

Estructura y Representación de Datos

Prof. Tatiana Ilabaca
Primer semestre 2021



Módulo 1
Introducción y conceptos
fundamentales

Conceptos fundamentales

Objetivos

- Conocer el concepto de tipo de dato y su representación en memoria
- Conocer una clasificación de los tipos de datos
- Conocer las características de un tipo de dato simple
- Aplicar los tipos de datos simples

Tipo de dato

Dato

- Representación formal de hechos, conceptos, etc., adecuada para su comunicación, interpretación y procesamiento por seres humanos o medios automáticos.
- Se caracteriza por:
 - Nombre
 - Valor
- Ejemplos:
 - Número de visitas a una página
 - Fecha de inscripción
 - Nombre de un laboratorio
 - Promedio de temperatura
 - etc.

Tipo de dato

Definición

- Especificación de un **dominio** (rango de valores) y de un conjunto válido de **operaciones**, a los que normalmente los traductores asocian un **esquema de representación interno** propio.
- Por ejemplo, en los lenguajes C y Java:
 - **int** representa a un valor numérico entero de 32 bits (4 bytes)
 - Rango de representación: - 2.147.483.648 al 2.147.483.647

Tipo	Tamaño (bytes)
char, unsigned char	1
short int, unsigned short int	2
int, unsigned int, long int, unsigned long int	4
float	4
double	8
long double	12

Tipos de datos en lenguaje C

Tipo de dato

Representación en memoria

- La memoria se compone de un conjunto de bytes **numerados consecutivamente**
- Se puede almacenar datos en **celdas** de memoria formadas por uno o más bytes, según sea la longitud del dato que se vaya a almacenar
- Por ejemplo, si en un programa se definen las siguientes variables:

```
short int a;    //2 bytes  
int b;         //4 bytes  
char c;        //1 byte  
  
a=10;  
b=80;  
c='A';
```

Tipo de dato

Representación en memoria

- La distribución en la memoria podría ser la siguiente:

```
short int a;    //2 bytes
int b;         //4 bytes
char c;        //1 byte

a=10;
b=80;
c='A';
```

Dirección de a: 24 Celda de a: 2 bytes Valor de a: 10

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59
60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99

Direccionamiento de la memoria.

Dirección de b: 50 Celda de b: 4 bytes Valor de b: 80

Tipo de dato

Tipo de dato (clasificación)

De acuerdo a *quién* lo define

- TD estándar (L.P)
 - `int`, `float`, `char`, etc.
- TD definido por el usuario
 - `typedef`

De acuerdo a su *representación interna*

- TD simple (primitivo)
 - `int`, `float`, `char`, etc.
- TD estructurado
 - `arreglos`, `listas`, etc.

Tipo de dato

Tipo de dato simple

- Ocupa sólo **una celda** de memoria
- Hace referencia a un **único valor** a la vez
- En este grupo se encuentran los números enteros y reales, caracteres, booleanos y enumerados.

- Por ejemplo:

Edad

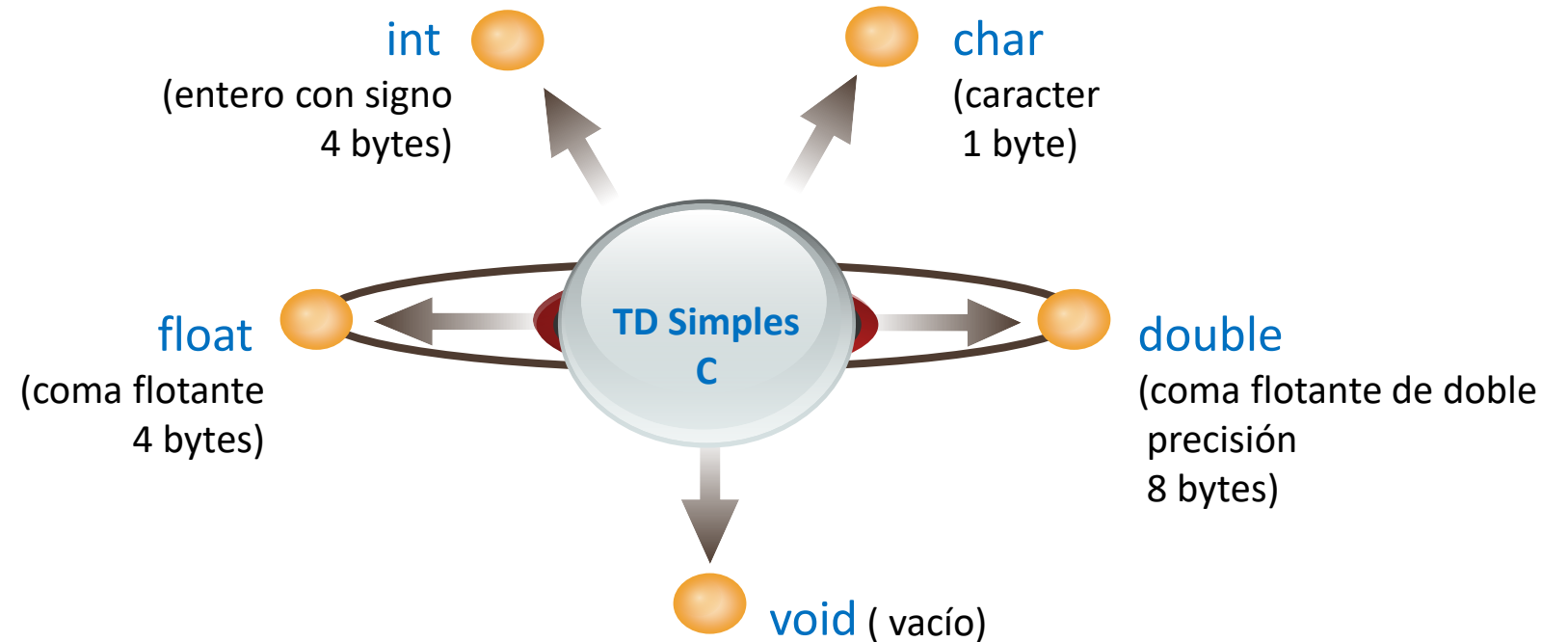
21

Nota

6.5

Tipo de dato

Tipos de datos simples en C



Actividad

Implementa en lenguaje C un programa que sume los dígitos de un número entero par positivo.

Análisis:

- Requisitos del valor de entrada
 - Número entero
 - Número par
 - Número positivo
- ¿Qué hacer con el número válido?
 - Sumar los dígitos que lo componen

Actividad

```
#include <stdio.h>

void main()
{
    int numero;
    int nro;
    short int digito;
    short int suma;

    do
    {
        printf("Ingrese un numero entero par positivo: ");
        scanf("%i",&numero);

        if(numero <= 0)
            printf("El numero ingresado no es positivo\n");

        digito=numero%10;

        if(digito%2 != 0)
            printf("El numero ingresado no es par\n");

    }while(numero <= 0 || digito%2 != 0);
```

Actividad

```
nro=numero;
suma=0;

while(numero != 0)
{
    suma=suma + digito;
    numero=numero/10;
    digito=numero%10;
}

printf("La suma de los digitos del numero %i es: %i",nro,suma);
}
```

Actividad

Desarrollar la **Guía de Práctica 1** publicada en el Aula Virtual.