

# DEVinHouse [Docentes]

## Módulo 1 - Atividade Avaliativa 2

Aluna: Júlia Caroline Pereira

<b>SUMÁRIO</b>	<b>1</b>
<b>1 INTRODUÇÃO E CONTEXTUALIZAÇÃO</b>	<b>1</b>
<b>2 ESCOLHA DA FERRAMENTA</b>	<b>1</b>
<b>3 CAPACIDADES</b>	<b>1</b>
<b>4 INSTRUMENTO AVALIATIVO</b>	<b>2</b>
<b>5 PROTÓTIPO EXEMPLO DA APLICAÇÃO</b>	<b>8</b>

---

### 1 INTRODUÇÃO E CONTEXTUALIZAÇÃO

A Avaliação Diagnóstica foi desenvolvida com o objetivo de verificar o nível de conhecimento prévio dos alunos em temas fundamentais da Unidade Curricular de Lógica de Programação, visto em momentos anteriores. Este instrumento foi aplicado na plataforma **Quizizz**, que permite um ambiente interativo e dinâmico, adequado ao perfil dos estudantes do curso Técnico em Desenvolvimento de Sistemas. O objetivo é identificar o que os alunos já sabem e ajustar as aulas para atender melhor às necessidades da turma.

### 2 ESCOLHA DA FERRAMENTA

A ferramenta **Quizizz** foi escolhida para implementar a Avaliação Diagnóstica por sua capacidade de promover um ambiente interativo, oferecendo feedback instantâneo aos alunos e permitindo que eles identifiquem rapidamente seus erros. Além disso, a plataforma facilita o acompanhamento dos resultados pelos professores, permitindo uma rápida análise das áreas que precisam ser reforçadas.

### 3 CAPACIDADES

O instrumento avaliativo foi formulado para levantar as seguintes capacidades:

- H1- Utilizar padrões de nomenclatura e convenções de linguagem na codificação de algoritmos.
- H4- Codificar algoritmos na resolução de problemas
- H6- Aplicar métodos e técnicas de programação
- H7- Identificar erros de acordo com o requisito do programa
- H8- Utilizar expressões aritméticas, relacionais e lógicos para sistemas computacionais.

#### 4 INSTRUMENTO AVALIATIVO

Segue abaixo o conteúdo completo da avaliação diagnóstica com as perguntas e alternativas:

ITEM 1
<p><b><u>CAPACIDADE</u></b></p> <ul style="list-style-type: none"> <li>• Aplicar métodos e técnicas de programação</li> </ul>
<p><b>Contexto:</b> No desenvolvimento de algoritmos, é necessário seguir uma sequência lógica para que as informações sejam corretamente manipuladas e apresentadas. Essas fases permitem organizar o fluxo do programa, começando com a captura de dados, a execução das operações e, por fim, a apresentação dos resultados.</p> <p><b>Comando:</b> Quais são as fases de um algoritmo, na ordem correta?</p> <p>a) Entrada, Saída e Processamento  <b>b) Entrada, Processamento e Saída</b>  c) Saída, Processamento e Entrada  d) Processamento, Entrada e Saída</p>

ITEM 2
<p><b><u>CAPACIDADE</u></b></p> <ul style="list-style-type: none"> <li>• Aplicar métodos e técnicas de programação</li> </ul>
<p><b>Contexto:</b> Em um processo de desenvolvimento de <i>software</i>, os fluxogramas são ferramentas importantes para representar a sequência lógica de um algoritmo ou sistema. Cada símbolo tem um propósito específico, e a compreensão desses símbolos é essencial para a criação de fluxogramas claros e precisos.</p>

**Comando:**

Qual símbolo é usado para indicar o início de um fluxograma?

- a) Retângulo
- b) Círculo
- c) Losango
- d) Elipse**

**ITEM 3****CAPACIDADE**

- Aplicar métodos e técnicas de programação

**Contexto:**

No desenvolvimento de algoritmos e programas, é comum a necessidade de armazenar dados temporários ou resultados intermediários durante a execução. Para isso, utiliza-se o conceito de variáveis, que permitem ao programador manipular e guardar valores conforme o programa avança.

**Comando:**

Quando é preciso o uso de uma variável em um programa?

- a) Quando quero deixar o programa mais difícil
- b) Quando preciso guardar um valor**
- c) Quando preciso arrumar um problema
- d) Quando não sei o que fazer

**ITEM 4****CAPACIDADE**

- Aplicar métodos e técnicas de programação

Combine as formas do fluxograma com seu significado:



b	Entrada	c	Início	e	Decisão	d	Processo	a	Saída
---	---------	---	--------	---	---------	---	----------	---	-------

**ITEM 5****CAPACIDADE**

- Aplicar métodos e técnicas de programação

Informe a estrutura mais adequada para cada caso:

- a) Programa que verifica se os números inseridos pelo usuário são pares e, ao final, mostre quantos são.
- b) O programa solicita a entrada de 10 notas para o usuário e realiza a média.
- c) Programa que fornece 5 opções de gênero de filme para o usuário escolher.
- d) O programa solicita um número ao usuário e gera números aleatórios até que seja igual ao inserido pelo usuário.

b	Estrutura de repetição (Para/Faca)
c	Múltipla escolha (Escolha/Caso)
d	Estrutura de repetição (Enquanto/Faca)
a	Estrutura condicional (Se/Entao)

**ITEM 6****CAPACIDADE**

- Utilizar padrões de nomenclatura e convenções de linguagem na codificação de algoritmos.

