

6. Estructuras de control

Programación en C.

Repaso

- ▶ Ya vimos cómo definir variables en C.

Pseudocódigo

```
num: Entero  
numreal: Real  
letra: Carácter  
Num: Entero  
NUM: Entero  
NuM: Entero
```



Lenguaje C

```
int num;  
float numreal;  
char letra;  
int Num, NUM, NuM;
```

La declaración de variables acaba en punto y coma

Distingue entre mayúsculas y minúsculas

Repaso

- ▶ También vimos cómo inicializar variables en C.

Pseudocódigo

```
num ← 3  
numreal ← 4.2  
letra ← 'a'
```



Lenguaje C

```
num = 3;  
numreal = 4.2;  
letra = 'a';
```

La inicialización de variables acaba en punto y coma

Repaso

- ▶ Los comentarios existen en ambos lenguajes.

Pseudocódigo

```
;Iniciación de variables  
num ← 3 ;Variable entera  
numreal ← 4.2  
letra ← 'a'
```



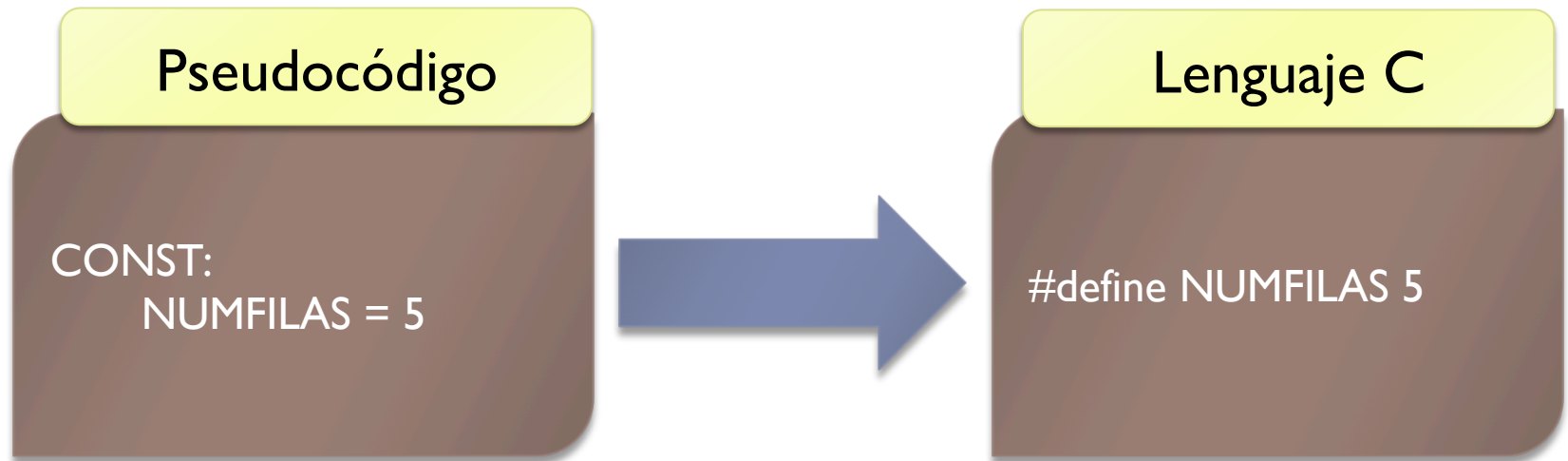
Lenguaje C

```
/* Inicialización */  
num = 3; //Variable entera  
numreal = 4.2;  
letra = 'a';
```

Hay dos maneras de especificar comentarios, entre `/* */` o bien empezando por `//`.

Repaso

- Definición de constantes.



Estructuras de control

- ▶ Cuando escribimos un algoritmo en pseudocódigo, usamos *estructuras de control*.
- ▶ Veamos cómo son en lenguaje C.



