

Introducción a los algoritmos

Informática I
Centro Regional Universitario Córdoba
UNDEF

6 de abril de 2021

¿Qué son los algoritmos?

Según la RAE

Conjunto ordenado y finito de operaciones que permite hallar la solución a un problema.

Según Deitel

Es el procedimiento para resolver un problema en términos de las acciones a ejecutar y el orden en el cual se llevan a cabo dichas acciones.

¿Cómo está compuesto un algoritmo?

- Entrada
- Procesamiento
- Salida

Preparar una taza de té

- Entrada: agua, saquito de té, taza
- Salida: té caliente
- Procedimiento:

Inicio del algoritmo

- Poner el agua fría dentro de la pava

Preparar una taza de té

- Entrada: agua, saquito de té, taza
- Salida: té caliente
- Procedimiento:

Inicio del algoritmo

- Poner el agua fría dentro de la pava
- Encender el fuego

Preparar una taza de té

- Entrada: agua, saquito de té, taza
- Salida: té caliente
- Procedimiento:

Inicio del algoritmo

- Poner el agua fría dentro de la pava
- Encender el fuego
- Poner la pava sobre el fuego

Preparar una taza de té

- Entrada: agua, saquito de té, taza
- Salida: té caliente
- Procedimiento:

Inicio del algoritmo

- Poner el agua fría dentro de la pava
- Encender el fuego
- Poner la pava sobre el fuego
- Esperar que el agua se caliente

Preparar una taza de té

- Entrada: agua, saquito de té, taza
- Salida: té caliente
- Procedimiento:

Inicio del algoritmo

- Poner el agua fría dentro de la pava
- Encender el fuego
- Poner la pava sobre el fuego
- Esperar que el agua se caliente
- Poner el saquito de té en la taza

Preparar una taza de té

- Entrada: agua, saquito de té, taza
- Salida: té caliente
- Procedimiento:

Inicio del algoritmo

- Poner el agua fría dentro de la pava
- Encender el fuego
- Poner la pava sobre el fuego
- Esperar que el agua se caliente
- Poner el saquito de té en la taza
- Agregar el agua caliente

Preparar una taza de té

- Entrada: agua, saquito de té, taza
- Salida: té caliente
- Procedimiento:

Inicio del algoritmo

- Poner el agua fría dentro de la pava
- Encender el fuego
- Poner la pava sobre el fuego
- Esperar que el agua se caliente
- Poner el saquito de té en la taza
- Agregar el agua caliente
- Apagar el fuego

Preparar una taza de té

- Entrada: agua, saquito de té, taza
- Salida: té caliente
- Procedimiento:

Inicio del algoritmo

- Poner el agua fría dentro de la pava
- Encender el fuego
- Poner la pava sobre el fuego
- Esperar que el agua se caliente
- Poner el saquito de té en la taza
- Agregar el agua caliente
- Apagar el fuego
- Esperar un minuto

Preparar una taza de té

- Entrada: agua, saquito de té, taza
- Salida: té caliente
- Procedimiento:

Inicio del algoritmo

- Poner el agua fría dentro de la pava
- Encender el fuego
- Poner la pava sobre el fuego
- Esperar que el agua se caliente
- Poner el saquito de té en la taza
- Agregar el agua caliente
- Apagar el fuego
- Esperar un minuto

Fin del algoritmo

Calcular la distancia recorrida por un móvil que se movió en MRUV

$$x(t) = V_0 t + \frac{1}{2} a t^2 \quad (1)$$

- Entrada: velocidad inicial, tiempo recorrido, aceleración
- Salida: distancia recorrida
- Procedimiento:

Inicio del algoritmo

- Ingresar la velocidad inicial del movil

Calcular la distancia recorrida por un móvil que se movió en MRUV

$$x(t) = V_0 t + \frac{1}{2} a t^2 \quad (1)$$

- Entrada: velocidad inicial, tiempo recorrido, aceleración
- Salida: distancia recorrida
- Procedimiento:

Inicio del algoritmo

- Ingresar la velocidad inicial del movil
- Ingresar el tiempo de desplazamiento

Calcular la distancia recorrida por un móvil que se movió en MRUV

$$x(t) = V_0 t + \frac{1}{2} a t^2 \quad (1)$$

- Entrada: velocidad inicial, tiempo recorrido, aceleración
- Salida: distancia recorrida
- Procedimiento:

Inicio del algoritmo

- Ingresar la velocidad inicial del movil
- Ingresar el tiempo de desplazamiento
- Ingresar la aceleración

