

El diseño de un programa
Jhon Jairo Padilla Aguilar, PhD.

Partes básicas de un programa

- **Instrucciones:**

- Indican al computador qué hacer paso a paso
- Un computador procesa información
- Convierte entradas en salidas
- Hay instrucciones para:
 - Ingresar información del exterior al computador (p.ej. Leer del teclado)
 - Sacar información del computador hacia el exterior (p.ej. Imprimir en pantalla)
 - Transformar información (p.ej. $x=a+5$)
 - Tomar decisiones: (p.ej. Si $A==3$ entonces imprima A)

Partes básicas de un programa

- **Variables:**

- La información se almacena en la memoria del computador en forma de variables
- Una variable tiene un nombre simbólico:
 - `A=5`
 - `opcion="Y"`
- Una variable es una parte de la memoria que almacena algún tipo de información
- Hay diferentes tipos de información o variables

Ejemplos de contenidos de las variables

- Cadenas de caracteres:
 - “Hola mundo”
- Numéricas:
 - Enteros (Integer): 432
 - Reales (float): -12,425781
- Booleanas:
 - Sólo toman dos posibles valores: Falso o Verdadero

Pasos para diseñar un programa

- Primero debes entender qué tarea debe hacer el programa
- Debes describir los pasos que debe dar el computador para realizar la tarea propuesta
 - Esta descripción se hace mediante un Algoritmo
- Algoritmo:
 - Listado ordenado de pasos a seguir para realizar una tarea

Algoritmo

- Se puede describir en palabras o mediante dibujos
- Descripción en palabras: usa pseudocódigo
 - Un lenguaje muy similar a nuestro lenguaje natural
- Descripción en Dibujos: Diagrama de Flujo
 - Utiliza símbolos gráficos para indicar: ingreso/egreso de información, transformación, decisiones, etc.

Definición de Algoritmo

- Secuencia finita, ordenada, y sin ambigüedad, de pasos para resolver un problema computacional.



Correcto?

Eficiente?

Pseudocódigo

- Representación en forma de texto del algoritmo
- Describe la estructura básica del programa
- No obedece las reglas estándar de un lenguaje
- Es independiente del lenguaje de programación a utilizar.

Ejemplo descripción en Pseudocódigo

Decisión simple:

```
si <condición> entonces  
    <acciones1>  
fin_si
```

Decisión doble:

```
si <condición> entonces  
    <acciones1>  
si_no  
    <acciones2>  
fin_si
```

Repetitivas:

```
mientras <condición> hacer  
    <acciones>  
fin_mientras  
repetir  
    <acciones>  
hasta_que <condición>  
desde <variable>←<v_inicial> hasta <v_final>  
    [incremento | decremento <incremento>] hacer  
    <acciones>  
fin_desde
```

Ejemplo 1- pseudocódigo

- Programa que lee un valor en grados Celcius y lo convierte a Kelvin

```
inicio
  leer(celsius)
  kelvin ← celsius + 273.15
  escribir(Kelvin)
fin
```

Ejemplo 2- pseudocódigo

- Algoritmo que lee 4 datos y calcula el producto, la suma, la media aritmética y luego los imprime en pantalla

```
inicio
  leer (a,b,c,d)
  producto ← a * b * c * d
  suma ← a + b + c + d
  media ← (a + b + c + d) / 4
  escribir (producto, suma, media)
fin
```

Diagrama de flujo

- Es la representación gráfica de un algoritmo
- Se utiliza una serie de símbolos, los cuales se relacionan por medio de líneas
- Los símbolos están definidos de acuerdo a la norma ANSI (instituto norteamericano de normalización)

Símbolos del diagrama de flujo



Representa el inicio y fin de un programa, ya sea el principal
O algún sub programa o modulo

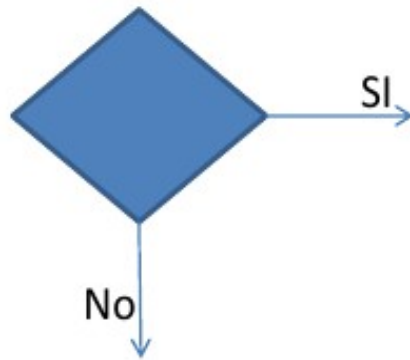


Permite el ingreso o salida de datos para la ejecución de una
Tarea sin importar el modo de entrada o salida



Este bloque representa la realización de cualquier tarea, sea una
Comparación, aritmética, ecuaciones entre otros.

