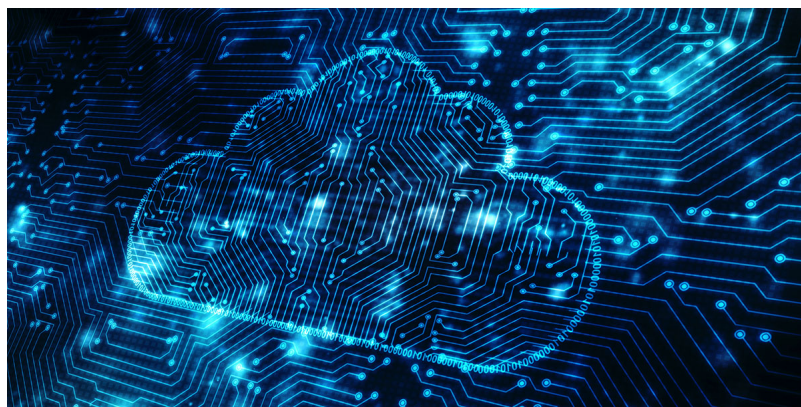


Organização de Computadores

Aula 2 - Hardware, Software e Conceitos

INTRODUÇÃO



Na aula anterior vimos um pouco da história da computação, como tudo começou e motivou o desenvolvimento tecnológico que conhecemos hoje e que continua evoluindo dia após dia. Também verificamos que o desenvolvimento se deu de diversas formas, tanto em hardware, a parte física do computador, como em software, a parte lógica do computador.

Nesta aula, vamos conhecer a estrutura física da divisão do computador, seus componentes e sua relação com as instruções a serem executadas através de programas. Também iremos estudar conceitos importantes para nosso dia a

dia em computação, que possuem relação direta com o bom entendimento da organização de computadores e que precisam fazer parte do vocabulário do aluno de tecnologia e futuro profissional de TI.

OBJETIVOS



Reconhecer os componentes de um computador.

Interpretar conceitos relacionados à computação.

Distinguir os passos para execução de um algoritmo e sua relação com linguagens de programação.

Relacionar BIT, BYTE e suas aplicações em computação.

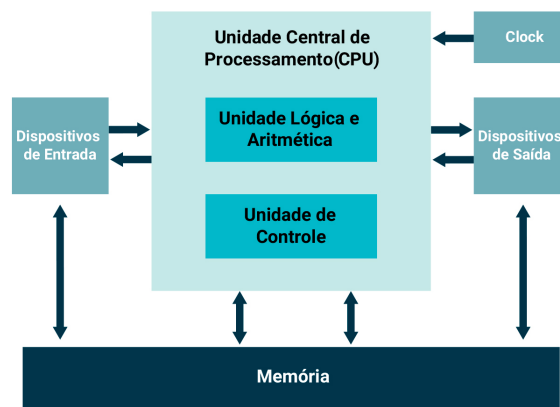
HARDWARE E SEUS COMPONENTES

Hoje temos contato com uma diversidade de dispositivos de computação, alguns com capacidade de processamento igual ao até mesmo superior a alguns computadores tradicionais. Por exemplo, o que conhecemos hoje como smartphones, telefones celulares de alta capacidade de processamento, possuem configurações de hardware realmente robustas.

O hardware de um computador é formado por diversos componentes e dispositivos interconectados, de forma que, através dessa comunicação, as suas funções em conjunto possam ser desempenhadas.

Cada equipamento possui a sua particularidade. Porém, como base, existem alguns componentes hoje presentes em qualquer equipamento, seja ele um computador de uso pessoal, uma calculadora, ou até mesmo o smartphone que citamos.

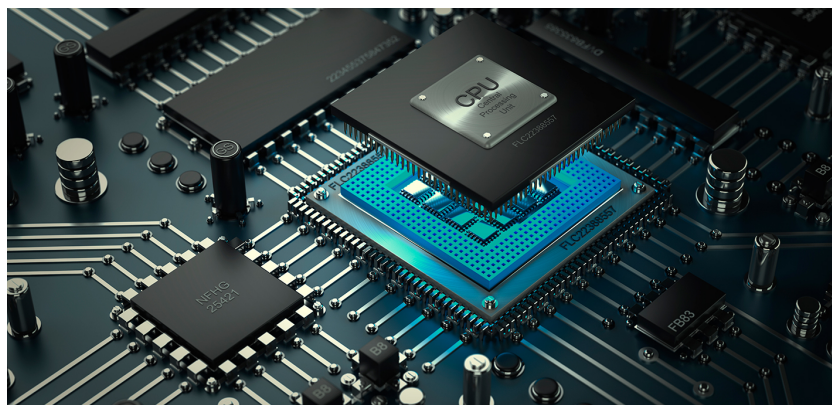
Veja abaixo os componentes do hardware cujas funcionalidades devemos conhecer:



Agora, vamos conhecer cada um desses componentes mais detalhadamente.