Calcular la distancia recorrida por un móvil que se movió en MRUV

$$x(t) = V_0 t + \frac{1}{2} a t^2 \quad (1)$$

- Entrada: velocidad inicial, tiempo recorrido, aceleración
- Salida: distancia recorrida
- Procedimiento:

Inicio del algoritmo

- Ingresar la velocidad inicial del movil
- Ingresar el tiempo de desplazamiento
- Ingresar la aceleración
- Aplicar la fórmula 1

Calcular la distancia recorrida por un móvil que se movió en MRUV

$$x(t) = V_0 t + \frac{1}{2} a t^2 \quad (1)$$

- Entrada: velocidad inicial, tiempo recorrido, aceleración
- Salida: distancia recorrida
- Procedimiento:

Inicio del algoritmo

- Ingresar la velocidad inicial del movil
- Ingresar el tiempo de desplazamiento
- Ingresar la aceleración
- Aplicar la fórmula 1
- Informar la distancia

Calcular la distancia recorrida por un móvil que se movió en MRUV

$$x(t) = V_0 t + \frac{1}{2} a t^2 \quad (1)$$

- Entrada: velocidad inicial, tiempo recorrido, aceleración
- Salida: distancia recorrida
- Procedimiento:

Inicio del algoritmo

- Ingresar la velocidad inicial del movil
- Ingresar el tiempo de desplazamiento
- Ingresar la aceleración
- Aplicar la fórmula 1
- Informar la distancia

Fin del algoritmo

Definición

El pseudocódigo es un lenguaje artificial e informal que ayuda a los programadores a desarrollar y probar algoritmos.

Ejemplo para calcular el volumen de una esfera

Inicio del algoritmo

- Imprimir: Ingrese el radio de la esfera

Definición

El pseudocódigo es un lenguaje artificial e informal que ayuda a los programadores a desarrollar y probar algoritmos.

Ejemplo para calcular el volumen de una esfera

Inicio del algoritmo

- Imprimir: Ingrese el radio de la esfera
- Leer: radio

Definición

El pseudocódigo es un lenguaje artificial e informal que ayuda a los programadores a desarrollar y probar algoritmos.

Ejemplo para calcular el volumen de una esfera

Inicio del algoritmo

- Imprimir: Ingrese el radio de la esfera
- Leer: radio
- $V_{esfera} = \frac{4}{3}\pi r^3$

Definición

El pseudocódigo es un lenguaje artificial e informal que ayuda a los programadores a desarrollar y probar algoritmos.

Ejemplo para calcular el volumen de una esfera

Inicio del algoritmo

- Imprimir: Ingrese el radio de la esfera
- Leer: radio
- $V_{esfera} = \frac{4}{3}\pi r^3$
- Imprimir: "El volumen de la esfera es V_{esfera} "

Definición

El pseudocódigo es un lenguaje artificial e informal que ayuda a los programadores a desarrollar y probar algoritmos.

Ejemplo para calcular el volumen de una esfera

Inicio del algoritmo

- Imprimir: Ingrese el radio de la esfera
- Leer: radio
- $V_{esfera} = \frac{4}{3}\pi r^3$
- Imprimir: "El volumen de la esfera es V_{esfera} "

Fin del algoritmo

Diagrama de flujo

- Es la representación gráfica de un algoritmo

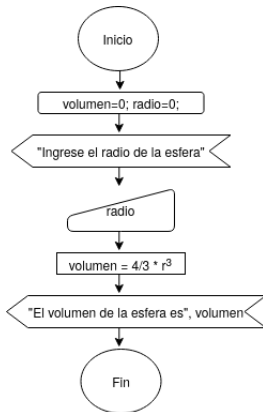


Figura: Diagrama de flujo para el cálculo del volumen de una esfera

Diagrama de flujo

- Es la representación gráfica de un algoritmo
- Se dibujan mediante símbolos de propósito especial: rectángulos, rombos, óvalos y círculos entre otros

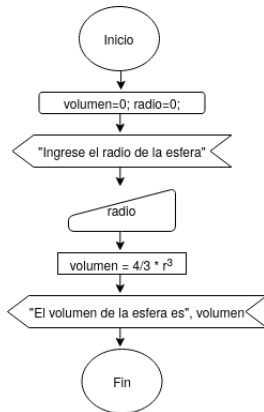


Figura: Diagrama de flujo para el cálculo del volumen de una esfera

Diagrama de flujo

- Es la representación gráfica de un algoritmo
- Se dibujan mediante símbolos de propósito especial: rectángulos, rombos, óvalos y círculos entre otros
- Estos símbolos se conectan mediante flechas llamadas **líneas de flujo**