Estructuras de control

ELEMENTOS BÁSICOS DE UN PROGRAMA

Palabras reservadas

Identificadores

Caracteres especiales

Constantes

Expresiones

Variables

Instrucciones



ESTRUCTURAS DE CONTROL

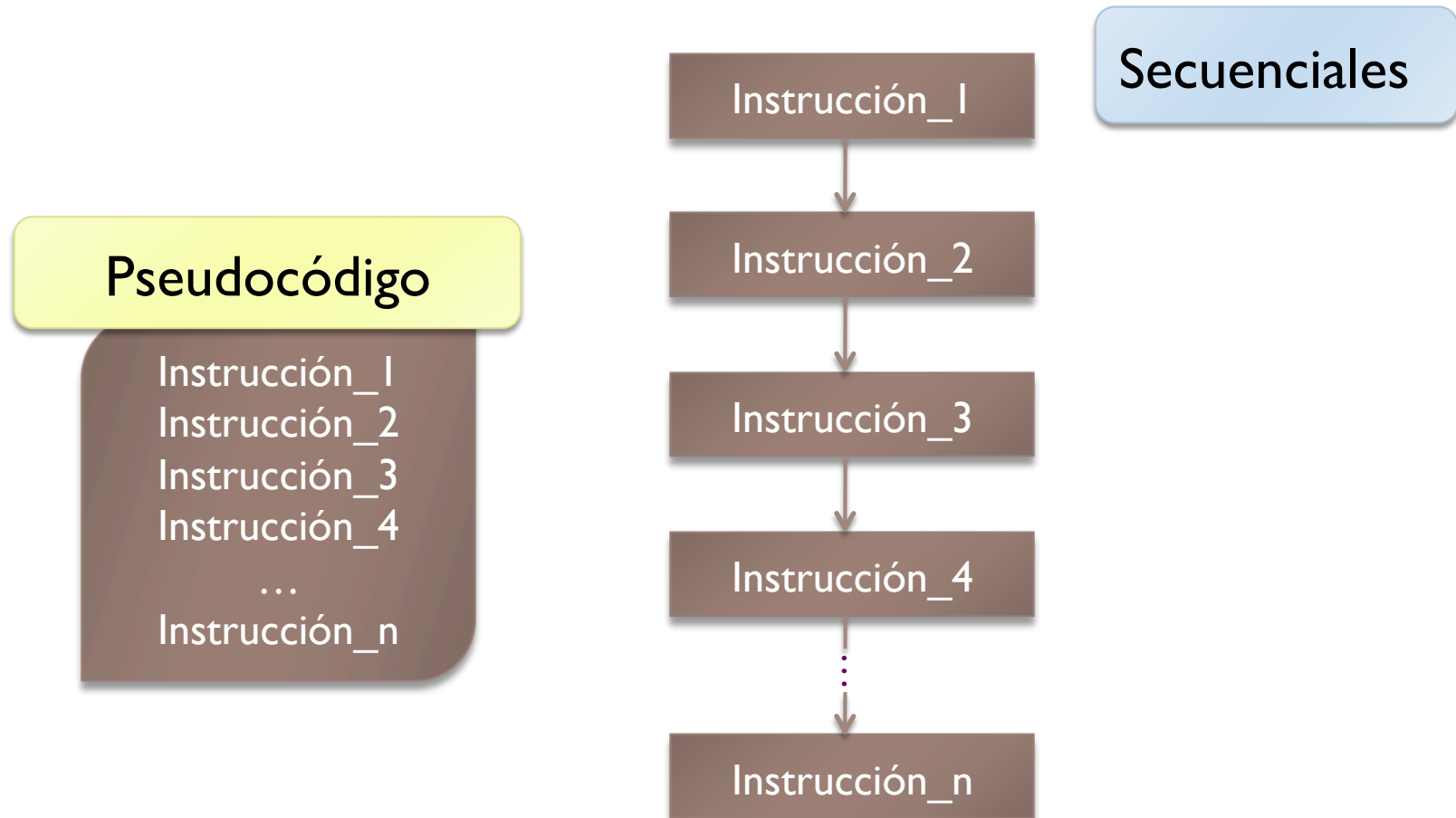
Secuenciales

Selectivas

Repetitivas

Estructuras de control secuenciales

- ▶ Las instrucciones se ejecutan una detrás de otra.



Estructuras de control secuenciales

- ▶ Las instrucciones se ejecutan una detrás de otra.

Secuenciales

Pseudocódigo

```
Instrucción_1  
Instrucción_2  
Instrucción_3  
Instrucción_4  
...  
Instrucción_n
```



Lenguaje C

```
Instrucción_1;  
Instrucción_2;  
Instrucción_3;  
Instrucción_4;  
...  
Instrucción_n;
```

Las instrucciones acaban en punto y coma

Estructuras de control secuenciales

► Ejemplo

ALGORITMO Imprime_num

ENTRADAS:

Num: ENTERO; Número que se lee

SALIDAS:

VARIABLES:

Num: ENTERO

INICIO

ESCRIBA "Escribe un número:"

LEA Num

ESCRIBA "El número introducido es:"

ESCRIBA Num

ESCRIBA "fin del algoritmo"