UNIDADE CENTRAL DE PROCESSAMENTO (UCP OU CPU)

É o dispositivo central de um computador. Muitos acreditam que a CPU é o gabinete do computador, mas, na verdade, ela é o que conhecemos hoje como processador.



Internamente, possui dois importantes componentes:

Unidade de Controle

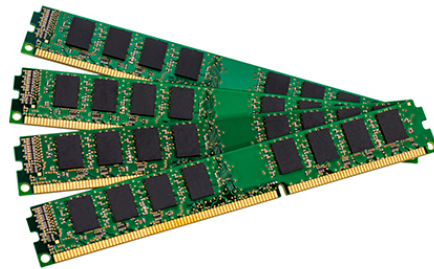
Responsável pelo gerenciamento de todo o fluxo de informações entre as unidades que fazem parte do hardware, bem como a execução das instruções.

Unidade Lógica e Aritmética

Responsável pelos cálculos e decisões a serem efetuadas de acordo com as instruções recebidas.

MEMÓRIA

Responsável pelo armazenamento das instruções e dados a serem executados ou manipulados pela CPU e seus componentes, bem como armazenamento de informações recebidas pelas Unidades de Entrada e Saída. Podemos dizer que a memória poderia ser memória RAM ou até mesmo o HD (ou disco rígido) como unidades de memória.



Memória RAM



Disco Rígido

DISPOSITIVOS DE ENTRADA

Todos os periféricos que são utilizados e interligados ao computador para envio de informações para o processamento. Por exemplo: mouse, teclado ou botões de função localizados no equipamento.



Mouse



Teclado

DISPOSITIVOS DE SAÍDA

Todos os periféricos utilizados para enviar alguma informação para o usuário. Uma impressora ou um monitor de vídeo são exemplos.



Impressora



Monitor

Esse é somente um resumo de tudo o que estudaremos de forma mais aprofundada nas próximas aulas, nos permitindo entender melhor a relação e a função de cada um desses componentes.

DISPOSITIVOS INTERNOS X DISPOSITIVOS EXTERNOS

É importante citarmos que o computador é formado por diversos componentes em sua montagem de hardware.

Em resumo, temos:

Dispositivos Internos

- Processador
- Placa-mãe
- Disco rígido
- Placa de rede

Dispositivos Externos

- Teclado
- Mouse
- Impressora

Saiba Mais

, Antes de continuar seus estudos, saiba mais sobre Placa-mãe ([galeria/aula2/docs/aula02.pdf](#)).

SOFTWARE

Software são todos programas responsáveis por administrar a execução de funções pelo hardware e automatizar as tarefas que o computador irá desempenhar.

Classificação de acordo com as funcionalidades

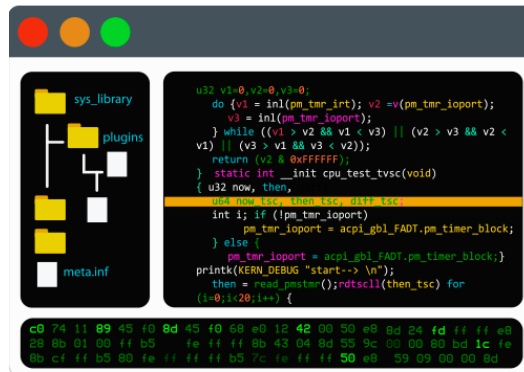
Cada software possui uma função específica, de forma que é necessário, neste momento, classificá-lo em duas categorias principais:

Software Básico:

Utilizado para produzir ou gerenciar a operação do computador e a relação dentre softwares aplicativos com o hardware.

Exemplos:

- Linguagens de programação;
- Sistemas Operacionais;
- Compiladores.

