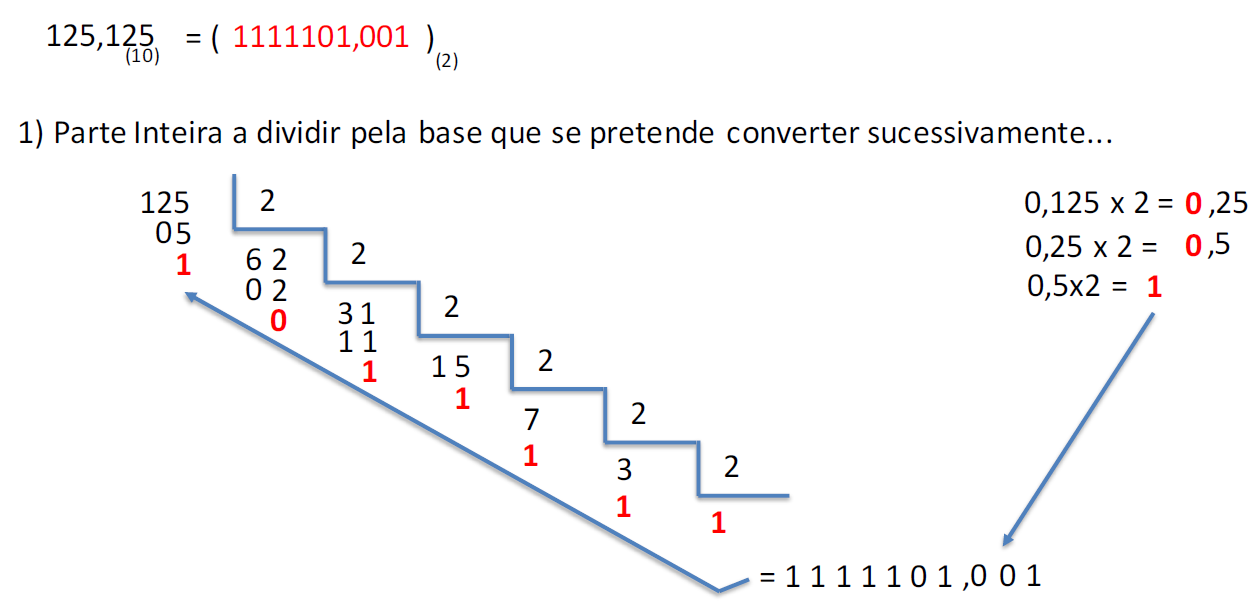
Octal -> Base 8 ->0, 1, 2, 3, 4, 5, 6, 7

Hexadecimal -> Base 16 ->0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

## **CONVERSÃO BASE “n” PARA BASE 10:**

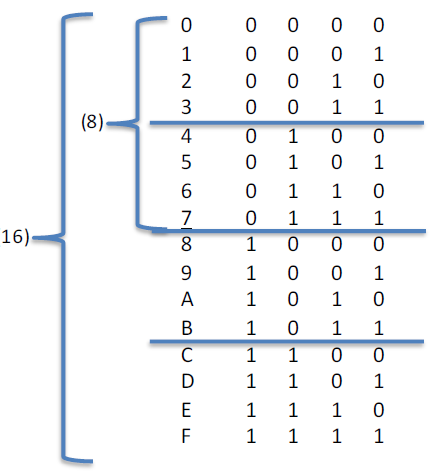


## **CONVERSÃO BASE 10 PARA BASE “n”:**



## **CONVERSÃO BASE “n” PARA BASE “n” (exceto o 10)**

Tabela Base:

