**Atividades – Fundamentos de Computação e Programação**

**Parte 1: Conceitos Básicos de Computação**

**1. O que é uma CPU e qual é sua função principal em um computador?**

A CPU é o "cérebro" do computador. Sua principal função é processar dados, executando instruções de programas em um ciclo contínuo de busca, decodificação e execução. Ela também controla o fluxo de informações e gerencia os recursos do sistema.

**2. Qual a diferença entre memória RAM e disco rígido (HD/SSD)? Dê um exemplo do uso de cada um.**

A diferença entre elas são:

* RAM: Memória temporária e rápida, usada enquanto os programas estão abertos.  
   *Exemplo:* Abrir e usar um navegador ou jogo.
* HD/SSD: Armazenamento permanente, guarda arquivos e programas mesmo com o computador desligado.  
   *Exemplo:* Salvar documentos, fotos e instalar programas.

**3. Liste 3 periféricos de entrada e 3 periféricos de saída. Explique a função de um deles.**

### Periféricos de Entrada:

* Teclado
* Mouse
* Scanner

### Periféricos de Saída:

* Monitor
* Impressora
* Caixas de som

### Função do Monitor (Periférico de Saída):

O monitor exibe informações gráficas e textuais geradas pelo computador, permitindo que o usuário veja o que está acontecendo no sistema, como documentos, imagens, vídeos e interfaces de programas.

**4. Cite dois sistemas operacionais e explique brevemente suas diferenças.**

### Windows

* Desenvolvedor: Microsoft  
  Público-alvo: Usuários comuns, empresas e gamers
* Características: Fácil de usar, compatível com muitos programas e hardware, mais vulnerável a vírus.

### Linux

* Desenvolvedor: Comunidade (diversas distribuições)
* Público-alvo: Usuários avançados, servidores e programadores
* Características: Código aberto, maior segurança, menos suscetível a vírus, mais técnico e usado em servidores.

### **Diferenças principais:**

* **Interface e usabilidade:** O **Windows** tem uma interface mais simples e voltada para o usuário comum, enquanto o **Linux** oferece maior controle e personalização, sendo mais técnico.
* **Compatibilidade de software:** O **Windows** tem uma **maior compatibilidade com jogos e programas comerciais**, enquanto o **Linux** é preferido por desenvolvedores e servidores pela sua **estabilidade e segurança**.

**5. O que é um software aplicativo? Dê três exemplos de aplicativos que você utiliza no dia a dia.**

Um software aplicativo é um programa desenvolvido para realizar tarefas específicas para o usuário, como edição de texto, navegação na internet, design gráfico, entre outras atividades.

Exemplos que uso no meu dia a dia:

**Google Chrome** - Para navegação na internet.

**Spotify** - Para ouvir música.

**Netflix** - Para streaming de filmes e séries.

**6. Explique, com suas palavras, como funciona o processo básico de ligar um computador até ele estar pronto para uso.**

Ligar o computador: Quando aperto o botão de ligar, o computador começa a receber energia.

Verificação inicial : O computador faz uma checagem rápida para ver se tudo está funcionando, como a memória e o processador. Se algo estiver errado, ele pode dar um erro na tela.

Carregar o sistema operacional: Depois dessa verificação, o computador começa a carregar o sistema operacional, como o Windows, do HD ou SSD para a memória RAM. Isso é o que permite para podermos utilizar o computador.

Tela inicial: Quando o sistema operacional está carregado, ele mostra a área de trabalho na tela, e ai podemos começar a usar o computador, abrindo programas e acessando arquivos.

**Parte 2: Algoritmos e Automação**

**7. O que é um algoritmo? Crie um algoritmo simples (em linguagem natural) que descreva o processo de escovar os dentes.**

Um algoritmo é uma sequência de passos lógicos ou instruções que descrevem como resolver um problema ou realizar uma tarefa específica.