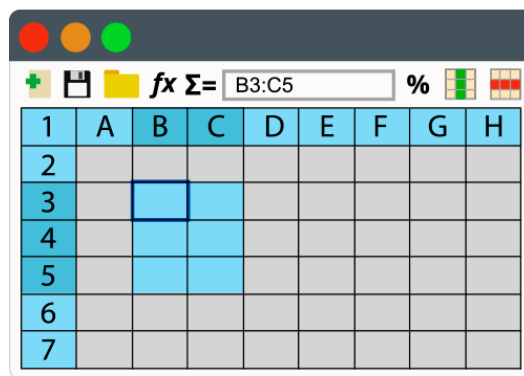


Software Aplicativo:

Seu foco é o usuário final. Normalmente, os erros ou falhas percebidas pelo usuário são identificados nos softwares que se encaixam nesta categoria.

Exemplos:

- Editores de Texto;
- Planilhas;
- Sistemas de Folha de Pagamento;
- Sistemas Comerciais;
- Sistemas de Controle de Estoque;
- Antivírus.



CLASSIFICAÇÃO SEGUNDO A FORMA DE AQUISIÇÃO OU DE DISTRIBUIÇÃO

Antes de entrarmos no assunto Sistema Operacional, é muito importante citarmos outra classificação de software, agora quanto à sua forma de aquisição ou de distribuição:

Software Livre

Conceito criado por Richard Stallman, em 1983, que defendia uma liberdade de uso para utilizar, copiar, distribuir, modificar e melhorar o software, na condição de que o ciclo continuasse, ou seja, todas as melhorias e modificações no software deveriam ser livres.

Muitos desenvolvedores têm usado essa filosofia criando softwares disponíveis livremente para uso, de forma que sua cópia ou redistribuição não caracteriza pirataria. Instituições públicas têm utilizado essa classificação em vista da economia gerada com a compra de softwares.

Software Proprietário

Não é de livre utilização, cópia ou distribuição. Seu uso somente é possível e permitido mediante a compra de licenças de uso, que podem ser por usuário, por máquina ou por empresa. Porém, sempre mediante o investimento financeiro conforme contratado com o fabricante.

O aspecto positivo de um software proprietário se dá pelo fato do fabricante ter a obrigação contratual de oferecer suporte ao aplicativo comprado. Sua cópia ou distribuição sem a compra da licença é considerada pirataria.

CONCEITOS BÁSICOS

Sistema

Conjunto de partes coordenadas que colaboram para a realização de um determinado objetivo.

Essa definição se aplica aos sistemas de computação em que partes (o teclado, a memória, o processador e os dispositivos periféricos) se coordenam para realizar um objetivo comum: computar. Lembre-se que sistema é simplesmente a descrição de um programa de computador.



Exemplo

, O SUS (Sistema Único de Saúde) é um exemplo de um sistema com o objetivo de que partes coordenadas colaborem para cuidar da saúde do cidadão.

ALGORITMO X LINGUAGEM DE PROGRAMAÇÃO X PROGRAMA

Qual é a relação entre esses termos?

Por definição, **algoritmo** é o conjunto de etapas (instruções) finitas, ordenadamente definidas, com o propósito de obter solução para um determinado problema.

Para que um algoritmo seja executado por um computador, é necessário que o processador entenda o que está sendo solicitado. A sequência precisa ser finita, exata e sem duplicidade de interpretação para que a solicitação do algoritmo seja feita com eficiência.

Veja um caso simples de transcrição de um algoritmo para, por exemplo, trocar o pneu de um carro:

```
Inicio
    se <o estepe está vazio> então
        chamar borracheiro
    senão
        levantar o carro
        desparafusar a roda
        remover a roda
        colocar o estepe
        parafusar a roda
        abaixar o carro
    fim se
Fim
```

Perceba que é uma forma de documentar uma sequência de processos não somente em TI. Trata-se de uma ferramenta de uso em diversas áreas de atuação em uma empresa.

Cada passo do algoritmo corresponde a um conjunto de instruções transcritas para a solução de um problema.

Fonte:

Um algoritmo ou um conjunto de algoritmos podem formar um programa.

Esse conjunto de instruções é transformado em uma linguagem própria (linguagem binária ou linguagem de máquina) que será executada em uma máquina.

COMO UM SER HUMANO PROGRAMA EM LINGUAGEM BINÁRIA OU LINGUAGEM DE MÁQUINA?

Sabemos que em computação tudo se baseia no sistema binário (0 e 1).

A solução encontrada foi o desenvolvimento de outras formas de linguagem de programação, que fossem mais próximas do entendimento do programador. Elas são conhecidas como linguagem de [Alto Nível \(glossário\)](#).

De forma contrária, linguagem de [Baixo Nível \(glossário\)](#) são aquelas linguagens mais complexas para o ser humano, em que é mais difícil a compreensão por parte do programador e são necessárias mais ações na programação.

