

Title: Algoritmo, diagramas de flujo y programas en C

Keyword

Problema
Algoritmo
Solución
Analizar
Pasos
Verificar
Precisión
Fin

Topic: Problemas y algoritmos

Notes: Formalmente definimos un algoritmo como un conjunto de pasos, procedimientos o acciones que nos permiten alcanzar un resultado o resolver un problema.

Las etapas para resolver un algoritmo son: 1) Analizar el problema; 2) Construir un algoritmo para resolver dicho problema; 3) Verificar que el algoritmo funcione correctamente.

Por otro lado las características que poseen los algoritmos son: 1) Precisión, los pasos que se deben seguir deben ser claros; 2) Determinismo, dado que los datos de entrada sean idénticos siempre arrojará el mismo resultado; 3) Finitud, debe tener un comienzo y un fin, no puede ser infinito.

Questions

¿Qué tan desarrollado puede ser el problema?

¿Se puede subdividir un algoritmo?

¿Hay ocasiones que el resultado sea distinto?

Consta de tres secciones o módulos principales:

1) Datos de entrada, 2) Procesamiento de los datos y 3) Impresión de resultados.

Al seguirse las tres secciones correctamente el algoritmo cumplirá su propósito de resolver el problema planteado.

Summary:

Los algoritmos son las acciones requeridos para llegar a la solución de un problema. Este problema debe presentarse de tal manera que su solución sea posible, para entonces de forma clara y precisa plantear los pasos que se seguirán para su solución. Puede tener bastante posible soluciones pero solo se deben presentar una por algoritmo.

Title: Algoritmo, diagramas de Flujo y programas en C









Keyword

Flujo
Pasos
Figuras
Programa
Construcción
Problema

Topic: Diagramas de Flujo

Notes: El diagrama de Flujo muestra gráficamente los pasos o procesos a seguir para alcanzar la solución de un problema. Es importante se construya correctamente ya que a partir de éste se escribe el programa en un determinado lenguaje de programación.

Los símbolos de este son los siguientes:

-  - Marca el inicio y el fin del diagrama de Flujo.
-  - Se utiliza para introducir los datos de entrada.
-  - Representa un proceso donde se colocan asignaciones.
-  - Se utiliza para representar una decisión de "Si" y "No".
-  - Se almacena un selector y dependiendo su valor se sigue por una rama.
-  - Expresan la dirección del Flujo del diagrama.
-  - Expresa conexión dentro de una misma página.
-  - Representa conexión entre páginas diferentes.

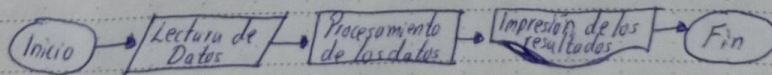
Questions

¿Qué colores pueden tener las Figuras?

¿Pueden haber casillas varias?

¿Puedo obviar utilizar algunos símbolos?

Para su construcción se siguen las siguientes etapas:



Summary:

Un diagrama de Flujo es la forma gráfica que tenemos para representar los pasos de un algoritmo para la resolución de un problema planteado. Este consta de distintas Figuras que cada una representa o sirve para acción específica. Es sumamente importante que este se construya siguiendo el orden establecido.

Title:

Algoritmo, diagrama de Flujo y programas en C

Keyword

Preglas
Construcción
Lineas
Diagrama
Correcto

Topic: Preglas para la construcción de diagramas de Flujo y ejemplo

Notes: Las reglas para la correcta construcción de diagramas de Flujo son:

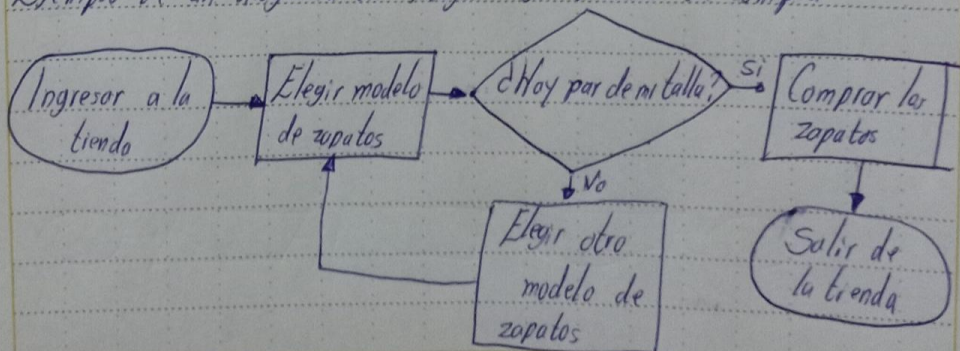
- 1) Debe tener Inicio y Fin.
- 2) Las líneas utilizadas para indicar la dirección deben ser correctas.
- 3) Todas las líneas deben estar conectadas.
- 4) Debe construirse de arriba hacia abajo y de izquierda a derecha.
- 5) La notación debe ser independiente del lenguaje de programación.
- 6) Al realizar tareas complejas es recomendable poner comentarios.
- 7) Si se requiere más de una hoja se usaron los conectores requeridos y se enumerarán las páginas.
- 8) No puede llegar más de una línea a un símbolo determinado.

Questions

¿Puedo combinar
dos diagramas de
Flujos?

¿Que tan detallado
puede ser su
construcción?

Ejemplo de un diagrama de Flujo: "Un proceso de compra"



Summary:

Las reglas para construir de forma correcta un diagrama de Flujo están para que todos los que lo lean lo pueden leer y entender con claridad para facilitar el proceso de codificación. Se presento un ejemplo donde el problema era comprar unos nuevos zapatos.