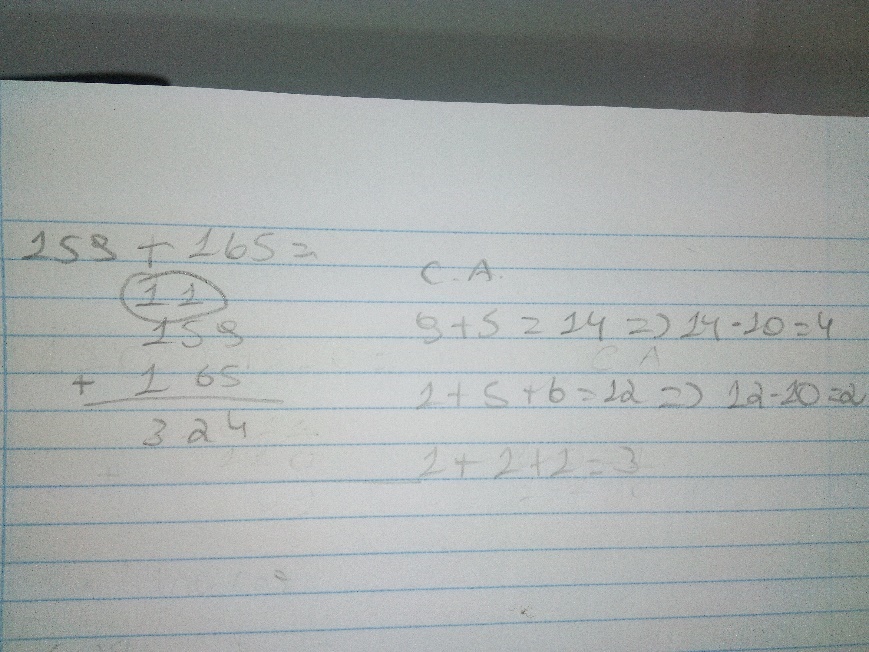


**Dividir por dígitos:** Para saber em quantos dígitos se dividir (x) deve-se pegar em 2^x=n; n=2,8,16 (n-número da base que ser quer transformar) e resolver a equação.

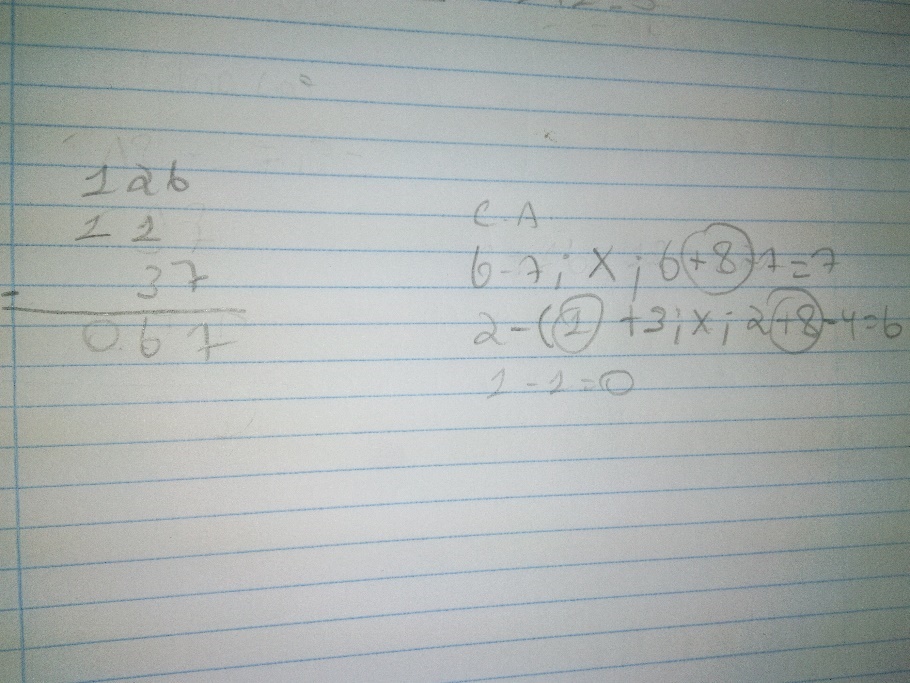
Ir a tabela verificar o numero do enunciado (vai ser o numero de dígitos já encontrado da direita para a esquerda) e por o resultado da direita para a esquerda.