EXEMPLOS DE CÓDIGOS COM A MESMA FUNÇÃO

Veja exemplos de códigos com a mesma função em linguagem de Alto e Baixo Nível. Dessa forma, fica mais fácil visualizar a diferença para um programador.



```
1 print("Hello World!")
```

Exemplo para exibir a mensagem "Olá Mundo" em Linguagem de Alto Nível - Python

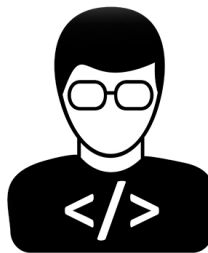
```

1  lea si, string
2  call printf
3  hlt
4  string db "Ola mundo!", 0
5  printf PROC
6      mov AL, [SI]
7      cmp AL, 0
8      je pfend
9      mov AH, 0Eh
10     int 10h
11     inc SI
12     jmp printf
13 pfend:
14     ret
15 printf ENDP

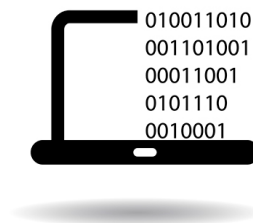
```

Exemplo para exibir a mensagem "Olá Mundo" em Linguagem de Baixo Nível - Assembly

O processo consiste:



No programador escrever o código, em sequências de instruções exatas e de fácil entendimento.



Em seguida, essa sequência precisa ser convertida em linguagem binária, que é entendida e executada pelo processador do computador.

Fonte:

Esses conversores são identificados como programas **compiladores** e **ligadores** (link editor ou linker).

Saiba Mais

, Antes de continuar seus estudos, clique aqui (<https://becode.com.br/linguagens-alto-nivel-x-baixo-nivel/>) para saber mais sobre Linguagens de Alto Nível e Linguagem de Baixo Nível.

Sendo assim, um programa é a formalização de um algoritmo em linguagem inteligível pelo computador (linguagem de máquina). Normalmente, é composto por diversos algoritmos e feito, na sua maioria, com uma tecnologia única de linguagem de programação, que é uma linguagem formal. Por exemplo: Cobol, Pascal, Fortran, Visual Basic, Lisp, Assembly, C++, Delphi, Ada, Java etc.

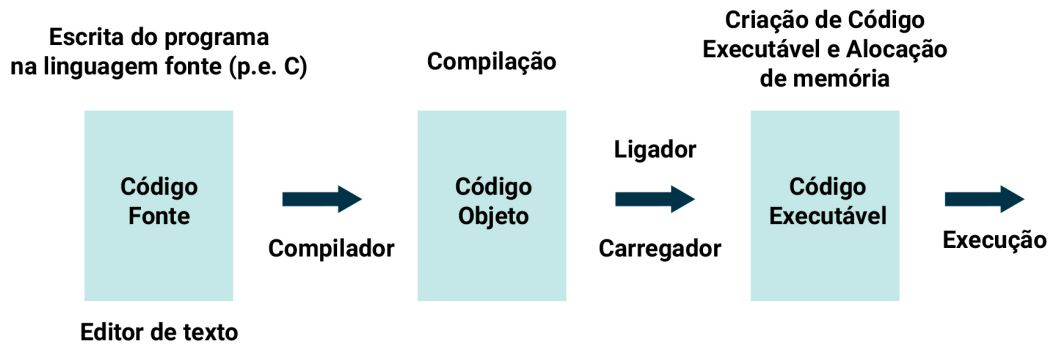
TRADUÇÃO

Como o próprio nome diz, consiste na tradução de um programa escrito em linguagem de Alto Nível para a linguagem binária da máquina correspondente.

Os exemplos de tradutores são:

- Assembler (Montador) da linguagem Assembly e um Compilador da linguagem C++;
- Visual Basic.

É feito o processo de **Compilação**, onde o programa escrito na linguagem fonte é traduzido para linguagem máquina e depois ligado e carregado para ser executado.



INTERPRETAÇÃO

O programa fonte é traduzido e executado, instrução a instrução, de modo interativo.

O **interpretador** traduz cada instrução para uma representação interna e interpreta-a simulando o funcionamento do processador.

Exemplo

, Um exemplo de tradutor pode ser a Máquina Virtual Java, que traduz os byte codes produzidos pela compilação de um programa Java em instruções binárias da máquina onde tal programa está sendo executado.



Vantagens - O ciclo escrita, execução, modificação e atualização é mais rápido.