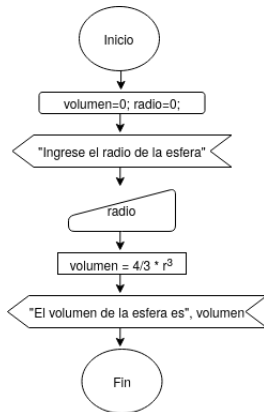


Figura: Diagrama de flujo para el cálculo del volumen de una esfera

Diagrama de flujo

- Es la representación gráfica de un algoritmo
- Se dibujan mediante símbolos de propósito especial: rectángulos, rombos, óvalos y círculos entre otros
- Estos símbolos se conectan mediante flechas llamadas **líneas de flujo**
- **Tienen un solo punto de inicio y fin.**

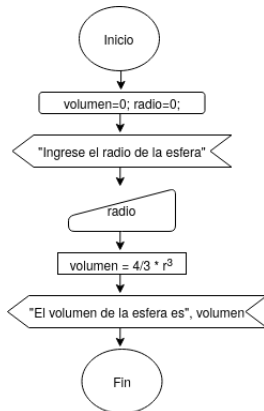


Figura: Diagrama de flujo para el cálculo del volumen de una esfera

Es una herramienta útil para entender que hace un determinado algoritmo o programa ya que esta es una ejecución a mano de nuestro algoritmo llevando el registro de los valores que cada variable tendrán.

Ejemplo para el cálculo del desplazamiento de un móvil visto anteriormente:

Ingreso velocidad inicial: $v_o = 10[m/s]$

Ingreso tiempo: $t = 30[s]$

Ingreso de la aceleración: $a = 10[m/s^2]$

Cálculo de: $x(t) = 10[m/s] * 30[s] + \frac{1}{2}10[m/s^2] * 30^2[m/s]^2$

- ¿Qué es un algoritmo?

- ¿Qué es un algoritmo?
- ¿Cómo está compuesto?

- ¿Qué es un algoritmo?
- ¿Cómo está compuesto?
- ¿Qué formas hay de representarlo?

- ¿Qué es un algoritmo?
- ¿Cómo está compuesto?
- ¿Qué formas hay de representarlo?

- ¿Qué es un algoritmo?
- ¿Cómo está compuesto?
- ¿Qué formas hay de representarlo?
 - Diagrama de flujo

- ¿Qué es un algoritmo?
- ¿Cómo está compuesto?
- ¿Qué formas hay de representarlo?
 - Diagrama de flujo
 - Pseudocódigo

- ¿Qué es un algoritmo?
- ¿Cómo está compuesto?
- ¿Qué formas hay de representarlo?
 - Diagrama de flujo
 - Pseudocódigo
- ¿Cómo probamos un algoritmo?

Tipos de datos en C

El lenguaje de programación C es **fuertemente tipado**, es decir que cada vez que se necesite declarar u operar con una variable, se debe definir y tener presente el tipo de la misma.

Nota: ver los tipos de datos *Unsigned*

Tipo de dato	Descripción	Rango
short	Valor entero de 2 bytes	-2^{16} a $2^{16} - 1$
int	Valor entero de 4 bytes	-2^{32} a $2^{32} - 1$
long	Valor entero de 8 bytes	-2^{64} a $2^{64} - 1$
char	Caracteres ASCII	-128 a 127
float	Valor decimal de 4 bytes	3.4×10^{-38} a 3.4×10^{-38}
double	Valor decimal de 8 bytes	1.7×10^{-308} a 1.7×10^{-308}
bool	Valor binario	True o False
void	Tipo de dato nulo	
string	Cadena de char	

Cuadro: Tipos de datos en C

Entrada y salida de datos I

Para imprimir por pantalla o ingresar datos a un programa en C, se debe informar **expresamente** el tipo de dato que se espera imprimir y/o recibir. Esto se realiza mediante el uso de **especificadores de formato**.

Tipo de dato	Especificador de formato
short	<i>%hd</i>
int	<i>%d</i>
long	<i>%li</i>
char	<i>%c</i>
float	<i>%f</i>
double	<i>%lf</i>

Cuadro: Especificadores de formato en C

Scanf

Es una función de la librería de entrada/salida de C que permite tomar información desde el teclado.

Sintaxis: `scanf(.especificador de formato", &variable);`

Ejemplo : `scanf(" %d", &edad);`

Printf

Es una función de la librería de entrada/salida de C que permite imprimir información por pantalla.