## **SOMAR BASES NUMERICAS**

Só se pode somar e subtrair números da mesma base se não fazer transformações necessárias.



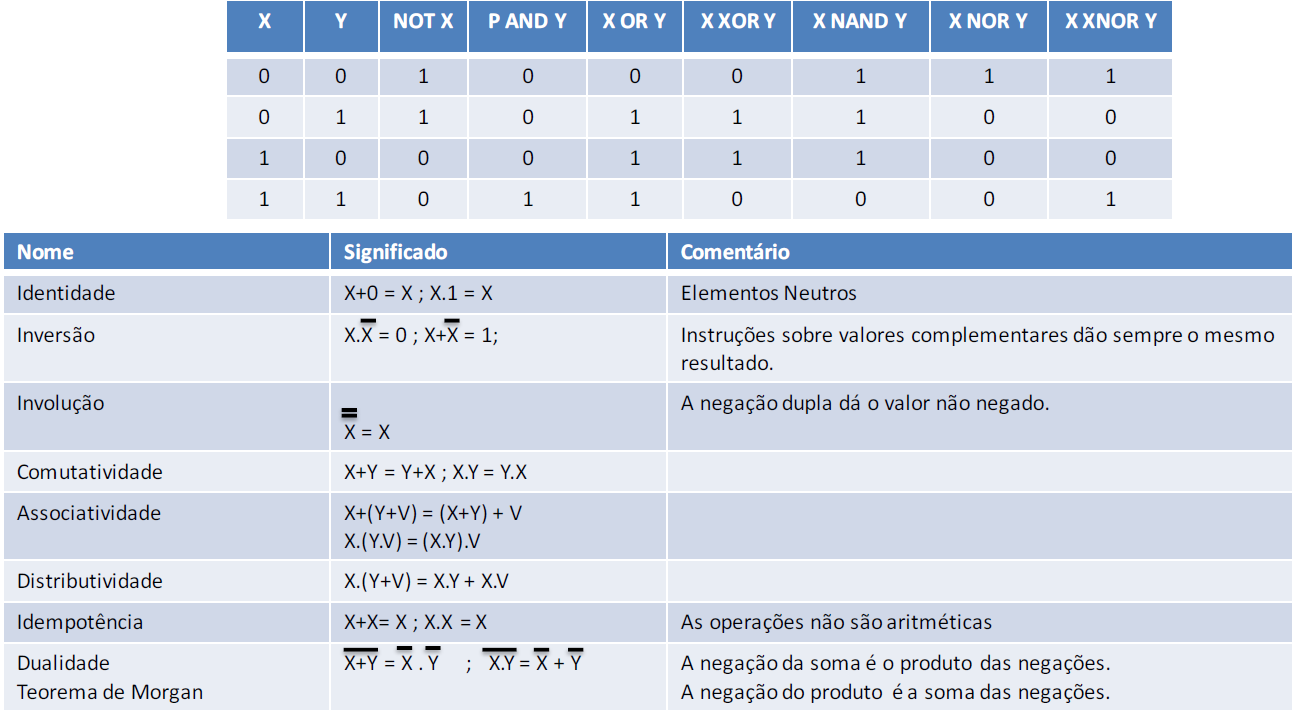
## **SUBSTRAIR BASES NUMÉRICAS**

****

## **ALGEBRA BOLEANA**

Descreve uma álgebra com dois valores (1 e 0) e três operações (AND, OR e NOT). Em que as prioridades são a NOT (|) a primeira, seguindo-se o AND [conjução, e)] (.) e por fim a OR [disjunção,ou] (+).

## **FUNÇÕES LÓGICAS**



# SISTEMAS OPERATIVOS

Um sistema operativo é um software que atua como intermediário entre os utilizadores e o hardware.