Símbolos del diagrama de flujo



Bloque de selección. Evalua la estructura condicional IF
Obteniendo 2 resultados logicos posibles SI o NO

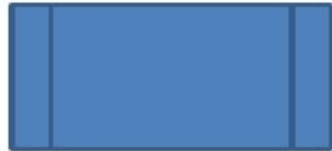


Conector que permite enlazar partes del diagrama
Sin utilizar lineas conectoras



Misma funcion del anterior, solamente que se utiliza
Para enlazar diferentes paginas

Símbolos del diagrama de flujo



Permite llamar una subrutina definida en otro Modulo del programa



Muestra los resultados en pantalla

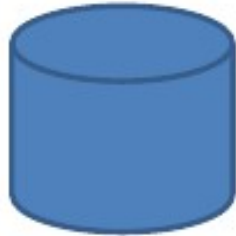


Muestra el resultado en una impresora



Representa el conteo de un ciclo repetitivo

Símbolos del diagrama de flujo



Almacenamiento en el disco duro

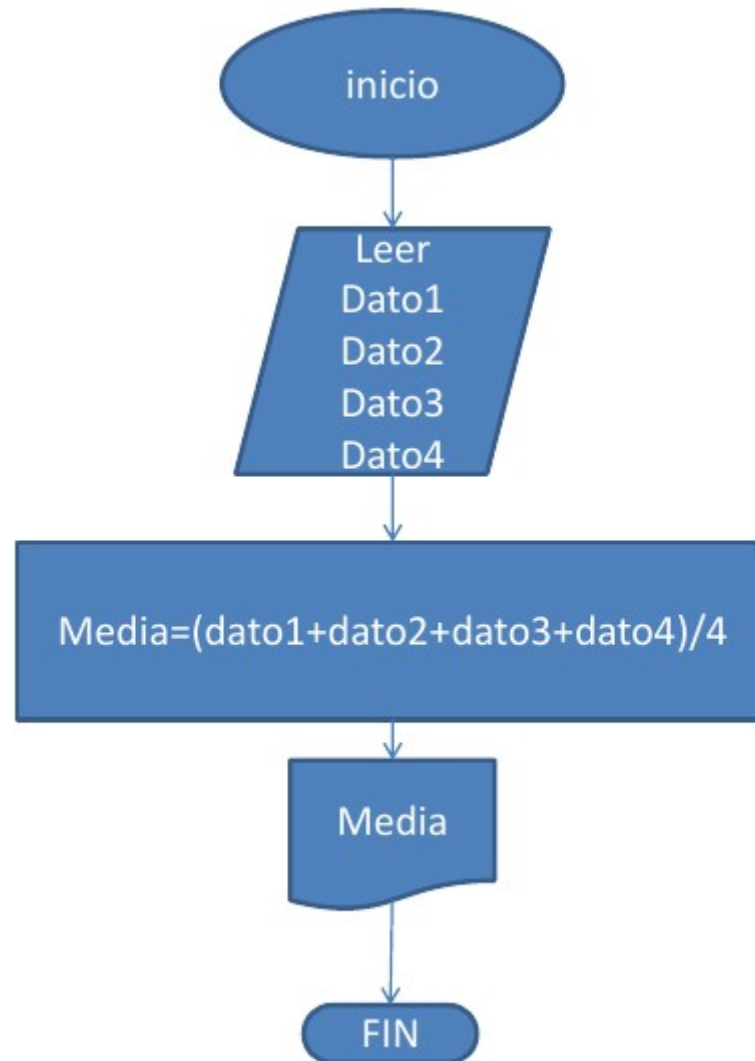


Entrada de datos desde archivo

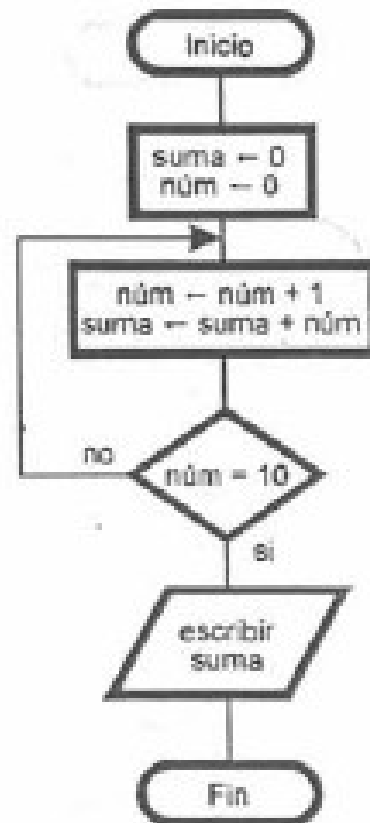


Entrada de datos desde el teclado

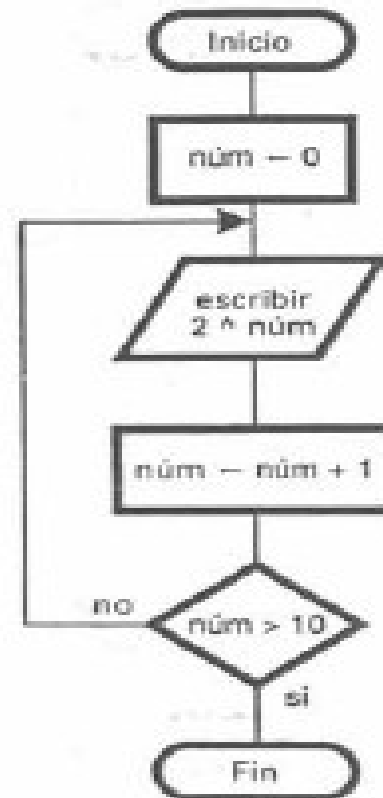
Ejemplo: Cálculo de la media aritmética



Ejemplo: calcular la suma de los números del 1 al 10



Ejemplo: Calcular las potencias de 2 con los exponentes del 0 al 10



Pasos para hacer un algoritmo

- Entender el problema a resolver
- Cuáles son los datos de entrada?
- Cuáles son los datos de salida?
- Qué pasos debo realizar para convertir las entradas en salidas?
- Descripción de los pasos en forma de algoritmo