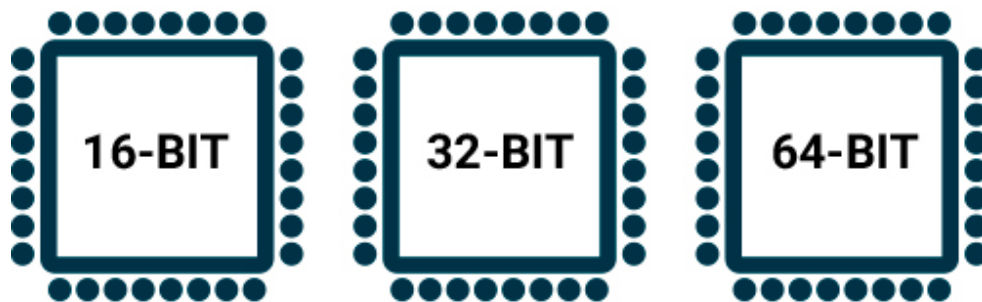
Desvantagens - A execução é mais lenta.

PALAVRA (WORD)

Em Ciência da Computação, palavra (em inglês: *word*) é a unidade natural de informação usada por cada tipo de computador em particular. É uma sequência de bits de tamanho fixo processada em conjunto em uma máquina.

Trata-se de uma característica importante de uma arquitetura de computador, pois reflete diretamente em vários aspectos da estrutura, da performance e de sua operação, indicando a unidade de transferência entre a CPU e memória principal.

Os computadores modernos possuem tamanho de palavra de 16, 32 e 64 bits, o que indica a unidade básica de transferência de dados entre a CPU e a memória principal.



Por isso, é importante ficarmos atentos para, caso tenhamos um equipamento 64 bits, também instalarmos softwares e aplicativos 64 bits, pois os mesmos foram desenvolvidos para usar essa transferência máxima entre CPU e Memória.

Quando instalamos um aplicativo em sua versão 32 bits, estamos diminuindo a capacidade de processamento do nosso equipamento. Por exemplo: Um processador de 32 bits possui palavra de comprimento fixo igual a 32 bits, ou 4 bytes. Já um processador de 64 bits possui palavra de 64 bits ou 8 bytes.

TEMPO DE ACESSO (LATÊNCIA)

Tempo para recuperação/escrever de uma informação na memória. Depende da tecnologia da memória utilizada.

Se formos comparar com uma vitrola, de disco de vinil, seria o tempo gasto desde o momento em que pegamos a agulha até quando colocamos a mesma no início da música que queremos ouvir, mas ainda não começou a tocar.

TEMPO DE RESPOSTA

Mede o desempenho global do sistema. Baseia-se no tempo gasto entre o início da solicitação e a apresentação da resposta.

Engloba o tempo de acesso ou latência. Considerando o mesmo exemplo da vitrola, seria o tempo gasto desde o momento em que pegamos a agulha até o tempo em que a música foi totalmente entregue a nós.

BITS E BYTES

Bits

Sabemos que, com o passar da história, toda informação a ser reconhecida em um computador segue atualmente um modelo de comunicação.

Os computadores conhecidos como digitais somente são capazes de entender informações em código binário (0s e 1s). Esses códigos binários são conhecidos como bits (Binary Digit) e podem assumir somente dois valores: 0 ou 1.

Esses valores são identificados em eletrônica através de variações de tensão. Por exemplo, em um circuito de 3V, uma tensão próxima a 0V (zero volt) representa o bit 0, e uma tensão próxima a +3V (3 volts positivos) representa o bit 1.

Bytes

Os valores 0s e 1s postos isoladamente em um computador não significam praticamente nada de informação. Por isso, os fabricantes de computadores começaram a instituir o Byte (Binary Term) como unidade padrão de representação da informação. Essa unidade era formada por um conjunto de bits.

No início, cada fabricante usava uma padronização. Existiram bytes, por exemplo, formados por 6 bits e 7 bits. Porém, a primeira codificação de 1 byte = 8 bits foi feita pela IBM, com a criação do código EBCDIC, em 1960.

Com o sucesso dos computadores IBM, padronizou-se que 1 byte = 8 bits, surgindo também o código ASCII de 8 bits em 1961. A partir daí, tornou-se padrão que todo byte seria equivalente a um conjunto de 8 bits.

K (KILO), M (MEGA), G (GIGA), T (TERA) E P (PETA)

Esses termos foram criados para indicar valores múltiplos de bits e bytes. São potências de 2 (afinal, estamos falando de uma base binária) e não de 10.

