



Experimento de Compreensão de Código

Este material faz parte de um experimento com o objetivo de investigar o grau de compreensão de código-fonte por estudantes iniciantes em programação. Essa atividade envolve a análise de trechos de código e a realização de tarefas que exigem leitura e interpretação de programas.

Instruções:

- Essa atividade é individual;
- Essa atividade possui 04 (quatro) exercícios, agrupados em 02 (duas) categorias;
 - Leia cada exercício com atenção;
 - Responda diretamente nas linhas ou espaços indicados;
 - **Os exercícios NÃO estão ordenados por complexidade;**
- Os códigos apresentados nessa atividade estão escritos na linguagem “C”;
- Não pode consultar qualquer material (impresso ou digital);
- Evite conversas durante a realização da atividade;
- Mantenha o celular desligado (ou guardado) durante todo o período da atividade;
- Em caso de dúvida, sinalize para o(a) aplicador(a) do experimento;
- Essa atividade terá duração de até 150 minutos, não podendo haver prorrogação;
- Ao concluir, sinalize para o(a) aplicador(a) para que esse documento seja entregue.

Essa atividade não possui caráter avaliativo (nota), sendo utilizada exclusivamente para fins de pesquisa. Sua participação neste experimento é voluntária e está contribuindo para o desenvolvimento de estratégias pedagógicas mais eficazes no ensino de programação.

Você pode desistir a qualquer momento, sem necessidade de justificativa e sem qualquer prejuízo acadêmico. As respostas serão analisadas de forma anônima e utilizadas apenas para fins de pesquisa. Não escreva seu nome ou qualquer informação que possa identificá-lo(a) nesse documento.

Apenas marque o curso que você está regularmente matriculado.

- | | |
|---|---|
| <input type="checkbox"/> Engenharia de Software | <input type="checkbox"/> Sistemas de Telecomunicações |
| <input type="checkbox"/> Engenharia Elétrica | <input type="checkbox"/> Sistemas para Internet |
| <input type="checkbox"/> Redes de Computadores | <input type="checkbox"/> Outro: _____ |

Muito obrigada! Valéria Cavalcanti



Definições Importantes!

Os erros de um código podem ser classificados em diferentes tipos, por exemplo: sintático e semântico.

O que é um erro SINTÁTICO?

São erros que impedem o código de ser executado, são erros de escrita que ferem as regras da linguagem de programação.

Exemplo:

```
num = (10 + 10/2  
// Faltou fechar os parênteses
```

O que é um erro SEMÂNTICO?

É o código que possui sintaxe correta, roda sem gerar erro, porém não faz o que era esperado. Ou seja, a lógica está errada.

Exemplo:

```
media_aritmetica = 10 + 10 + 10 / 3  
// Faltou isolar a soma dos três valores com parênteses
```

Anote seu Código



Categoria: **Correção de Erros**

Problema: **A**

A seguir, está um programa que deveria ler 04 (quatro) números inteiros (positivos, negativos ou zero) e informar o maior valor entre eles. Mas, não funciona. Apresenta erros! **✗**

```
#include <stdio.h>

int main() {
    int n1, n2, n3, n4;

    scanf("%d", &n1);
    scanf("%d", n2);
    scanf("%d", &n3);
    scanf("%s", &n4);

    if (n1 > n2 && n1 > n3 && n1 > n4 {
        printf("%d\n", n1);
    } else if (n2 > n3 && n2 < n4) {
        printf("%d\n", n2);
    } else if (n3 > n4) {
        printf("%d\n", n3);
    } else {
        printf("%d\n", &n1);
    }
    return 0;
}
```

Analise o código e identifique os erros presentes. Para cada erro encontrado, destaque diretamente no código e indique se é um erro sintático (**SI**) ou semântico (**SE**).

Reescreva o código CORRIGIDO nesse espaço, a seguir:



Categoria: **Correção de Erros**

Problema: **B**

A seguir, está um programa que deveria ler os nomes e as notas de 400 pessoas. Ao final, o programa deveria exibir a média e a quantidade de notas que possuem valor acima dessa média calculada. Mas, não funciona. Apresenta erros! **✗**

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define NUM_ALUNOS 400

int main() {
    char nomes[NUM_ALUNOS][100];
    int notas[NUM_ALUNOS];
    i, qtd = 0;
    float media = 0;

    for (i = 0; i < NUM_ALUNOS; i) {
        printf("Informe o nome: ");
        scanf("%[^\n]", nomes[i]);
        printf("Informe a nota: ");
        scanf("%d", notas[i]);
        media -= notas[i];
    }

    for (i = 0; i < NUM_ALUNOS; i++) {
        if (notas[i] > media) {
            --qtd;
        }
    }
    media /= NUM_ALUNOS;
    printf("Média: %.2f\n", &qtd);
    printf("Quantidade de notas acima da média: %d\n", qtd);
    return 0;
}
```

Analise o código e identifique os erros presentes. Para cada erro encontrado, destaque diretamente no código e indique se é um erro sintático (**SI**) ou semântico (**SE**).

Reescreva o código CORRIGIDO nesse espaço, a seguir:



Categoria: **Enunciado**

Problema: **A**

A seguir, está apresentado um trecho de código.

```
#include <stdio.h>

int main() {
    int vetor[10] = {0,0,0,0,0,0,0,0,0,0};
    int i, x = 0;

    for (i = 0; i < 10; i++) {
        printf("Digite um número: ");
        scanf("%d", &vetor[i]);
    }
    for (i = 0; i < 10; i++) {
        if (vetor[i] > vetor[x]) {
            x = i;
        }
    }
    printf("%d\n", vetor[x]);
    return 0;
}
```

Leia com atenção e elabore um enunciado que poderia ter sido dado ao programador para que ele implementasse esse código.



Categoria: **Enunciado**

Problema: **B**

A seguir, está apresentado um trecho de código.

```
#include <stdio.h>

int main() {
    float lista[8] = {0,0,0,0,0,0,0,0};
    int i;

    for (i = 0; i < 8; i++) {
        printf("Valor: ");
        scanf("%f", &lista[i]);
    }
    for (i = 1; i < 8; i++) {
        if (lista[i] < lista[i - 1]) {
            break;
        }
    }
    printf("%d\n", i == 8);
    return 0;
}
```

Leia com atenção e elabore um enunciado que poderia ter sido dado ao programador para que ele implementasse esse código.