### **Algoritmo: Escovar os dentes**

1. Pegar a escova.
2. Colocar a pasta de dente na escova (de maneira moderada).
3. Molhe a escova (se necessário).
4. Posicionar a escova na frente dos dentes.
5. Escovar os dentes inferiores, superiores e a parte de trás deles.
6. Escovar a língua.
7. Enxaguar a boca com água.
8. Enxágue a escova de dentes.
9. Guarde a escova de dentes no lugar adequado.

**8. Descreva o processo que representa a tarefa de preparar um sanduíche.**

1. Abrir o pão com o auxílio de uma faca.
2. Passar algo de sua preferência (manteiga, patê, maionese, ketchup).
3. Por o queijo no pão.
4. Por tomate e alface no pão (já lavados e cortados).
5. Por cebolas cortadas junto com o alface e o tomate.
6. Por o bife (já frito e cortado) em cima de tudo e fechar com a outra parte do pão**.**

**9. Escreva um pseudocódigo que descreva como fazer login em uma rede social.**

Início

Exibir "Digite seu nome de usuário:"

Ler nome\_de\_usuario

Exibir "Digite sua senha:"

Ler senha

Verificar se nome\_de\_usuario e senha estão corretos:

Se nome\_de\_usuario e senha são válidos:

Exibir "Login bem-sucedido! Bem-vindo à rede social."

Caso contrário:

Exibir "Erro: nome de usuário ou senha incorretos. Tente novamente."

Repetir o processo de login

Fim

* O **usuário** é solicitado a digitar seu **nome de usuário** e **senha**.
* O **sistema** verifica se as credenciais fornecidas são válidas.
* Se as credenciais forem **corretas**, o usuário é **autenticado** e o login é **bem-sucedido**.
* Caso contrário, o sistema pede para tentar novamente, até que as credenciais estejam corretas.

**10. Dê um exemplo real de tarefa que pode ser automatizada com programação (pode ser algo do seu cotidiano).**

### Como automatizar:

* Lâmpadas inteligentes: Usar lâmpadas que podem ser controladas via smartphone ou assistente virtual (Google Home, Alexa).
* Programação: Configurar horários para ligar e desligar as luzes automaticamente ou acionar as luzes quando seu telefone se conecta ao Wi-Fi.

### Vantagens:

* Conveniência: Lâmpadas ligam/desligam sozinhas.
* Eficiência: Economiza energia.
* Segurança: Simula presença em casa quando você não está.

**Parte 3: Introdução à Programação**

**11. O que é programação e por que ela é importante no mundo atual?**

Programação é o ato de dar instruções para um computador realizar tarefas. É como escrever uma receita detalhada para que a máquina faça exatamente o que você quer.

Ela é crucial hoje porque tudo ao nosso redor envolve software: desde smartphones e aplicativos até carros, eletrodomésticos, sistemas bancários e inteligência artificial. A programação é a base para a inovação, a automação de processos, a solução de problemas complexos e a criação de novas tecnologias que impactam todos os aspectos da nossa vida. Sem programação, o mundo digital como conhecemos simplesmente não existiria.

**12. Explique o que são as estruturas de decisão (if/else). Escreva um exemplo simples em pseudocódigo.**

As estruturas de decisão (também chamadas de estruturas condicionais) são usadas em programação para permitir que o programa tome decisões com base em condições. Elas verificam se uma condição é verdadeira ou falsa e, dependendo do resultado, executam um bloco de código diferente. As estruturas mais comuns são:

* If (Se): Executa um bloco de código se a condição for verdadeira.
* Else (Senão): Executa um bloco de código caso a condição do if seja falsa.
* Elif (Caso contrário, se): Executa um bloco de código se uma condição alternativa for verdadeira, quando há mais de uma possibilidade a ser verificada.

