

2--Tipos de instrucciones. Identificadores. Variables y constantes

Resumen del Tema

Tipos de Instrucciones en Programación

En programación, las instrucciones son las líneas de código que el ordenador ejecuta para realizar tareas específicas. Estas pueden clasificarse en varias categorías, incluyendo:

- Instrucciones de Asignación: Se utilizan para asignar valores a las variables (e.g., $x = 5$).
- Instrucciones de Control de Flujo: Dirigen el flujo de ejecución del programa, como las instrucciones condicionales (`if, else`) y los bucles (`for, while`).
- Instrucciones de Entrada/Salida: Permiten al programa interactuar con el usuario o con otros sistemas, como leer del teclado o escribir en la pantalla.

Identificadores

Los identificadores son los nombres que se dan a los elementos como variables, funciones, clases, entre otros, en un programa. Estos deben ser únicos y usualmente deben seguir ciertas reglas específicas dependiendo del lenguaje de programación, como empezar con una letra y no incluir espacios.

Variables y Constantes

- Variables: Son símbolos que representan y almacenan datos que pueden cambiar durante la ejecución del programa. Una variable tiene un tipo de dato asociado, que define el tipo de valor que la variable puede almacenar (e.g., entero, decimal, texto).
- Constantes: A diferencia de las variables, las constantes retienen valores que no cambian a lo largo de la ejecución del programa. Se utilizan para almacenar valores que se saben que no necesitarán modificarse, como el valor de pi (π).

Cuestionario

1-¿Qué es una instrucción en el contexto de la programación?

- A) Un error en el código
- B) Una línea de código que realiza una tarea específica
- C) Una consulta en una base de datos

2-Menciona y describe dos tipos de instrucciones de control de flujo:

- A) `if` y `for`: `if` ejecuta código basado en una condición, `for` repite código un número específico de veces
- B) `print` y `input`: ambos son usados para procesamiento de texto
- C) `delete` y `remove`: ambos eliminan elementos de una estructura de datos

3-¿Qué diferencia existe entre una variable y una constante?

- A) No hay diferencia
- B) Una variable puede cambiar su valor, una constante no
- C) Una constante puede cambiar su valor, una variable no

4-Da un ejemplo de una instrucción de asignación:

- A) `if x > 10`
- B) `x = 5`
- C) `print(x)`

5-¿Cuál es la función de los identificadores en un programa?

- A) Determinar el tiempo de ejecución del programa
- B) Asignar un nombre a las variables y funciones
- C) Guardar datos en la memoria del ordenador

6-Explica con un ejemplo, ¿qué es una instrucción de entrada/salida?

- A) `x = 5`: Asigna el valor 5 a x
- B) `print("Hello World")`: Muestra el mensaje "Hello World"
- C) `if x > 0`: Verifica si x es mayor que 0

7-¿Por qué es importante seguir las reglas de nomenclatura para los identificadores en programación?

- A) Para hacer el código más rápido
- B) Para evitar errores de compilación
- C) Para que el código sea más fácil de leer y mantener

8-Describe un escenario en el que sería más adecuado usar una constante en lugar de una variable:

- A) Cuando el valor cambia con frecuencia
- B) Cuando el valor representa una configuración que podría cambiar
- C) Cuando el valor no cambia, como el número de días en una semana

9-¿Qué tipo de instrucción utilizarías para repetir un bloque de código varias veces?

- A) `if`
- B) `for`
- C) `return`

10-¿Qué deberías considerar al elegir el nombre de una variable o constante?

- A) El nombre debe ser lo más corto posible
- B) El nombre debe ser descriptivo y seguir las convenciones de nomenclatura
- C) El nombre debe contener números y caracteres especiales para distinguirlo fácilmente