Tipos de datos

### Lenguaje C

La palabra unsigned en realidad es un modificador aplicable a tipos enteros, aunque si no se especifica un tipo se supone int. El modificador es una palabra clave de C que indica que una variable, o función, no se comporta de la forma normal. Hay también un modificador signed, pero como los tipos son por defecto con signo, casi no se utiliza. Las variables de coma flotante son siempre con signo y, en el caso en que el exponente sea positivo, puede obviarse el signo del mismo. La siguiente tabla muestra todas las combinaciones permitidas que se ajustan al estándar ANSI.

|  |  |  |
| --- | --- | --- |
| **Tipo** | **bits** | **Intervalo** |
| **char** | 8 | -128 a 127 |
| **unsigned char** | 8 | 0 a 255 |
| **signed char** | 8 | -128 a 127 |
| **int** | 16 | -32 768 a 32 767 |
| **unsigned int** | 16 | 0 a 65535 |
| **signed int** | 16 | igual que int |
| **short int** | 8 | -128 a 127 |
| **unsigned short int** | 8 | 0 a 255 |
| **signed short int** | 8 | igual que short int |
| **long int** | 32 | -2 147 483 648 a 2147 483 647 |
| **signed long int** | 32 | -2 147 483 648 a 2 147 483 647 |
| **unsigned long int** | 32 | 0 a 4 294 967 296 |
| **float** | 32 | 3.4E-38 a 3.4E+38 |
| **double** | 64 | 1.7E-308 a 1.7E+308 |
| **long double** | 128 | 3.4E-4932 a 3.4E+4932 (aprox. 24 dígitos de precisión) |

Es preciso tener presente que en sistemas que permiten longitudes de palabra mayores de 16 bits un **short int** puede no ser igual que un **signed char**.

### Lenguaje C++

Existen varios tipos de variables, y cada uno corresponde a un tamaño máximo de un número, un carácter o incluso una verdad. Cuanto mayor sea el número que pueda admitir, mas espacio en memoria ocupará. 

1.1 bool

Por lo general utiliza **1** byte de memoria, valores: **true** o **false**. 

1.2 char

Utiliza generalmente **1** byte de memoria, permite almacenar un carácter, valores; **256** caracteres. 

1.3 unsigned short int

Utiliza generalmente **2** bytes de memoria, valores: de **0** a **65 535** 

1.4 short int

Utiliza generalmente **2** bytes de memoria, valores: de **-32768** a **32767**. 

1.5 unsigned long int

Utiliza generalmente **4** bytes de memoria, valores: de **0** a **4 294 967 295**. 

1.6 long int

Utiliza generalmente **4** bytes de memoria, valores: de **-2 147 483 648** a **2 147 483 647**. 

1.7 int (16 bits)

Utiliza generalmente **2** bytes de memoria, valores: de **-32 768** a **32 767**. 

1.8 int (32 bits)

Utiliza generalmente **4** bytes de memoria, valores: de **-2 147 483 648** a **2 147 483 647**. 

1.9 unsigned int (16 bits)

Utiliza generalmente **2** bytes de memoria, valores: de **0** a **65 535**. 

1.10 unsigned int (32 bits)

Utiliza generalmente **2** bytes de memoria, valores: de **0** a **4 294 967 295**. 

1.11 double

Utiliza generalmente **8** bytes de memoria, valores: de **2.2e-308** a **3.4e-38**. 

1.12 float

Utiliza generalmente **4** bytes de memoria, valores: de **1.2e-308** a **3.4e-38**.   
Atención!   
El tamaño de las variables en memoria puede variara de un PC a otro.

### Lenguaje Java

Estas últimas son utilizadas por las matrices, las clases y las interfaces.

|  |  |  |
| --- | --- | --- |
| Tipos de datos primitivos | | |
| **Tipo** | **Tamaño y formato** | **Rango** |
|  | enteros |  |
| byte | 8 bits - complemento a 2 | {\displaystyle -2^{7}\ al\ 2^{7}-1} |
| short | 16 bits - complemento a 2 | {\displaystyle -2^{15}\ al\ 2^{15}-1} |
| int | 32 bits - complemento a 2 | {\displaystyle -2^{31}\ al\ 2^{31}-1} |
| long | 64 bits - complemento a 2 | {\displaystyle -2^{63}\ al\ 2^{63}-1} |
|  | números reales |  |
| float | 32 bits - IEEE 754 |  |
| double | 64 bits - IEEE 754 |  |
|  | otros |  |
| char | 16 bits - caracteres UNICODE | '\u0000' al '\uffff' |
| boolean | 1 bit | true o false |

En otros lenguajes de programación, el formato o el tamaño de los tipos primitivos dependen del microprocesador o del sistema operativo en el que se están ejecutando. En cambio, Java pretende ser independiente de la plataforma y mantiene los formatos sin cambios. Para los caracteres alfanuméricos utiliza la codificación UNICODE de 16 bits, para permitir la inclusión de varios alfabetos.