### **Exemplo de pseudocódigo:**

let idade = 20; // A variável "idade" recebe o valor 20

let nome = "João"; // A variável "nome" recebe o valor "João"

if (idade >= 18) { // Verifica se a idade é maior ou igual a 18

console.log(nome + " é maior de idade."); // Se verdadeiro, imprime que a pessoa é maior de idade

} else { // Caso contrário, se a idade for menor que 18

console.log(nome + " é menor de idade."); // Imprime que a pessoa é menor de idade

}

* O programa verifica se a variável **idade** é maior ou igual a 18.
* Se a condição for verdadeira (idade maior ou igual a 18), o programa imprime "Você é maior de idade."
* Se a condição for falsa (idade menor que 18), o programa imprime "Você é menor de idade."

**13. Para que servem os laços de repetição (for/while)? Escreva um exemplo de repetição em pseudocódigo.**

Os laços de repetição (ou loops) são estruturas que permitem que um bloco de código seja executado várias vezes, até que uma condição seja atendida. Eles são extremamente úteis quando você precisa realizar uma tarefa repetitiva, como percorrer uma lista, contar até um número específico, ou repetir uma ação enquanto uma condição for verdadeira.

Existem dois tipos principais de laços de repetição:

* for: Usado quando se sabe de antemão o número de vezes que o código deve ser repetido.
* while: Usado quando o número de repetições não é conhecido de antemão, e o loop continua enquanto uma condição for verdadeira.

### Exemplos de laços de repetição:

1. For: Quando se sabe o número exato de repetições.
2. While: Quando não se sabe o número de repetições, mas sabe-se a condição para continuar repetindo.

### **Exemplo de pseudocódigo com laços de repetição (em JavaScript):**

#### **1. Exemplo com o laço for:**

Este exemplo utiliza o laço **for** para imprimir os números de 1 a 5.

// Laço 'for' em JavaScript para repetir a ação de imprimir números

for (let i = 1; i <= 5; i++) { // A variável 'i' começa em 1, vai até 5, e incrementa 1 a cada repetição

console.log(i); // Imprime o valor de 'i' na tela

}

**14. Crie um dicionário com pelo menos 3 pares de chave:valor representando um aluno (exemplo: nome, idade, nota).**

// Criando um objeto para armazenar informações de alunos

let aluno1 = {

nome: "João", // Chave: "nome", Valor: "João"

idade: 20, // Chave: "idade", Valor: 20

nota: 8.5 // Chave: "nota", Valor: 8.5

};

let aluno2 = {

nome: "Maria", // Chave: "nome", Valor: "Maria"

idade: 22, // Chave: "idade", Valor: 22

nota: 9.0 // Chave: "nota", Valor: 9.0

};

let aluno3 = {

nome: "Carlos", // Chave: "nome", Valor: "Carlos"

idade: 19, // Chave: "idade", Valor: 19

nota: 7.5 // Chave: "nota", Valor: 7.5

};

// Acessando e imprimindo os dados de um aluno

console.log(aluno1); // Exibe as informações do aluno1

**15. Explique a diferença entre uma lista e uma tupla em JavaScript. Quando você usaria cada uma?**

Em JavaScript, listas são representadas por arrays, que são coleções de dados mutáveis, ou seja, podemos adicionar, remover ou modificar seus elementos durante a execução. Já as tuplas, embora não existam como tipo nativo em JavaScript, podem ser simuladas com arrays imutáveis usando Object.freeze(), tornando seus elementos constantes após a criação. Usariam um arraylist quando precisar de uma coleção de dados que possa ser alterada ao longo do tempo, como em uma lista de tarefas. Já uma tupla (ou array imutável) seria útil quando você precisa de uma coleção de valores fixos que não devem ser modificados, como coordenadas ou valores constantes.

**Parte 4: Funções e Organização do Código**

**16. O que é uma função em programação? Para que ela serve?**

Em programação, uma função é um bloco de código com um nome específico que realiza uma tarefa bem definida. Você pode chamar essa função várias vezes em seu programa, sem precisar reescrever o mesmo código repetidamente.

Para que serve?