FIN

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int Num;
```

```
    printf("Escribe un número: ");
```

```
    scanf("%d", &Num);
```

```
    printf("El número introducido es %d\n",  
          Num);
```

```
    printf("Fin del algoritmo");
```

```
    return 0;
```

```
}
```

Estructuras de control secuenciales

► Ejercicio

EJERCICIO

- I. Realice un algoritmo en pseudocódigo que lea un valor en kilómetros y lo convierta a:
 - a) millas (1km son 0,62 millas; 1 milla son 1,60km).
 - b) pies (1 pie son 30,48cm).
 - c) yardas (1 yarda son 91,44cm).



Estructuras de control secuenciales

► Ejercicio

ALGORITMO Convierte

ENTRADAS:

Km: REAL; Número de km

SALIDAS:

Millas: REAL ; Número de millas

Pies: REAL ; Número de pies

Yardas: REAL ; Número de yardas

VARIABLES:

Km: REAL

Millas: REAL

Pies: REAL

Yardas: REAL

INICIO

ESCRIBA "Escribe un número: "

LEA Km

Millas \leftarrow Km / 1.60 ;

Pies \leftarrow Km * 100000 / 30.48 ;

Yardas \leftarrow Km * 100000 / 91.44;

ESCRIBA "Las millas son:"

ESCRIBA Millas

ESCRIBA "Las yardas son:"

ESCRIBA Yardas

ESCRIBA "Los pies son:"

ESCRIBA Pies

ESCRIBA "Fin del algoritmo"

FIN

Estructuras de control secuenciales

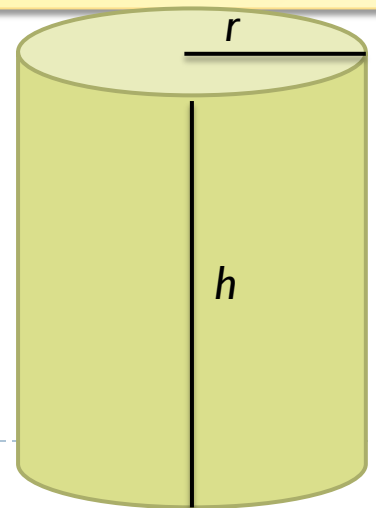
```
//Librería que contiene las funciones scanf y printf
#include <stdio.h>
//Función principal del programa
int main () {
    // Declaro las variables de mi función
    float Km, Millas, Yardas, Pies;
    printf("Escribe un número de kilómetros: ");
    scanf("%f", &Km); //Guarda el número leído en la variable Km
    //Convierto los km a millas:
    Millas = Km /1.60;
    //Convierto los km a pies:
    Pies = Km * 100000 / 30.48 ;
    //Convierto los km a yardas:
    Yardas = Km * 100000 / 91.44;
    //Imprimo los resultados
    printf("Las millas son:%f \nLas yardas son:%f \n Los pies son:
           %f \nFin del algoritmo\n", Millas, Yardas, Pies);
    //Fin del programa
    return 0;
}
```

Estructuras de control secuenciales

► Ejercicio

EJERCICIO

1. Realice un algoritmo en pseudocódigo que calcule el volumen de un cilindro. El usuario ha de introducir el radio y la altura del cilindro.



Estructuras de control secuenciales

► Ejercicio

ALGORITMO VolumenCilindro

ENTRADAS:

h: REAL; Altura del cilindro

r: REAL; Radio de la base

SALIDAS:

volumen: REAL; Volumen del cilindro

VARIABLES:

h: REAL

r: REAL

volumen: REAL

INICIO

ESCRIBA "Introduzca el radio de la base:"

LEA r

ESCRIBA "Introduzca la altura del
cilindro:"

LEA h

volumen $\leftarrow 3.14 * r * r * h$

ESCRIBA "El volumen es:"

ESCRIBA volumen

ESCRIBA "Fin del algoritmo"

FIN

Estructuras de control secuenciales

```
#include <stdio.h> //Contiene scanf y printf

//Función principal del programa
int main ()
{
    // Declaro las variables de mi función
    float radio, altura, volumen;

    printf("Escribe el radio del cilindro: ");
    scanf("%f", &radio); //Guarda el número leído en radio
    printf("Escribe la altura del cilindro: ");
    scanf("%f", &altura); //Guarda el número leído en altura
    //Calculo el volumen
    volumen= 3.14 * radio * radio * altura;
    //Imprimo el resultado
    printf("El volumen del cilindro es %f", volumen);

    //Fin del programa
    return 0;
}
```


Estructuras de control secuenciales

► Ejercicio

EJERCICIO

1. Crea un programa que lea una temperatura en grados Celsius por teclado y lo pase a Farenheit imprimiendo el resultado por pantalla.

