

Evaluación diagnóstica

Nombre: Estalyn Licoy

NRC: 1323

Fecha: 07/11/2024

1: ¿Cuál será la salida del siguiente código en C++?

```
int a = 5;
```

```
int b = 10;
```

```
a = a + b;
```

```
b = a - b;
```

```
a = a - b;
```

a) 5 10

b) 10 5

c) 15 5

d) 0 0

```
std::cout << a << " " << b;
```

2: Corrige el siguiente fragmento de código

```
int x = 10;
```

R:

```
int x = 10;
```

```
while (x > 0);
```

```
while (x > 0) {
```

```
std::cout << x << " ";
```

```
std::cout << x << " ";
```

```
x--;
```

```
x--;
```

```
}
```

3: Complete el siguiente código para calcular la suma de los primeros 100 números naturales.

```
#include <iostream>
```

```
int main () {
```

```
    int suma = 0;
```

```
    for (int i = 1; i <= 100; i++) {
```

```
        suma += i;
```

```
    }
```

```
    std::cout << "La suma es: " << suma << std::endl;
```

```
    return 0;
```

```
}
```

ESTILO

4.- ¿Cuáles son las etapas para resolver un problema en programación?

Explica cada una.

- 1.- **Comprensión del problema:** analizar que se debe resolver y que restricciones tiene el problema
- 2.- **Diseño del algoritmo:** Planificar como será la lógica de los algoritmos para poder programar
- 3.- **Codificación:** Traducir el diseño al lenguaje de programación es decir pasar del algoritmo a la programación
- 4.- **Pruebas y depuración:** Verificar el funcionamiento del programa y corregir errores
- 5.- **Documentación y mantenimiento:** Crear manuales para mantener el código actualizado.

5.- ¿Cuál de las siguientes opciones es una característica clave de POO?

- a) Reutilización del código mediante la herencia.
- b) Separación entre lógica y presentación
- c) Ejecución secuencial
- d) Uso de estructuras de datos primitivas

6.- ¿Qué es UML y cómo ayuda a modelar sistemas orientados a Objetos? Además, menciona y explica los cuatro principales fundamentos de la POO.

- 1.- **Encapsulamiento:** Protege los datos de una clase agrupando atributos y métodos, controlados con su acceso con modificadores de visibilidad
- 2.- **Herencia:** Permite que una clase de tipo hija reutilice atributos y métodos de la clase anterior o superior creando una jerarquía
- 3.- **Polimorfismo:** Habilidad de usar métodos con el mismo nombre pero diferente uso o comportamiento
- 4.- **Abstracción:** Oculta detalles de clases ya sean internas y muestra solo las funciones importantes

7. ¿Qué es un lenguaje de bajo y alto nivel

Alto nivel: Son lenguajes que se aproximan al lenguaje humano como Python, que son fáciles para desarrollo de software

Bajo nivel: Son lenguajes más cercanos al lenguaje de máquina son diseñados para interactuar con hardware directamente y es más complejo.