NAME

Isaac Felix

PAGES

4/7

SPEAKER/CLASS

Programación

DATE - TIME

23/01/2025

Title: Algoritmo, diagrama de Flujo y Programación en C

Keyword

Datos
 Numérico
 Carácter
 Símbolos
 Simple
 Estructurado

Topic: Tipos de datos

Notes: Los datos se clasifican en simples y estructurados; los datos simples solo ocupan una casilla de memoria. Dentro de este grupo de datos se encuentran principalmente los enteros, los reales y los caracteres.

Sus tipos en lengua C son:

int	- Enteros	-	-32,768 a +32,767
float	- Reales	-	3.4×10^{-38} a 3.4×10^{38}
long	- Enteros de largo alcance	-	-2 ¹⁴⁷ , 483, 648 a 2 ¹⁴⁷ , 483, 648
double	- Reales de doble precisión	-	1.7×10^{-308} a 1.7×10^{308}
char	- Carácter	-	Símbolos del abecedario, símbolos especiales, etc.

Questions

¿Un dato simple puede tener propiedades de estructurado?

¿Puedo combinarlos?

Un dato estructurado tiene varios componentes. Los arreglos, cadena de caracteres y registros son algunos de los más conocidos e utilizados.

Summary:

En el lenguaje de programación C se utilizan los datos para que un programa funcione, estos datos se dividen en simples y estructurados cada uno contando con su propio propósito y objetivo.

NAME
Isaac Féliz

PAGES
5/7

SPEAKER/CLASS
Programación

DATE - TIME
23/01/2025

Title: Algoritmo, diagrama de Flujo y Programación en C

Keyword

Fijo
Variable
Identificador
Valor
Programa
Declarar

Topic: Constantes y Variables

Notes: Las constantes son datos que no cambian durante la ejecución de un programa. Para nombrar las constantes utilizamos identificadores. Las constantes se deben definir antes de comenzar el programa principal, y éstos no cambiarán su valor durante la ejecución del mismo.

Ej: `const int nu1 = 20; /* es una constante tipo entero */`
`#define nu1 20;`

Las variables son objetos que pueden cambiar su valor durante la ejecución de un programa. También se utilizan identificadores y se deben declarar en el programa principal y en las funciones.

Ej: `void main(void)`
`{`

`...`
`int va1, va2; /* Declaración variables tipo entero */`
`...`

Questions

¿Se pueden operar variables con constantes?

¿Cuántos valores pueden llegar a tener?

Summary: Las constantes y las variables se utilizan siempre a la hora de codificar para almacenar los valores de los datos y poder operar con ellos a voluntad. Las constantes siempre mantienen su valor y las variables cambian su valor cuando pasan por un bloque de asignación.

NAME

Isaac Felix

PAGES

6/7

SPEAKER/CLASS

Programación

DATE - TIME

23/01/2025

Title: Algoritmo, diagrama de Flujo y Programación en C

Keyword

Aritmética

Resultado

Operaciones

Suma

Resta

Multiplicación

División

Topic: Operadores aritméticos

Notes: Los operadores aritméticos nos permiten realizar operaciones entre operandos, números, constantes o variables. El resultado de una operación aritmética siempre es un número.

Estos son los operadores en C:

(+ Suma), (- Resta), (* Multiplicación), (/ División), (% Módulo).

Questions

¿Hasta qué número pueden llegar las operaciones?

¿Cuántas operaciones se pueden realizar?

Se respetan la misma jerarquía de los operadores y se aplican de izquierda a derecha: (1° [*, /, %]) (2° [+ , -]).

El lenguaje C presenta la forma de simplificar el uso de los operadores aritméticos: (+ \Rightarrow +=) (- \Rightarrow -=) (* \Rightarrow *=) (/ \Rightarrow /=) (% \Rightarrow %=)

Ej: += $x=25;$

$y=5;$

$x+=5 \Rightarrow x = x+5 = x = 30$

Summary:

Los operadores aritméticos nos permiten poder manejar los datos tipo numérico y poder llegar a los resultados que buscamos. Es posible utilizar constantes y variables por igual, además en el caso del lenguaje C es posible simplificar algunas acciones.

Title: Algoritmo, diagramas de Flujo y Programación en C

Keyword

Lógica

Falso

Verdadero

Negación

Conjunción

Disyunción

Topic: Operadores Lógicos

Notes: Los operadores lógicos permiten formular condiciones complejas a partir de condiciones simples, son de conjunción (&&), disyunción (||) y negación (!)

! Negación

Ej: $x = (1(7 > 15)); \quad x = 1$

&& Conjunción

Ej: $x = (35 > 20) \&\& (20 \leq 23); \quad x = 1$

ambos respuestas son iguales

Questions

¿Puedo combinar distintos operadores lógicos?

|| Disyunción

$x = (35 > 20) || (20 \leq 18); \quad x = 1$

las respuestas son distintas

¿En que situación concretas los puedo utilizar?

Estos operadores solo pueden lanzar resultados de 0 y 1 si se aplican sus condiciones antes vistas.

El 0 significa falso y el 1 significa verdadero.

Summary:

Los operadores lógicos, además de ser usados en Electrónica Digital, son utilizados en todos los tipos de programación estas respuestas simples de 0 y 1 ayudan a la toma de decisiones de los programas y son uno de los operadores más importantes.