Explique por que o padrão de nomenclatura de variáveis em código fonte é tão importante.

R:

## ITEM 7

### CAPACIDADE

- Utilizar padrões de nomenclatura e convenções de linguagem na codificação de algoritmos.

Verifique o código abaixo e informe se está utilizando os padrões de nomenclatura de variáveis da forma correta. Explique:

```
/*Faça um programa que solicite n números reais e informe qual é o menor e maior valor dentre eles, bem como a média dos números.
? Exemplo, dados os valores 14, 3, 10, 2, 1, 8, 15 e 5, a saída é "Maior=15, Menor=1, Média= 7,25"*/

#include <stdio.h>
int main()
{
    int numero, maiorNumero=0, menorNumero, somaNumeros=0;
    float mediaNumeros=0;

    for (int i = 0; i < 3; i++){
        printf("Informe um número\n");
        scanf("%d", &numero);

        if (i == 1){
            menorNumero = numero;
        }

        if (numero > maiorNumero){
            maiorNumero = numero;
        }
        if (menorNumero > numero){
            menorNumero = numero;
        }

        somaNumeros = somaNumeros + numero;
    }

    mediaNumeros = somaNumeros/3;

    printf("O maior número é: %d\n", maiorNumero);
    printf("O menor número é: %d\n", menorNumero);
    printf("A média dos números é: %.2f\n", mediaNumeros);
}
```

R:

## ITEM 8

### **CAPACIDADE**

- Utilizar expressões aritméticas, relacionais e lógicas para sistemas computacionais.

#### **Contexto:**

O desenvolvedor agora precisa criar um programa para verificar se três números inteiros formam os lados de um triângulo. Para isso, ele precisa verificar a validade da condição: a soma de dois lados de um triângulo deve ser maior que o terceiro lado.

#### **Comando:**

Qual expressão lógica pode ser utilizada para verificar se os três números podem formar um triângulo?

a) se  $(\text{lado1} + \text{lado2} > \text{lado3})$  e  $(\text{lado1} + \text{lado3} > \text{lado2})$  e  $(\text{lado2} + \text{lado3} > \text{lado1})$

b) se  $(\text{lado1} + \text{lado2} \geq \text{lado3})$  ou  $(\text{lado1} + \text{lado3} \geq \text{lado2})$  ou  $(\text{lado2} + \text{lado3} \geq \text{lado1})$

c) se  $(\text{lado1} + \text{lado2} < \text{lado3})$  e  $(\text{lado1} + \text{lado3} < \text{lado2})$  e  $(\text{lado2} + \text{lado3} < \text{lado1})$

d) se  $(\text{lado1} == \text{lado2})$  ou  $(\text{lado2} == \text{lado3})$  ou  $(\text{lado1} == \text{lado3})$

## ITEM 9

### **CAPACIDADE**

- Utilizar expressões aritméticas, relacionais e lógicas para sistemas computacionais.
- Aplicar métodos e técnicas de programação
- Identificar erros de acordo com o requisito do programa

#### **Contexto:**

Uma empresa de logística está desenvolvendo um sistema para calcular o tempo estimado de entrega de pacotes, baseado na distância e na velocidade média do transporte. Durante os testes, a equipe percebeu que o tempo estimado estava incorreto, pois o sistema estava considerando distâncias negativas como válidas.

```

1 Algoritmo "Tempo de entrega"
2
3 Var
4 // Seção de Declarações das variáveis
5 tempoo: real
6 distancia: real
7 velocidade: real
8 horas: real
9 Inicio
10 // Seção de Comandos, procedimento, funções, operadores, etc...
11 escreval("Informe a distância")
12 leia(distancia)
13
14 escreval("Informe a sua velocidade")
15 leia(velocidade)
16
17 tempoo <- distancia/velocidade * 60
18 escreval("Você irá levar ", tempoo, " minutos para percorrer")
19 escreval(distancia, "km de distância a ", velocidade, "km/h" )
20
21 Finalgoritmo

```

**Comando:**

Considerando o código acima, qual seria a melhor abordagem para resolver esse problema e garantir que apenas distâncias válidas sejam consideradas no cálculo do tempo estimado de entrega?

R:

## ITEM 10

### **CAPACIDADE**

- Aplicar métodos e técnicas de programação

**Contexto:**

Durante o desenvolvimento de *software*, é comum a necessidade de realizar operações repetitivas. Para isso, utilizam-se estruturas de repetição, que permitem executar um bloco de código várias vezes.

**Comando:**

Qual estrutura de repetição é mais adequada quando o número de iterações é conhecido previamente?

- a) while
- b) do-while
- c) for
- d) switch

## 5 PROTÓTIPO EXEMPLO DA APLICAÇÃO

Devido ao uso de algumas funções premium da plataforma **Quizizz**, não foi possível gerar um link público para o quiz. Entretanto, as perguntas foram organizadas em slides, a partir da perspectiva do aluno, disponíveis através do link abaixo:

 Formulário de conhecimento prévio - DevInHouse