



Unidad II Algoritmos y herramientas para el desarrollo de algoritmos

M.C. Juan Carlos Olivares Rojas



Agenda

- 2.1 Pseudocódigo
- 2.2 Diagramas de flujo

2.1 Pseudocódigo

- Es una manera de expresar un algoritmo en un lenguaje común, el cual puede ser español (como en nuestro caso), inglés, etc.
- La ventaja de utilizar pseudocódigo es que su implementación es casi directa en cualquier lenguaje de programación de alto nivel.

Algoritmo para la suma de un número

Entrada:

num1, num2 : números enteros

Salida:

suma: número entero con el resultado de sumar num1 y num2

Inicio

suma \leftarrow num1 + num2

Fin

Pseudocódigo del algoritmo

DECLARACIONES:

num1, num2, suma: enteros

INICIO

LEER num1 //Entradas

LEER num2

suma <- num1 + num2 //Procesos

IMPRIMIR suma //Salidas

FIN

Programa en Pascal para sumar dos números

```
program Suma;  
uses  
    Crt;  
var  
    num1, num2, suma: integer;  
begin  
    write('Introduce el número 1?'); {entradas}  
    read(num1);  
    write('Introduce el número 2?');  
    read(num2);  
    suma := num1 + num2; {procesos}  
    writeln('El resultado de la suma es: ', suma); {salidas}  
end
```

Programa en C para sumar dos números

```
#include <stdio.h>
int main(void)
{
    int num1, num2, suma;
    printf("Introduce el número 1?"); /*Entradas*/
    scanf("%d", num1);
    printf("Introduce el número 2?");
    scanf("%d", num2);
    suma = num1 + num2 /*Procesos*/
    printf("El resultado de la suma es %d", suma); /*Salidas*/
    return 0;
}
```

Programa en Java para sumar dos números

```
import java.io.*
public class Suma
{
    private int num1, num2, suma;
    public static void main(String args)
    {
        System.out.println("Introduce el número1?");
        num1 = System.in.readInt();
        System.out.println("Introduce el número2?");
        num2 = System.in.readInt();
        suma = num1 + num2;
        System.out.printl("El resultado de la suma es: " + suma);
    }
}
```

Programa en C# para sumar dos números

```
using System;
namespace suma
{
    public class suma
    {
        private int num1, num2, suma;
        public static void(string args[])
        {
            Console.write("Introduce el primer número?");
            num1 = Console.readInt();
            Console.write("Introduce el segundo número?")
            num2 = Console.readInt();
            suma = num1 + num2;
            Console.writeln("El resultado de la suma es: {0}", suma);
        }
    }
}
```

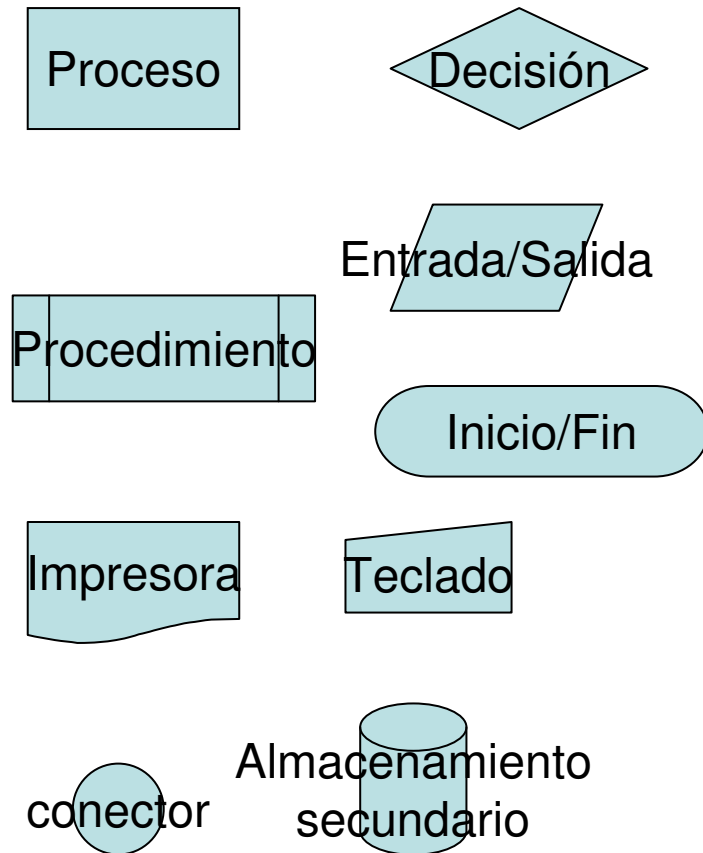
Algoritmos a desarrollar

- Encontrar el mayor de 3 números
- Determinar el número máximo de una serie
- Verificar si un entero positivo es primo
- Determinar el número primo mayor que un número
- Calcular el área de un rectángulo
- Calcular las raíces de una ecuación cuadrática.

2.2 Diagramas de flujo

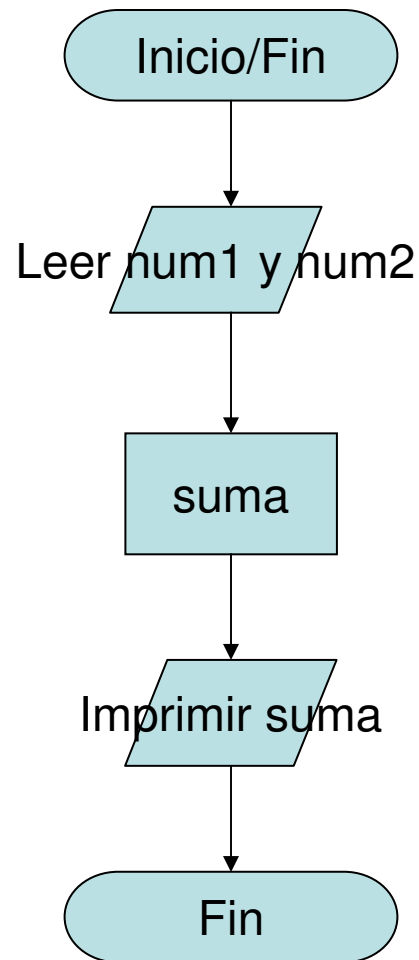
- Es una manera gráfica de especificar un algoritmo.
- Tiene como ventaja que se puede visualizar de mejor manera un algoritmo.
- La desventaja radica en que para algoritmos demasiado grandes se vuelve complejo hacer el seguimiento

Simbología



- Está es la nomenclatura más estandarizada. Existen elementos para actividades que casi no se utilizan, lectura de tarjetas perforadas, escritura en cintas

Diagrama de flujo para sumar dos números



Otros métodos para especificar algoritmos

- Investigar otros métodos para especificar algoritmos cómo:
 - Diagramas de Warnier
 - Diagramas NS
 - Tablas de decisión
 - Árboles de decisión

¿Preguntas?

¿?