## **Funções do S.O.:**

* Tornar simples a utilização do hardware de uma máquina.
* “Obter o máximo rendimento do hardware, utilizando a mais eficazmente possível capacidade de processamento” do sistema. (otimização)
* Gerir Processos
* Gerir a Memória
* Gerir os Ficheiros
* Gerir os Dispositivos de Entrada e de Saída
* Gerir os Utilizadores

**Aplicação** - Conjunto de programas e de informação persistente.

**Recursos lógicos** - São o software (conjunto de aplicações) e a sua gestão é da responsabilidade do sistema operativo, estabelecendo o contacto entre os referidos recursos e o hardware (recursos físicos).

Um sistema operativo é um **gestor de recursos.** No âmbito dos recursos lógicos, é importante sublinhar que os mesmos devem ser o mais abstratos possível. A noção de abstração faculta ao sistema a capacidade para a reutilização dos recursos lógicos independentemente do recurso que os utilizam).

## **Principais Recursos Lógicos geridos pelos S.O. :**

**A) Processos-** Os processos simplificam a gestão da execução dos programas de forma abstrata. São um conjunto de instruções que o computador executa sem questionar o tipo de linguagem de programação ou o tipo de hardware existente, como se de uma máquina virtual se tratasse.

**B) Memória Virtual-** A memória é fundamental para que os processos se executem. Facilmente a memória principal pode ser totalmente absorvida, criando a execução de processos uma tarefa muito complexa. **Espaços de endereçamento** – conjunto de endereços discretos que a memória pode aceder. Associado a cada processo encontra-se um espaço de endereçamento. A memória é fundamental para que os processos se executem. A memória virtual compõe o uso da RAM com a memória secundária, tipicamente denominada por Discos Rígidos. O que aconteceria se um processo necessitar de um espaço de endereçamento maior que o fisicamente permitido? RAM, mais cara, menos “abundante”, mais rápida... Discos, menos caros, mais “abundante”, mais lentos... Ram + Discos = Simulação de uma memória mais vasta, sem prejudicar de forma significativa o desempenho.

**C) Sistema de Ficheiros-** Se existem processos...espaços de endereçamento...memória principal...memória secundária... É preciso organizar! Onde e como ficam os dados depois de processados? São necessárias chamadas ao sistema para criar, alterar e eliminar ficheiros que guardam dentro dos mesmos esses dados. O sistema de ficheiros é responsável pela organização dessas informações, informação persistente. O sistema de ficheiros hierarquiza a estrutura que recebe os ficheiros e virtualiza os dispositivos de memória de massa.

**D) Periféricos-** O Sistema operativo comunica com os dispositivos de entrada/saída através de uma lógica virtual (periféricos virtuais). Os periféricos virtuais (tipicamente conhecidos como “drivers”) permitem ao S.O. disponibilizar aos programas esses dispositivos, sem que seja condição da sua existência física.

**E) Utilizadores-** O Sistema operativo é também responsável pelos níveis ou perfis de utilizadores, garantindo o uso correto por parte dos utilizadores da informação que lhes diz respeito. Por exemplo, os ficheiros em UNIX, são protegidos com 9 bits de proteção a agrupados por 3 bits, um para o proprietário, outro para os membros do grupo do proprietário e para qualquer utilizador. Cada bit com uma tripla função, uma de leitura (read), outra de escrita ( write) e outra de execução (execute). O proprietário pode ler, escrever e executar. Já os membros do grupo só podem ler e executar, todos os outros só podem executar.

## **OBJETOS (recursos) de um sistema operativo:**

* Processos
* Espaços de endereçamento virtuais
* Ficheiros
* Periféricos virtuais
* Utilizadores

## **INTERFACE**

INTERFACE OPERACIONAL Textual ou INTERFACE OPERACIONAL Gráfica

BIBLIOTECA CHAMADAS AO SISTEMA (INTERFACE PROGRAMÁTICA) – Interface funcional do sistema, permitem que as linguagens de programação possam aceder ao sistema operativo de forma eficaz no sentido de uma uniformização aplicacional natural, entre aplicações e S.O.