* Reutilização de código: Evita repetição, tornando o código mais limpo e fácil de manter.
* Organização: Divide o programa em partes menores e lógicas, facilitando o entendimento e a depuração.
* Modularidade: Permite criar "peças" de código independentes que podem ser usadas em diferentes partes do programa ou até em outros projetos.
* Abstração: Esconde os detalhes de implementação de uma tarefa, permitindo que você se concentre no que a função faz, não em como ela faz.

**17. Escreva um exemplo de função com parâmetro e retorno (em pseudocódigo ou linguagem de sua escolha).**

// Função que recebe um parâmetro e retorna o dobro do valor

function dobrarNumero(numero) {

let resultado = numero \* 2; // Processa o número, dobrando-o

return resultado; // Retorna o resultado

}

// Chamando a função e armazenando o retorno

let numeroDobrado = dobrarNumero(5); // Passa o valor 5 para a função

console.log(numeroDobrado); // Exibe 10

**Parâmetro**: numero é o parâmetro que a função dobrarNumero recebe quando chamada.

**Retorno**: A função retorna o valor do número multiplicado por 2.

A função é chamada com o valor 5, e o **retorno** da função é 10, que é armazenado na variável numeroDobrado.

**18. Explique o que significa “escopo de variável”. Dê um exemplo prático.**

O **escopo de variável** refere-se ao contexto em que uma variável é declarada e ao qual ela tem acesso dentro de um programa. Em outras palavras, o escopo define onde a variável pode ser acessada e manipulada. Existem principalmente dois tipos de escopo em JavaScript:

* **Escopo global**: A variável está disponível em qualquer parte do código.
* **Escopo local**: A variável está disponível apenas dentro de uma função, bloco ou estrutura onde foi declarada.

// Variável no escopo global

let nome = "João";

function saudacao() {

// Variável no escopo local

let idade = 25;

console.log(nome); // Acessando a variável do escopo global

console.log(idade); // Acessando a variável do escopo local

}

saudacao();

console.log(nome); // Acessando a variável do escopo global

// console.log(idade); // Erro: 'idade' não é acessível aqui, fora da função

* **Escopo global**: Variáveis podem ser acessadas de qualquer parte do código.
* **Escopo local**: Variáveis são acessíveis apenas dentro do bloco onde foram declaradas (como funções ou blocos de código).

**Parte 5: O Mundo da Tecnologia**

**19. Quais áreas da tecnologia mais chamam sua atenção? Pesquise uma profissão na área e descreva o que ela faz.**

O Administrador de Banco de Dados (DBA) é o profissional responsável por gerenciar e manter os bancos de dados de uma empresa. Ele cuida da segurança, desempenho, backups, restauração e controle de acesso aos dados, garantindo que tudo funcione corretamente e com segurança. É uma profissão essencial em empresas que lidam com grandes volumes de informações, como bancos, e-commerces e instituições públicas.

**20. Quais são três plataformas ou sites que você pode usar para continuar aprendendo programação? Comente qual mais te interessa.**

O YouTube é ótimo para programação porque oferece aulas gratuitas, tutoriais práticos, e explicações visuais que ajudam a entender melhor os conceitos. É acessível no PC e celular, ideal para quem quer aprender no seu ritmo e revisar quantas vezes quiser.

O **Mimo** é um ótimo app para aprender programação de forma **simples e interativa**. Ele é ideal para iniciantes, com **lições curtas**, **exercícios práticos** e uma abordagem parecida com jogos. Funciona tanto no **celular quanto no PC**, sendo perfeito para aprender **HTML, CSS, JavaScript, Python** e outros temas de tecnologia, mesmo com pouco tempo no dia.

O **freeCodeCamp** é bom porque oferece **cursos gratuitos**, completos e práticos em várias áreas da programação. Ele ensina **HTML, CSS, JavaScript, Python, banco de dados, APIs**, entre outros, com **exercícios interativos** direto no navegador. Além disso, você pode ganhar **certificados** ao concluir os módulos e desenvolver **projetos reais** para montar um portfólio. É ideal para aprender do zero e funciona bem tanto no **PC** quanto no **celular**.