

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
José Sánchez	1	Pichardo/TMC-101	22/5/2025

Title: Capítulo 1 - Fundamentos de Programación

Keyword	Topic:
<ul style="list-style-type: none"> - Problema - Algoritmo - Pasos - Procedimiento - Acción - Finitud 	<p>Problemas y algoritmos</p> <p>Notes:</p> <ul style="list-style-type: none"> • Los humanos usan los algoritmos para resolver problemas cotidianos. • Formalmente, un algoritmo es un conjunto de pasos para alcanzar un resultado o resolver un problema. • A menudo aplicamos algoritmos de forma automática sin enumerar los pasos.
Questions	
<p>¿Qué es un algoritmo?</p> <p>¿Cuándo usamos algoritmos inconscientemente?</p>	<ul style="list-style-type: none"> • Consta de 3 secciones principales: Datos de entrada, procesamiento e impresión de resultados.

Summary: Este tema introduce el concepto de algoritmo como una serie de pasos o acciones que realizamos para resolver problemas, tanto en la vida diaria, como en computación.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
José Sánchez	2	Pichardo/TMC-101	22/5/2025

Title: Capítulo 1 - Fundamentos de Programación

Keyword - Diagrama de flujo - Representación gráfica - Algoritmo - Símbolos - Reglas - ISO - ANSI	Topic: Diagrama de flujo Notes: • Es la representación gráfica de un algoritmo. Muestra los pasos o procesos para resolver un problema de forma esquemática. • Se usan símbolos estandarizados (según ISO y ANSI) para representar diferentes acciones, como inicio/fin, lectura, proceso, decisión, decisión múltiple, impresión y correctores. • Las reglas para construir diagramas de flujo incluyen tener un inicio y un fin, usar líneas rectas para el flujo, conectar las líneas a los símbolos, construirlos de arriba a abajo y de izquierda a derecha, y usar notación independiente del lenguaje de programación.
Questions ¿Qué representa un diagrama de flujo? ¿Por qué es importante construir correctamente un diagrama de flujo?	

Summary: El diagrama de flujo es una forma gráfica de visualizar un algoritmo, detallando los pasos para resolver un problema. Su correcta elaboración es esencial para la posterior codificación del programa. Se rige por reglas y usa símbolos estandarizados.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
José Sánchez	3	Pichardo/TMC-101	22/6/2025

Title: Capítulo 1- Fundamentos de Programación

Keyword	Topic:
<p>Datos simples</p> <p>Datos estructurados</p> <ul style="list-style-type: none"> - Enteros - Reales - Caracteres - Registros - Identificadores - Asignación 	<p>Eso de datos, identificadores y constantes</p> <p>Notes:</p> <ul style="list-style-type: none"> • Los datos pueden ser simples o estructurados. • Los simples ocupan solo una casilla de memoria. • Los estructurados hacen referencia a un grupo de casilla de memoria. • Los identificadores se usan para nombrar casillas de memoria para almacenar datos.
Questions	
<p>¿Qué tipos de datos existen?</p> <p>¿Cómo se nombran los datos?</p> <p>¿Cuál es la diferencia entre una constante y una variable?</p>	<ul style="list-style-type: none"> • Las constantes no cambian su valor durante la ejecución del programa. • Las variables pueden cambiar su valor durante la ejecución de un programa. • Reciben un valor a través de la asignación.

Summary: Los datos se clasifican en simples y estructurados, almacenados en memoria con identificadores. Las constantes mantienen su valor, mientras que las variables pueden cambiarse mediante asignación.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
José Sánchez	4	Pichardo/TMC-101	22/5/2025
Title: Capítulo 1 - Fundamentos de Programación			

Keyword	Topic:
<p>Tabla de verdad</p> <ul style="list-style-type: none"> - Conjunción (&) - Disyunción () - Jerarquía - Lógicos - Negación (!) - Decremento - Incremento 	<p>Operadores</p> <p>Notes:</p> <ul style="list-style-type: none"> • Los operadores son necesarios para realizar operaciones y se clasifican principalmente en aritméticos, relacionales y lógicos. • Los operadores aritméticos permiten realizar cálculos numéricos entre operandos, mientras que los relacionales se usan para comparar dos operandos. • Los operadores lógicos permiten formular condiciones complejas a partir de condiciones simples.
Questions	
<p>¿Cómo se interpreta el valor numérico (1-0) de una expresión lógica en el contexto de la toma de decisiones?</p>	<p>La evaluación de expresiones que contienen múltiples operadores se rige por una jerarquía de prioridades, donde los paréntesis tienen la más alta.</p>

Summary: Este tema aborda los operadores usados en el lenguaje C para realizar operaciones, clasificándolos en aritméticos, relacionales y lógicos. Se presentan operadores específicos de C como los simplificados, incremento y decremento y el operador coma. Se enfatiza en la jerarquía de operadores y el orden de evaluación como elementos clave.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
José Sánchez	5	Richardo/TMC-101	22/6/2026

Title: Capítulo 1 - Fundamentos de Programación

Keyword Diagrama de flujo - Esquemática - Algoritmo - Cálculo	Topic: Construcción de diagramas de flujo Notes: • La construcción de diagramas de flujo requiere un razonamiento estructurado y flexible para diseñar soluciones a problemas. • El diagrama de flujo representa la solución detallada del problema y sirve como guía para la posterior escritura del programa. • A través de los ejemplos, se demuestra la aplicación práctica de los conceptos teóricos sobre algoritmos y diagramas de flujo.
Questions ¿Cómo se relacionan las secciones de entrada, procesamiento y salida de un algoritmo con los símbolos usados en un diagrama de flujo?	

Summary: Este tema se centra en la aplicación práctica de la teoría de algoritmos y diagramas de flujo. Destaca que la construcción del diagrama es la parte clave del diseño de la solución. Presenta ejemplos resueltos de problemas con sus respectivos diagramas de flujo, ilustrando cómo se traducen los pasos lógicos del algoritmo a una representación gráfica.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
José Sánchez	6	Pichardo / TMC-101	22/5/2025

Title: Capítulo 1 - Fundamentos de Programación

Keyword	Topic:
<ul style="list-style-type: none"> - Programa - Computadora - Lenguaje de programación - C - Formato de variables - Bloque - Main - Reglas semánticas 	<p>Programas</p> <p>Notes:</p> <ul style="list-style-type: none"> • Un programa es un conjunto de instrucciones para la computadora, escrito en un lenguaje de programación como C, y generalmente basado en el diseño previo de un diagrama de flujo. • El lenguaje C tiene reglas sintácticas (formación de instrucciones) y semánticas (significado de instrucciones) y es de tipo estructurado.
Questions	
<p>¿Cuál es la función específica del símbolo & cuando se usa con la instrucción scanf?</p>	<ul style="list-style-type: none"> • Los programas en C incluyen bibliotecas, como <code>stdio.h</code> para funciones de entrada/salida (<code>printf</code>, <code>scanf</code>), y la ejecución comienza en la función <code>main</code>. • Para la entrada y salida de datos, es obligatorio especificar el formato de las variables (<code>%d</code>, <code>%f</code>, etc), usando el operador de dirección & con <code>scanf</code>, y se pueden usar caracteres de control (<code>\n</code>, <code>\t</code>) para formatear la impresión.

Summary: Este tema define un programa como las instrucciones que sigue la computadora, escritas en un lenguaje como C, basándose en C, incluyendo la necesidad de incluir bibliotecas (`stdio.h`, `math.h`), la función principal `main`, y el uso de bloques {} y punto y coma ;. Se detallan los caracteres de control y los especificadores de formato.