- Prueba de escritorio para evaluar si es correcto el algoritmo

Prueba de escritorio

- Se busca probar diferentes situaciones:
 - Una típica para la cual fue diseñado el algoritmo
 - Situaciones atípicas que contemplan posibles errores por parte de los usuarios:
 - P.ej. Si el dato de entrada debía estar entre 0 y 100, y el usuario ingresa un valor por fuera de ese rango, debería considerarse el imprimir un mensaje indicando al usuario que lo intente de nuevo con un valor entre 0 y 100

Paso siguiente: Codificación

- Una vez diseñado el algoritmo y probado en escritorio, se pasa a convertir esos pasos en un conjunto de instrucciones del lenguaje particular a usar.
- En nuestro curso usaremos Python como lenguaje, por lo que debemos estudiar su sintaxis y posibilidades

Taller

1. Diseñe un diagrama de flujo que dado un numero entregue su tabla de multiplicar desde el multiplicador 1 hasta el 15.
2. Hacer un diagrama de flujo para calcular la hipotenusa de un triangulo rectángulo.
3. Hacer un programa que transforme la velocidad en km/h a m/s
4. Calcular el promedio ponderado acumulado de 5 materias que valen 1,4,7,5 y 3 créditos respectivamente.

Referencias

- Diapositivas “Introducción a la programación”. Jairo Viola.
- “Fundamentos de programación: Libro de Problemas”. Luis Joyanes Aguilar. Ed. McGraw-Hill.