

Algoritmos en Diagramas de flujo y Pseudocódigo

IF1300-Introducción a la computación e informática

Propiedades de los algoritmos

- El científico Donald Knuth ofreció cinco requisitos para un algoritmo:
 - **Carácter finito:** Un algoritmo siempre debe terminar después de un número finito de pasos.
 - **Precisión:** Cada paso de un algoritmo debe estar precisamente definido; las operaciones a llevar a cabo deben ser especificadas de manera rigurosa y no ambigua para cada caso.

Propiedades de los algoritmos

- **Entrada:** Un algoritmo tiene cero o más entradas: cantidades que le son dadas antes de que el algoritmo comience, o dinámicamente mientras el algoritmo corre.
- **Salida:** Un algoritmo tiene una o más salidas: cantidades que tienen una relación específica con las entradas.
- **Eficacia:** También se espera que un algoritmo sea eficaz.

Problema - Enunciado

- $2 + 2 = 4$
- $1234 + 2000 = 3234$
- $0 + 0 = 0$
- $10 + 20 = 30$

Redactar el enunciado, cual seria un enunciado:

Se requiere, realizar un algoritmo que reciba dos números enteros realice la operación de suma entre ellos el resultado de la suma.

Reglas y Estructura

- Se debe poner un nombre **significativo** al algoritmo.
 - Inicia con **Mayúscula**
- Se debe poner un nombre **significativo** a las variables utilizadas.
 - Inician con **minúscula**

Que son las Variable?

Primer acercamiento:

- Es un **nombre** que representa el valor de un dato.
- Es un espacio en **memoria** en la computadora donde se almacena información.

- Se presentan ejemplos concretos sobre el significado de una variable:

Tipos de datos

- **Entero:** Subconjunto finito de los números enteros, cuyo rango o tamaño dependerá del lenguaje en el que posteriormente se codifique el algoritmo y de la computadora.
- **Real:** Subconjunto de los números reales limitado no sólo en cuanto al tamaño, sino también en cuanto a la precisión.
- **Lógico:** Conjunto formado por los valores Verdadero y Falso.

Tipos de datos

- **Carácter:** Conjunto finito y ordenado de los caracteres que la computadora reconoce.
- **Cadena:** Los datos (objetos) de este tipo, contendrán una serie finita de caracteres, que podrán ser directamente traídos o enviados a/desde consola.
- **NOTA:** PseInt, permite tipos de datos numéricos, lógicos y carácter entre otros.

CONSTANTE

- Es un dato cuyo valor **no cambia** durante todo el desarrollo del algoritmo.
- Se debe **inicializar** al inicio del programa al momento de ser **creada**.

Reglas para nombrar variables y constantes

- Los nombres deben ser significativos, sugiriendo lo que representan.
- No pueden haber 2 variables con el mismo nombre.
- No podrá coincidir con palabras reservadas, propias del lenguaje algorítmico.
- Se admitirá un máximo de 32 caracteres.
- Comenzará siempre por un carácter alfabético y los siguientes podrán ser letras, dígitos o el símbolo subrayado.
- Las constantes se nombrarán totalmente con mayúscula para diferenciarlas de las variables.

Ejemplos de nombres de variables y CONSTANTES

Variables:

Definir primerOperando **Como Entero**;

Definir peso, estatura, edad **Como Real**;

Definir esCasada, tieneBeca **Como Logico**;

Definir genero **Como Caracter**;

Definir nombre **Como Cadena**;

CONSTANTES:

Definir SALARIO_FIJO **Como Entero**;

SALARIO_FIJO = 500000;

Definir PI **Como Real**;

PI = 3,14

...

Algunos usos comunes de variables

- Variable utilizada como **Contador**: Variable entera cuyo valor se incrementa o se decrementa en un valor **constante** cada vez que se produce una determinada acción. Por ejemplo: contar de uno en uno los alumnos de la clase.
- Ejemplo:
 - `totalAlumnos = 0;`
 - `totalAlumnos = totalAlumnos + 1;`
 - El valor de la variable `totalAlumnos` es ??

