**REFUERZO DE TEMAS Y CONCEPTOS**

1. **¿Qué es un computador?**

Una computadora es un dispositivo electrónico que procesa información mediante programas y circuitos. Puede recibir, almacenar y ejecutar datos para realizar diversas tareas, como cálculos, edición de textos o navegación por internet.

**ANALOGIA**

* El procesador (CPU) es el chef, que sigue recetas (instrucciones) para preparar platos (tareas).
* La memoria RAM es la encimera, donde se colocan los ingredientes y se trabaja en ellos temporalmente. Cuanto más grande sea, más platos puede preparar al mismo tiempo.
* El disco duro es la despensa, donde se almacenan los ingredientes (datos y programas) hasta que se necesiten.
* La pantalla y el teclado son como la mesa y los utensilios de cocina, que permiten interactuar con el chef.

1. **¿Qué es un programa?**

Un programa es un conjunto de instrucciones que una computadora sigue para realizar una tarea específica, como escribir un documento, navegar por internet o reproducir música.

**ANALOGIA**

Un programa es como una receta de cocina:

* La receta (programa) contiene instrucciones detalladas.
* El chef (computadora) sigue la receta paso a paso.
* Los ingredientes (datos) se procesan según las instrucciones.
* El plato final (resultado) depende de la correcta ejecución de la receta.

1. **Lógica de Programación – Pensamiento estructurado para resolver problemas.**

La lógica de programación es el conjunto de reglas y técnicas que permiten estructurar soluciones a problemas mediante algoritmos y código. Se basa en el pensamiento lógico y estructurado para diseñar instrucciones claras que una computadora pueda ejecutar.

**ANALOGIA**

Imagina que quieres organizar un viaje:

Definir el destino → Identificar el problema a resolver.

Planificar la ruta → Diseñar el algoritmo con pasos lógicos.

Reservar hoteles y transporte → Estructurar los datos y recursos necesarios.

Ejecutar el viaje → Implementar el código y seguir las instrucciones.

Resolver imprevistos → Depurar errores y optimizar el proceso.

1. **Algoritmos**

Un algoritmo es un conjunto de pasos ordenados y finitos que se siguen para resolver un problema o realizar una tarea. Es la base de la programación, ya que las computadoras ejecutan instrucciones basadas en algoritmos.

**ANALOGIA**

Un algoritmo es como una receta de cocina:

Lista de ingredientes → Datos de entrada.

Pasos detallados → Secuencia de instrucciones.

Plato final → Resultado esperado.

**5. Variables**

Son espacios en la memoria que almacenan datos como números o texto.  
**Ejemplo:** nombre = "Ana" almacena el nombre "Ana".  
**Analogía:** Como una caja etiquetada donde guardas cosas específicas.

**6. Tipos de datos**

Diferentes formas en que se almacenan valores, como números, texto o valores booleanos (verdadero/falso).  
**Ejemplo:** Un número de teléfono es un dato tipo texto, aunque sean números.  
 **Analogía:** Como diferentes tipos de envases para diferentes líquidos.

**7. Operadores**

Símbolos que realizan operaciones matemáticas o lógicas.  
**Ejemplo:** 5 + 3 = 8 usa el operador +.  
**Analogía:** Como usar una calculadora para sumar o restar.

**8. Estructuras de control**

Permiten tomar decisiones en un programa, como if y else.  
**Ejemplo:** Si llueve, llevas paraguas; si no, sales sin él.  
**Analogía:** Como un semáforo que cambia el flujo de los autos.

**9. Bucles**

Permiten repetir instrucciones varias veces.  
**Ejemplo:** Contar del 1 al 10 con un for.  
**Analogía:** Como una lavadora que repite ciclos de lavado.

**10. Entrada y salida**

Permiten interactuar con el usuario, como escribir en pantalla o recibir datos.  
**Ejemplo:** Un cajero automático recibe tu PIN y muestra saldo.  
**Analogía:** Como hablar y escuchar en una conversación.

**11. Funciones básicas**

Permiten reutilizar código sin repetirlo.  
**Ejemplo:** Una función suma(a, b) que suma dos números.  
**Analogía:** Como una plantilla para hacer cartas.

**12. Primer lenguaje (Python)**

Un lenguaje de programación con sintaxis simple.  
**Ejemplo:** print("Hola, mundo") en Python.  
**Analogía:** Como aprender un nuevo idioma para comunicarse.

**13. Comentarios en código**

Notas en el código para explicar su funcionamiento.  
**Ejemplo:** # Esto imprime un mensaje.  
**Analogía:** Como notas en un libro para recordar puntos clave.

**14. Errores comunes**

Tipos de errores en código: sintaxis (gramática) y ejecución (fallos al correr el programa).  
**Ejemplo:** Olvidar un : en Python genera error.  
**Analogía:** Como un error de ortografía en una carta.

**15. Depuración**

Proceso para identificar y corregir errores en un programa.  
**Ejemplo:** Usar print() para ver qué está fallando.  
**Analogía:** Como revisar un ensayo antes de enviarlo.

**16. Compiladores vs. intérpretes**

El compilador traduce todo el código antes de ejecutarlo, el intérprete lo hace línea por línea.  
**Ejemplo:** C usa compiladores; Python usa intérprete.  
**Analogía:** Como traducir un libro completo (compilador) vs. traducir en tiempo real (intérprete).

**17. Cadenas de texto**

Manipulación de texto dentro del código.  
**Ejemplo:** "Hola" + " mundo" → "Hola mundo".  
**Analogía:** Como unir bloques de palabras para formar frases.

**18. Hardware básico**

Partes físicas de un computador: CPU, memoria, disco duro.  
**Ejemplo:** El procesador es el "cerebro" del computador.  
**Analogía:** Como el motor de un carro.

**19. Software**

Programas que permiten que el hardware funcione.  
**Ejemplo:** Windows, Linux, juegos, navegadores.  
**Analogía:** Como las reglas de un juego de mesa.

**20. Sistemas operativos**

Software que gestiona los recursos del computador.  
**Ejemplo:** Windows, macOS, Linux.  
**Analogía:** Como el director de una orquesta organizando músicos.