## **CRITÉRIOS DE QUALIDADE DOS S.O.:**

**1 – Desempenho-** O S.O. é um gestor e nessa qualidade deve obter um equilibrio, o mais elevado possível entre o investimento e o desempenho dos recursos.

**2 – Segurança-** O S.O. é um dos pilares para a garantia das politicas de segurança de um sistema, quer seja vista de ume forma isolada ou em rede.

**3 – Robustez-**

**3 a – Fiabilidade-** Probabilidade de não existir falhas de paragem no sistema

**3 b – Disponibilidade-** Quanto menos falhar existir mais disponível é um sistema.

**4 – Qualidade das Interfaces-** Completas, de fácil utilização e documentadas

## **EVOLUÇÃO HISTÓRICA DOS S.O.:**

**Monitor de Controlo -** Primeiros programas (Software) que permitiam executar tarefas básicas (Carregar software na memoria central**,** editar software**,** monitorizar a execução do software e é baseado em sessões de trabalho.

**Multiprogramação -** A Multiprogramação faz com que a CPU fique com vários programas simultaneamente ativos na memória (consiste em dividir a memória em N partições fixas de tamanhos variados) e permite que um programa seja executado enquanto outro espera por E/S.

**Sistemas Interativos -** Permitiram uma interatividade entre os seus utilizadores e o sistema central através de umterminal.A aparente disponibilidade do computador central aos seus utilizadores em simultâneo, eradevida á elevada velocidade de processamento face à velocidade operativa dos seusutilizadores. (Mesmo que cada utilizador demore 1 segundo por cada três teclaspressionadas o processador conseguia ser muito mais rápido)**.** A interação evoluiu tendencialmente para uma melhor e mais fácil comunicação entremáquina e homem.

**Memória Virtual-** Espaço de endereçamento virtual**.** Baseado em observação estatística (Principio da localidade de referência)**.** Basicamente define que os programas que se podem manterda memória principal, são aquela cuja probabilidade de seexecutarem na vizinhança de endereçamento (da instruçãoatual) é elevada.

## **CLASIFICAÇÃO DOS S.O.:**

**S.O. Para Computadores Pessoais,** são S.O. Destinados a um único utilizador, cuja principal preocupação é garantir a melhor usabilidade possível.

**S.O. De Tempo Virtual,** os mais difundidos. O tempo de execução dos programas, não tem relação com o tempo cronológico exterior (Job-shop). O Exterior não limita o tempo de execução dos programas.

**S.O. De Tempo Real,** sistemas onde a noção de tempo é relevante. O objetivo deste tipo de S.O. é estabelecer uma resposta com os seus utilizadores baseada num intervalo de tempo previamente especificado. Se esse objetivo (cumprimento temporal) não for cumprido pode já não ter relevância para a aplicação. Várias tarefas criticas são processadas em simultâneo. (Exemplo: Solaris, Spectra)

**S.O. Embebidos,** dimensão reduzidos destinados a equipamentos igualmente de tamanho reduzido, por exemplo para telecomandos, telemóveis, PDA. São mais limitados. (Exemplo: Palm OS, Windows CE, Symbian, Windows Phone 7, Android, etc…)

## **ORGANIZAÇÃO/ ESTRUTURA DOS S.O.**

**Núcleo -** Conjunto dos programas que implementam as funcionalidades básicas do Sistema Operativo.

**Biblioteca de chamadas do sistema-** Interface programática, estabelece a ligação entre as aplicações e o S.O.

**Processos do Sistema -**Conjunto de processos que se executam fora do núcleo.

O núcleo inclui ainda outros mecanismos tais como:

- A segurança

- A gestão dos utilizadores

- As interfaces gráficas

- A gestão da mobilidade

- A gestão da energia

- Criação, eliminação E sincronização dos processos.

- Escalonamento de Processos.