Algunos usos comunes de variables

- Variable utilizada como **Acumulador**: Variable entera o real cuyo valor va almacenando una cantidad **variable** (el valor de la cantidad cambia constantemente).
- Por ejemplo: guardar en una alcancía los **vueltos**:
 - `dineroEnAlcancia = 0;`
 - `Vuelto = ?;`
 - `dineroEnAlcancia = dineroEnAlcancia + vuelto`

Algunos usos comunes de variables

- Variable utilizada como **Bandera, interruptor o switch**: Variable lógica que sólo puede tomar dos valores, **verdadero o falso** (1 ó 0).
- Ejemplo:
 - tieneBeca = verdadero;

Nota: por lo general es utilizada durante las operaciones condicionales.

Video Galaga

• Proceso de Asignación

- **Asignación:** Para asignar un valor a una variable se utiliza el símbolo “=” que en este contexto significa “es reemplazado por”.
- De manera general se tiene:
- `resultadoSuma = primerOperando + segundoOperando;`
- El valor de **resultadoSuma** se sustituye por el valor de expresión:
 - `(primerOperando + segundoOperando)`

Instrucciones de Pseudocódigo

- Ejemplos de asignación:
 - edad = 50;
 - provincia = "Alajuela";
 - suma = operando1 + operando2;
 - contador = contador + 2;
 - peso = 5,35;
 - resultadoFormula = numero1 / 2 * 3;
 - tieneBeca = verdadero;
 - genero = 'F';

Expresión

- Una expresión es una combinación de operadores y operandos.
- Los **operandos** podrán ser: constantes, variables, valores y otras expresiones.
- Los **operadores** podrán ser: de cadena, aritméticos, relacionales o lógicos.

Expresión numérica

- Los operandos que intervienen en ellas son numéricos, el resultado es también de tipo numérico y se construyen mediante los operadores **aritméticos**.
- Se pueden considerar análogas a las fórmulas matemáticas.

Expresión numérica

- Operadores aritméticos

Significado	Operador Aritmético	Ejemplo
Suma	+	$12.5 + 10.2 = 22.7$
Resta	-	$45 - 10 = 35$
Multiplicación	*	$20 * 4 = 80$
División	/	$27 / 5 = 5$
Potenciación	^	$2 ^ 5 = 32$
Resto de la división entera (residuo o módulo)	% o MOD	$27 \% 5 = 2$

Expresión lógica o booleana

- Su resultado será **verdadero o falso únicamente**. Se construyen mediante los operadores relacionales y lógicos.

Expresión lógica o booleana

- Operadores relacionales

Operador	Operador Relacional	Ejemplo	Resultado
Mayor que	>	$25 > 5$	Verdadero
Menor que	<	$25 < 25$	Falso
Igual a	==	$25 \text{ == } 4$	Falso
Menor o igual que	$\text{<}=\text{}$	$50 \text{ <}= 100$	Verdadero
Mayor o igual que	$\text{>}=\text{}$	$500 \text{ >}= 1$	Verdadero
Distinto	$\text{!}=\text{ ó } \text{<}>\text{}$	$1 \text{ !}=\text{ } 6$	Verdadero

Expresión lógica o booleana

- Operadores lógicos

Significado	Operador Lógico
Multiplicación lógica	y (\wedge)
Suma lógica	o (\vee)
Negación lógica	no (\neg)

Expresión lógica o booleana

- Tabla de verdad para el operador lógico Y
 - El resultado del Y lógico entre dos operandos es verdadero solamente cuando ambos valores son verdaderos.

a	b	$a \wedge b$
falso	falso	falso
falso	verdadero	falso
verdadero	falso	falso
verdadero	verdadero	verdadero

Expresión lógica o booleana

- Tabla de verdad para el operador lógico O
 - El resultado del O lógico entre dos operandos es verdadero siempre que alguno de los dos operandos es verdadero.

a	b	$a \vee b$
falso	falso	falso
falso	verdadero	verdadero
verdadero	falso	verdadero
verdadero	verdadero	verdadero

Expresión lógica o booleana

- Tabla de verdad para el operador lógico **NO**
 - La negación se aplica sobre un único operando booleano y permite obtener su valor opuesto.
 - La negación de un operando produce un valor verdadero solamente cuando el operando es falso y viceversa.

a	$\neg a$
falso	verdadero
verdadero	falso

Prioridad de operadores

Orden	Tipo	Operadores
1	Paréntesis	()
2	Multiplicativos	* / %
3	Aditivos	+ -
4	Relacionales	< > <= >=
5	Igualdad	== !=
6	NO lógico	¬
7	Y lógico / O lógico	^ ∨
8	Asignación	=

Ejemplos de fórmulas

Fórmula	Expresión
bc	$b*c$
$b^2 - 4ac$	$b ^ 2 - 4 * a * c$
$\frac{y2 - y1}{x2 - x1}$	$(y2 - y1) / (x2 - x1)$
$x + y + z$	$x + y + z$

La potencia lleva el mismo signo del Y lógico

Diagramas de flujo

- Herramienta utilizada para la representación visual y gráfica de algoritmos
- Compuestos por una serie de símbolos icónicos unidos por flechas.
- Ayudan a representar de manera visual soluciones de algoritmos.

