В C съществуват няколко основни типа данни:

* **Char** – един байт; може да съдържа един символ от локалната кодова таблица.
* **Int** – цяло число; обикновено отразява естествената големина на целите числа на хоста.
* **Float** – число с плаваща запетая и единична точност.
* **Double** – число с плаваща запетая и двойна точност.

Към типа int могат да бъдат приложени определенията **short** и **long**.

Short и int числата трябва да са поне 16 бита. Short не трябва да бъде по-дълго от int. Long числата са поне 32 бита и са по-дълги от int.

| **Data Type**  | **Memory (bytes)**  | **Range**  | **Format Specifier**  |
| --- | --- | --- | --- |
| short int  | 2  | -32,768 to 32,767  | %hd  |
| unsigned short int  | 2  | 0 to 65,535  | %hu  |
| unsigned int  | 4  | 0 to 4,294,967,295  | %u  |
| int  | 4  | -2,147,483,648 to 2,147,483,647  | %d  |
| long int  | 4  | -2,147,483,648 to 2,147,483,647  | %ld  |
| unsigned long int  | 4  | 0 to 4,294,967,295  | %lu  |
| long long int  | 8  | -(2^63) to (2^63)-1  | %lld  |
| unsigned long long int  | 8  | 0 to 18,446,744,073,709,551,615  | %llu  |
| signed char  | 1  | -128 to 127  | %c  |
| unsigned char  | 1  | 0 to 255  | %c  |
| float  | 4  |  | %f  |
| double  | 8  |  | %lf  |
| long double  | 12  |  | %Lf  |

We can use the [sizeof() operator](http://quiz.geeksforgeeks.org/sizeof-operator-c/) to check the size of a variable. See the following C program for the usage of the various data types:

filter\_none

edit
close

play\_arrow

link
brightness\_4
code

|  |
| --- |
| #include <stdio.h> int main() {     int a = 1;     char b ='G';     double c = 3.14;     printf("Hello World!\n");       //printing the variables defined above along with their sizes     printf("Hello! I am a character. My value is %c and "           "my size is %lu byte.\n", b,sizeof(char));     //can use sizeof(b) above as well       printf("Hello! I am an integer. My value is %d and "           "my size is %lu  bytes.\n", a,sizeof(int));     //can use sizeof(a) above as well       printf("Hello! I am a double floating point variable."           " My value is %lf and my size is %lu bytes.\n",c,sizeof(double));     //can use sizeof(c) above as well       printf("Bye! See you soon. :)\n");       return 0; }  |

**Непримитивните типове в C включват указатели, масиви, структури, union тип и функции.[**[**редактиране**](https://bg.wikipedia.org/w/index.php?title=%D0%A2%D0%B8%D0%BF_%D0%B4%D0%B0%D0%BD%D0%BD%D0%B8&veaction=edit&section=14) **|** [**редактиране на кода**](https://bg.wikipedia.org/w/index.php?title=%D0%A2%D0%B8%D0%BF_%D0%B4%D0%B0%D0%BD%D0%BD%D0%B8&action=edit&section=14)**]**

* **Указателят** е променлива, която съдържа адреса на дадена променлива.
* **Масивите** са поредици от наредени последователно в паметта елементи (променливи) от един и същ тип, които могат да бъдат указани поотделно, като се добави индекс към името, с което са декларирани. Деклариране на масив в C:

type arrayName [ arraySize ];

* **Структурата** се използва за представяне на запис. Може да съдържа например полета за книги: име на книга, автор, заглавие, id.
* **Union** е специален тип данни в C, който позволява да се съхраняват различни типове данни на една и съща позиция в паметта. Може да се дефинира union с много членове, но само един член може да има стойност в даден момент. Осигурява се ефективен начин за използване на място в паметта.

Специален тип **void** показва, че не е налична стойност

https://www.programiz.com/c-programming/c-data-types