- Tratamento das Interrupções

- Comunicação entre processos.

**Gestão de Processos**: Multiplexa a máquina física entre um conjunto de processos e ações sobre os mesmos tais como:

**Memória:** Gere a memória física e memória virtual. Usa algoritmos na manipulação do espaço de endereçamento dos processos.

**Entrada / Saída:** O Núcleo gere os periféricos através da interface de chamadas ao sistema.

**Os periféricos:** têm também características muito particulares que são manuseadas através de software, representando um “virtualismo” entre essas características e máquina física, situação gerida também pelo núcleo, comumente chama-se a esse software os DRIVERS.

Sistema de Ficheiros de uma forma eficiente o armazenamento dos dados nos discos (memórias secundárias). Nomeadamente organização, a administração e a partilha segura dos ficheiros.

**Comunicação entre Processos:** Cooperação entre processos => Mecanismos de sincronização e comunicação. Assim é possível a existência de aplicações concorrentes no âmbito de uma comunicação distribuída, **IPC (Inter-Process Communication)** e **Memória Partilhada; Semáforos; Filas de Mensagens, etc.**

**Modo Núcleo (Kernel Mode)** (Mais privilegiado) - O Processador pode aceder a toda a memória.

**Modo Utilizador (User Mode)** (Menos privilegiado) - O Processador pode aceder apenas ao espaço de endereçamento atribuído.

Esta comutação de modos é controlada automaticamente através de exceções ou interrupções. AS INSTRUÇÕES QUE SE PODEM APENAS EXECUTAR EM MODO KERNEL, VAMOS CHAMAR DE **INSTRUÇÕES PROTEGIDAS.**

## **Exceções e Interrupções**

Exceção – Condição especial que altera o comportamento normal de um programa. São eventos inesperados que acontecem no processador (por exemplo, overflow/divisão por 0, interrupção do relógio do processador, posição de memória inválida. O tratamento de exceções é efetuado através de uma rotina especifica para o efeito: Exception handling e neste caso implica a mudança de estado: Modo User -> Modo Kernel. (alterando-se também o espaço de endereçamento).

**Assíncronas (Interrupções)**, independente do programa em execução, por exemplo um problema no disco, no relógio do processador ou falha de energia.

**Síncronas (Exceções)**, dependem do programa, pois são provocadas pelo mesmo, por exemplo, a divisão por 0, o acesso a um endereço não autorizado, ou o forçar de um nível de permissão restrito (Leitura / Escrita).

## **Rotina Tratamento de Exceções e Rotina de Interrupção.**

A tabela que regista as rotinas de tratamento chama-se **Tabela de Interrupções** Cada exceção ou interrupção tem um vetor associado. As exceções já estão pré-definidas pela arquitetura do processador, com vetores (índices) já definidos na tabela de interrupções, todas as outras posições livres podem ser ocupadas pelos periféricos utilizados. O vetor é o parâmetro da instrução INT, que desencadeia a exceção.

## **Controlo de aceitação das interrupções**

**1) Inibição (disable)** – Inibe as interrupções a partir de uma existente.

**2) Desinibe (Enable)** – Desinibe as interrupções, tendo em conta a prioridade das mesmas.

O Controlo de aceitação pode ser feito através de software, no entanto existem alguns processadores cujo controlo de aceitação é feito via hardware do processador. (IPL – Interrupt Priority Level). Uma Interrupção ou uma exceção, exige sempre a paragem de um programa para a execução de outro. É necessário utilizar um mecanismo (normalmente através do registo contador, Program Counter e do registo do estado do processador) que salvaguarde a posição de execução do programa interrompido, para retomar o mesmo mais tarde a partir da mesma posição.)