Diagramas de flujo

- **Características de los diagramas de flujo**
 - Los símbolos representan acciones o funciones en el programa.
 - Las flechas representan el orden de realización de las acciones o funciones, marcando el sentido o flujo lógico del algoritmo.
 - Cada símbolo tendrá al menos una flecha que conduzca a él y una flecha que parta de él, a excepción de los terminadores y conectores.
 - Se leen de arriba a abajo y de izquierda a derecha.

Ejemplo de un problema

- Sumar dos números enteros
 - **Entradas:** recibe dos números enteros, vamos a llamarlos
 - a (primerOperando)
 - b (segundoOperando)
 - **Proceso:**
 - se suman el valor contenido en **primerOperando** y **segundoOperando**
 - el resultado se almacena en c (resultadoSuma).
 - **Salida:** la variable **resultadoSuma** devuelve el resultado de la suma entre el **primerOperando** y **segundoOperando**

Pseudocódigo

- Es un pseudolenguaje intermedio entre el natural del programador y el lenguaje de programación seleccionado, considerándose por tanto un lenguaje de pseudoprogramación.
- No existe una sintaxis estándar para el pseudocódigo, utiliza una mezcla de lenguaje natural (utilizando como base la lengua nativa del programador) y una serie de símbolos, términos y otras características propias de los lenguajes de programación de alto nivel

Estructura Pseudocódigo

- Debe tener un inicio de proceso y un fin de proceso
(note la **indentación = tabulación**)

-

nombre del algoritmo

creación de variables y constantes

inicio

 Instrucción1

 Instrucción2

 InstrucciónN

 ...

fin

Instrucciones de Pseudocódigo

- Ejemplo de algoritmo:

Proceso SumaSimple

 Definir primerOperando,segundoOperando, resultadoSuma
 Como Número;

 primerOperando = 13;

 SegundoOperando = 25;

 resultadoSuma = primerOperando + segundoOperando;

 Escribir "El resultado es: ",resultadoSuma;

FinProceso

- **NOTA:** Después de estas instrucciones **resultadoSuma** contendrá el valor 38

Instrucciones de Pseudocódigo

- **Leer:** pide al usuario un valor que será asignado a la variable. Se utiliza el siguiente formato:
 - **Leer** argumento 1, argumento 2, ... , argumento n
- **Ejemplo**

...

Leer primerOperando;

....

Instrucciones de Pseudocódigo

- **Escribir:** variables, resultados y mensajes que se muestran al usuario. Se utiliza el siguiente formato:
 - Escribir argumento1, argumento 2, ... , argumento n;
 - Escribir "mensaje ", argumento1;
- **Ejemplo:**

...

Escribir "El resultado es: ",resultadoSuma;

...

Instrucciones de Pseudocódigo

- Los argumentos del Escribir pueden ser cadenas de caracteres entrecomilladas o variables:
 - Si son cadenas de caracteres entrecomillados se imprime literalmente lo que está entre comillas
 - Si son variables se imprime el contenido de dicha variable (no el nombre)

Instrucciones de Pseudocódigo

- Ejemplo de diferentes argumentos:

...

sueldo = 1000;

trabajador = “Martínez”;

Escribir “el sueldo de ”, trabajador , “ es: ”,
sueldo , “ Euros.”;

...

La instrucción Escribir, muestra lo siguiente:
el sueldo de Martínez es: 1000 Euros.

Instrucciones de Pseudocódigo

- Ejemplo de un algoritmo que suma 2 números ingresados por el usuario:

Proceso Suma

Definir primerOperando,segundoOperando, resultadoSuma Como
Número;

Escribir "Ingrese el primer numero:";

Leer primerOperando;

Escribir "Ingrese el segundo numero:";

Leer segundoOperando;

resultadoSuma = primerOperando + segundoOperando;

Escribir "El resultado es: ",resultadoSuma;

FinProceso