Sintaxis: `printf(.especificador de formato", variable);`

Ejemplo: `printf("Su edad es: " %d , edad);`

Nota: para estos ejemplos se supone que la variable se ha declarado de tipo int. Ver ejemplos siguientes.

Entrada y salida de datos III

```
1  #include <stdio.h>
2  int main()
3  {
4      int edad=0;
5      float peso=0;
6      printf("Ingrese su edad \n");
7      scanf(" %d",&edad);
8      printf("Ingrese su peso\n");
9      scanf(" %f",&peso);
10     printf("Su edad es %d y su peso %f\n",edad,peso);
11 }
```

Operadores en C I

Operadores de asignación:

Operador	Acción	Ejemplo	Resultado
=	Asignación básica	$x = 10$	x vale 10
*=	Asignación producto	$x * = 10$	x vale 100
/=	Asignación división	$x / = 2$	x vale 50
+=	Asignación suma	$x + = 5$	x vale 55
-=	Asignación resta	$x - = 7$	x vale 48

Cuadro: Operadores de asignación

Operadores aritméticos:

Operador	Acción	Ejemplo	Resultado
-	Resta	$x = 12 - 3$	x vale 9
+	Suma	$x = 12 + 3$	x vale 15
*	Multiplicación	$x = 12 * 3$	x vale 36
/	División	$x = 12 / 3$	x vale 4
-	Decremento	$x = 12; x --$	x vale 11
++	Incremento	$x = 12; x ++$	x vale 13
%	Modulo	$x = 13 \% 2$	x vale 1

Cuadro: Operadores de asignación

Precedencia de operadores

El lenguaje C evalúa las expresiones aritméticas en una secuencia precisa, por lo general son las mismas que aplicaríamos en el álgebra:

- 1 Las operaciones de multiplicación, división y módulo se resuelven primero. Si en una misma operación aparecen varias de ellas, se resuelven de izquierda a derecha

Precedencia de operadores

El lenguaje C evalúa las expresiones aritméticas en una secuencia precisa, por lo general son las mismas que aplicaríamos en el álgebra:

- 1 Las operaciones de multiplicación, división y módulo se resuelven primero. Si en una misma operación aparecen varias de ellas, se resuelven de izquierda a derecha
- 2 Las operaciones de suma y resta se aplican después. Si hubiese varias de estas, C separará en términos al igual que se haría en el álgebra

Precedencia de operadores

El lenguaje C evalúa las expresiones aritméticas en una secuencia precisa, por lo general son las mismas que aplicaríamos en el álgebra:

- 1 Las operaciones de multiplicación, división y módulo se resuelven primero. Si en una misma operación aparecen varias de ellas, se resuelven de izquierda a derecha
- 2 Las operaciones de suma y resta se aplican después. Si hubiese varias de estas, C separará en términos al igual que se haría en el álgebra
- 3 Luego de resueltas todas las operaciones, se procede a la asignación

Precedencia de operadores: ejemplo

Ecuación de una recta en forma algebraica

$$y(x) = ax + b \quad (2)$$

Ecuación de una recta en C

$$y = a * x + b \quad (3)$$

Precedencia de operadores

- 1 Operación $a * x$

Precedencia de operadores: ejemplo

Ecuación de una recta en forma algebraica

$$y(x) = ax + b \quad (2)$$

Ecuación de una recta en C

$$y = a * x + b \quad (3)$$

Precedencia de operadores

- 1 Operación $a * x$
- 2 Operación $+b$

Precedencia de operadores: ejemplo

Ecuación de una recta en forma algebraica

$$y(x) = ax + b \quad (2)$$

Ecuación de una recta en C

$$y = a * x + b \quad (3)$$

Precedencia de operadores

- 1 Operación $a * x$
- 2 Operación $+b$
- 3 Asignación del resultado a la variable y

Precedencia de operadores: ejemplo en C

```
1  #include <stdio.h>
2  int main()
3  {
4      float m=0; /*Pendiente*/
5      float y=0; /*Variable dependiente*/
6      float x=0; /*Variable independiente*/
7      float b=0; /*Ordenada al origen*/
8      printf("Ingrese el valor de la pendiente \n");
9      scanf("%f",&m);
10     printf("Ingrese el valor de la ordenada al origen \n");
11     scanf("%f",&x);
12     printf("Ingrese el valor de la variable indep \n");
13     scanf("%f",&b);
14     y=m*x+b; /*Operacion*/
15     printf("el valor de y(%f)=%f \n",x,y);
16
17 }
```