## **Processos do sistema**

São processos que tipicamente são executados no arranque do sistema (bootstrap). Exectuam-se em backgroud. Também podem ser iniciados sem ser no arranque, de acordo com a necessidade do sistema. Estes processos em UNIX chamam-se de deamon, em Windows, Services (task manager)

FUNCÕES DO NÚCLEO 🡪CHAMADAS SISTEMA 🡪 PROCESSOS SISTEMA

## **EVOLUÇÃO DA ORGANIZAÇÃO DO NUCLEO DOS S.O.**

**Núcleos Monolíticos**- 1ºs Sistemas Operativos.Não modular, maior rendimento. Sistemas mais vastos, logo, mais código, logo, mais probabilidade de erro (que na introdução de novo código como na manutenção do código existente).Gestor de Periféricos ao mesmo nível do Núcleo do S.O. -> Perigoso**.** Probabilidade de tentativas à criação de Gestores de Periféricos “maléficos”.Unix, Linux e Windows -> Núcleos Monolíticos.

**Núcleos em Camadas** - Altera-se uma camada (nível) sem prejudicar o funcionamento de outras.Um nível encapsula os outros níveis anteriores.Cada uma das camadas tem uma interface interna, para implementar cadauma das suas funcionalidades, podendo recorrer (encapsulamento) aos níveisanteriores.Primeiro sistema em camadas: THE (Estrutura totalmente lógica).Multics, baseia-se neste modelo, que criou um sistema (em memória) paraproteger cada um dos anéis, através de espaços de endereçamento diferentes.Assim era possível criar um mecanismo que se invoque privilégios necessários àinvocação entre os níveis.

**Micronúcleos** - Objetivo em reduzir a complexidade do núcleo, aumentar a segurançae a robustez. Fornecer apenas através do micronúcleo os serviços básicos. Baseia-se num conjunto de processos sistema (Servidores), implementa-mas funcionalidades que o micronúcleo implementa, fora do núcleo do S.O. Sistema pioneiro: Mach. Os micronúcleos têm a vantagem da modularidade (possibilidade de coloca mais servidores) de acordo com as necessidades de evolução. Também invocar outros servidores de outras máquinas ligadas em rede e a separação entre os mecanismos base e mecanismos do sistema.

## **LINUX**

Listas – ls

Nova pasta – mkdir

Mudar de diretorio – cd

“Aqui” - .

Limpar o ecrã – clear

Ligar ficheiros(atalho) – ln, ln -s

Mover ficheiros e pastas – mv

Copiar ficheiros – cp

Copiar pastas – cp -r

Mostrar “onde estou” - pwd

Pesquisa de ficheiro – find / locate / whereis / which

Remover Ficheiros e Pastas – rm / rm -rf / rmdir

Copiar para sistema remoto – scp

\* - qualquer caractere

? - um caractere qualquer

~ - igual a $HOME

\ - Caractere de Scape

/ - Sepação directorios

$ - Variável

[ ] - Range de caracteres

' ' - valor real (ex.: '$HOME' = @HOME )

“ ” - valor dinámico (ex.: “$HOME” = /home/user )

## **WINDOWS:**

\*-Representa vários caracteres

?-Representa um caracter

Echo- escrever

Echo off- produz texti

Echo on- Ativa as informações do sistema

ECHO.-Salta uma linha

IF EXIST \*.bat ( echo existe ) ELSE ( echo não existe )

FOR /L %%x IN (Inicio,Incremento,Fim) DO ...

For %%A IN (01 02 03 04 05 06 07 08 09 10) DO (ECHO %%A)

PAUSE- Força uma pausa

%- Variável´

Rem-serve para fazer comentários

Dir- visualiza o conteúdo de um directorio

cd- mostra o nome da diretoria ou muda de diretoria

md-cria uma diretoria

rd-remove uma diretoria

copy com- cria um ficheiro

type- mostra o conteúdo

ren-Renomeia um ficheiro ou diretório

del- elimina um ficheiro

copy-copia ficheiro

xcopy- move um ficheiro

cls-limpa ecrã

set /a- valor aritmético

set /p- fazer uma pergunta

goto-Volta ao inicio

choise /t- segundos

choise /d-default

choise /n- esconder