



PROGRAMACIÓN ESTRUCTURADA

Fundamentos de Programación

Peñaloza Castañeda Luna Quetzalli

Código de estudiante: 221618438

Maestra: Sánchez Rosario Patricia

Horario: Martes y Jueves 11:00 am – 12:55 pm

Viernes 30 de Agosto de 2024

NRC: 200274

¿Qué es la programación estructurada?

La programación estructurada es una metodología de desarrollo de software diseñada para mejorar la calidad del código y reducir significativamente el tiempo de desarrollo. Se basa en el principio de descomponer un programa en partes mas pequeñas y manejables, conocidas como subrutinas o funciones. Este enfoque facilita la comprensión, modificación y mantenimiento del código, permitiendo a los desarrolladores manejar programas complejos de manera más eficiente.

Esta metodología nació con el propósito de simplificar la vida de los programadores, especialmente en las fases de mantenimiento y mejora de software. En 1966, Corrado Böhm y Giuseppe Jacopini formularon el teorema del programa estructurado, el cual demostró que cualquier programa podía ser escrito utilizando únicamente tres estructuras de control básicas: secuencias, condiciones y repeticiones. Esta proposición tuvo un impacto significativo, ya que no solo permitió la construcción de programas mas rápidos y sencillos, sino que también redujo la complejidad de las pruebas y la depuración de los mismos.

En la programación estructurada, las instrucciones se ejecutan de manera secuencial, siguiendo un orden lógico de principio a fin. Este enfoque se aplica en diversos lenguajes de programación, como Python, la familia C, entre otros.

En este tema encontraremos muchos elementos que es importante definir, entonces los describiré a continuación:

Variables y tipos de datos

En la programación, una variable es un espacio en la memoria que se utiliza para almacenar valores o datos. Los tipos de datos determinan el tipo de información que una variable puede contener. Entre los tipos de datos más comunes se encuentran: *Enteros (int)*: Representan números enteros.

Cadenas (str): Representan secuencias de caracteres.

Flotantes (float): Representan números racionales.

Booleanas (bool): Tienen dos valores posibles, como verdadero o falso.

Operadores

Los operadores son símbolos que indican al compilador o interprete que debe realizar una operación específica. Existen, de igual manera, diferentes tipos de operadores, entre ellos:

Operadores aritméticos: Se utilizan para realizar operaciones matemáticas básicas como suma, resta, multiplicación y división.

Operadores lógicos: Evalúan expresiones booleanas devolviendo valores de verdadero o falso.

Operadores relacionales: Comparan dos valores y devuelven un valor booleano según la relación entre ellos.

Estructura de control de flujo

Las estructuras de control de flujo permiten dirigir la ejecución del programa en función a condiciones específicas y son la parte central de la programación estructurada. Las principales son:

Estructura secuencial: Las instrucciones se ejecutan en un orden lineal, de principio a fin.

Estructura condicional: Utiliza estructuras como 'if-else' para tomar decisiones basadas en condiciones. Dependiendo del valor de una expresión booleana se ejecuta una u otra sentencia.

Estructura iterativa: Permite repetir la ejecución de un bloque de código mientras se cumpla una condición booleana. Los ciclos o bucles 'while' y 'for' son ejemplos de estructuras iterativas.

Con estas tres estructuras básicas es posible desarrollar la mayoría de aplicaciones y programas. En la programación estructurada, estas estructuras suelen combinarse, por ejemplo, una estructura secuencial puede estar anidada dentro de una estructura condicional o iterativa, creando un flujo de control más complejo y flexible.

Beneficios de la programación estructurada

Simplicidad: La estructura secuencial y lógica elimina la necesidad de rastrear complejos saltos entre líneas de código

Claridad: Los programas tienen una estructura clara y comprensible facilitando la lectura y el mantenimiento.

Facilidad para depuración: Identificar y corregir fallos es mas sencillo gracias a la organización del código.

Reducción de costos: El mantenimiento es menos costoso porque las modificaciones son fáciles de implementar en estructuras claramente definidas

Mejora del rendimiento: Los programadores pueden trabajar de manera más eficiente, lo que se traduce en un aumento en su productividad.

Bibliografía

1. Grupo Docente ISCyP. (s/f). *Programación Estructurada*. Webs.um.es. Recuperado el 30 de agosto de 2024, de <https://webs.um.es/ldaniel/iscyp17-18/10-programacionEstructurada.html>
2. Tecnológica, I. [@informaciontecnologica1836]. (s/f-a). *QUE ES LA PROGRAMACIÓN ESTRUCTURADA | La programación que NECESITAS UTILIZAR*. Youtube. Recuperado el 30 de agosto de 2024, de <https://www.youtube.com/watch?v=JOH4Nc9zzxY>
3. Tecnológica, I. [@informaciontecnologica1836]. (s/f-b). *TODOS los TEMAS de PROGRAMACION ESTRUCTURADA ¿en 8 minutos?* Youtube. Recuperado el 30 de agosto de 2024, de <https://www.youtube.com/watch?v=b4w67q7mhQY>
4. (S/f-a). Unir.net. Recuperado el 1 de septiembre de 2024, de <https://unirfp.unir.net/revista/ingenieria-y-tecnologia/programacion-estructurada/>
5. (S/f-b). Mheducation.es. Recuperado el 1 de septiembre de 2024, de <https://www.mheducation.es/bcv/guide/capitulo/8448148703.pdf>