Assim:

Kilo (= K)

Representa 1024 unidades (2^{10}), e não 1000.

Mega (= M)

Representa 1.048.576 unidades (2^{20}) ou 1024.K
(Afinal, $2^{20} = 2^{10} \cdot 2^{10}$).

É isso que nos permite entender o quadro abaixo, relativo às representações das unidades.

1 byte	8 bits
1 kilobyte	1024 byte
1 megabyte	1024 kilobyte
1 gigabyte	1024 megabyte
1 terabyte	1024 gigabyte

Veja como fica a conversão através das potências de 2:

Sufixo	Sufixo
Kilo (K)	$2^{10} = 1.024$
Mega (M)	$2^{20} = 1.048.576$
Giga (G)	$2^{30} = 1.073.741.824$
Tera (T)	$2^{40} = 1.099.511.627.776$
Peta (P)	$2^{50} = 1.125.899.906.843.624$
Exa (E)	$2^{60} = 1.152.921.504.607.870.976$
Zeta (Z)	$2^{70} = 1.180.591.620.718.458.879.424$
Yotta (Y)	$2^{80} = 1.208.925.819.615.701.892.530.176$

CAPACIDADE DE ARMAZENAMENTO X CAPACIDADE DE TRANSFERÊNCIA

Esse tema é interessante e polêmico, pois esclarece muitas dúvidas, principalmente nas representações de bits e bytes.

Quando falamos em armazenamento, normalmente utilizamos representação em Bytes (B maiúsculo).

Exemplos:

1 Kbyte ou 1 KB = 1024 Bytes

1 Mbyte ou 1 MB = 1024 Kbytes

Quando falamos em capacidade de transferência usamos bits (b minúsculo), normalmente indicado em bits por segundo (bps).

Exemplos:

1 Kbit ou 1 Kb = 1024 bits

1 Mbits ou 1 Mb = 1024 Kbits

Perceba que usando os primeiros exemplos de cada item, temos 1 KB (Kbyte) e 1 Kb (Kbit), que são itens completamente diferentes.

Uma vez que sabemos que 1 byte equivale a 8 bits, teríamos:

1 KB (Kbyte)

1 Kb (Kbit)

8 Kb (Kbits)

1 Kb (Kbit)

8 . 1024 . bits

1 . 1024 . bits

8192 bits

1024 bits

Note também que as representações parecem iguais, mas os resultados finais são diferentes, já que de um lado trabalha-se com Bytes e do outro Bits.

É por isso que, quando contratamos um link de internet banda larga para nossa casa, por exemplo, de 10 Mbps, nunca conseguimos uma taxa de download de 10 Megabytes por segundo. Afinal, o link contratado equivale a 10 Megabits por segundo.

ATIVIDADE

1 - Considerando que você possui um link de internet de **10 Mbps**, qual seria a taxa de download em Kbytes por segundo? Quanto tempo, em minutos, você conseguiria fazer o download de um arquivo de 600 Mbytes?

Resposta Correta

2 - Dos aspectos abaixo, referentes à software livre, marque a alternativa INCORRETA:

- ☐ a) Permitido usar
- ☐ b) Permitido compartilhar
- ☐ c) Permitido distribuir
- ☐ d) Permitido modificar
- ☐ e) Permitido vender

Justificativa

3 - Os dispositivos de E/S (Entrada e Saída), ou periféricos, são os equipamentos acoplados ao sistema de computação, que efetuam as funções de entrada e saída. Mouse, Monitor de vídeo e Teclado são, respectivamente, periféricos de:

- ☐ a) Entrada, entrada e saída de dados
- ☐ b) Entrada, saída e entrada de dados
- ☐ c) Saída, entrada e entrada de dados
- ☐ d) Saída, entrada e saída de dados
- ☐ e) Entrada, saída e saída de dados

Justificativa

4 - Qual dos programas abaixo pode ser considerado como software básico?

- ☐ a) Editores de Texto
- ☐ b) Sistemas Operacionais
- ☐ c) Editores de Planilhas
- ☐ d) Sistemas de Folha de Pagamento
- ☐ e) Sistemas Comerciais

Justificativa

Glossário

ALTO NÍVEL

São exemplos de linguagem de Alto Nível:

- Pascal;
- Java;
- C;
- C++;
- Cobol;
- Delphi;
- Python.

BAIXO NÍVEL

Um exemplo de Linguagem de Baixo Nível é a Linguagem Assembly.

1 BYTE = 8 BITS

