

NOME COMPLETO: DRIELLE VIANA VIEIRA

Matrícula:

TURMA: 2021.1

Questões Objetivas – Aula 07

1. O que o código abaixo retorna?

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

```
int compara(int a, int b);
```

```
int main(){
```

```
    int n1, n2, res;
```

```
    printf("Digite dois valores separados por espaço:\n");
```

```
    scanf("%d %d", &n1, &n2);
```

```
    res = compara(n1, n2);
```

```
    printf("o menor número eh: %d\n", res);
```

```
    return 0;
```

```
}
```

```
int compara(int a, int b){
```

```
    if (a > b) {
```

```
        return b;
```

```
    }else{
```

```
        return a;
```

```
    }
```

```
}
```

- a) Retorna o maior valor.
- b) Retorna uma comparação entre os dois valores.
- c) **Retorna o menor valor**
- d) Retorna 0
- e) O programa contém erro e não será executado.

2. Na linguagem de programação em C, as funções permitem a criação de programas em módulos, em que todas as variáveis, que são descritas nas definições de função, são locais, pois são conhecidas apenas na função em que são definidas. Cada biblioteca-padrão tem um cabeçalho que contém os protótipos de função para todas as funções nessa biblioteca, assim como definições de vários tipos de dados e constantes que são necessárias para estas funções. Uma dessas bibliotecas tem a seguinte explicação: contém as definições comuns de tipo usadas pela C para realizar cálculos. Assinale-a.
- a. `<math.h>`
 - b. `<stddef.h>`**
 - c. `<locale.h>`
 - d. `<assert.h>`
3. A respeito das linguagens de programação orientadas a objeto, é correto afirmar que:
- a. O polimorfismo é o conceito utilizado para definir funções de nomes diferentes porém com códigos iguais.
 - b. O método conhecido com construtor é sempre executado quando um objeto é criado.
 - c. O uso de herança aumenta a eficiência do programa, mas acaba gerando códigos repetitivos.
 - d. O construtor default é aquele que recebe, no mínimo, um argumento como parâmetro.**
4. O conceito de Programação Orientada a Objetos que permite construir objetos especializados utilizando características de objetos mais generalistas, possibilitando reuso de código à medida que os atributos e métodos de classes já existentes podem gerar novas classes mais específicas, é o de:
- a. Associação de classes
 - b. Herança**
 - c. Encapsulamento
 - d. Classe abstrata
 - e. Poliformismo