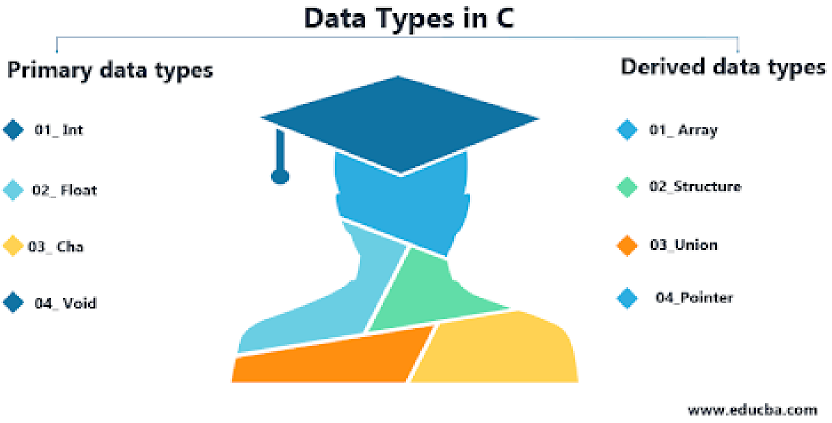
EDUCBA
(https://www.educba.com/software-development/)

← (https://www.educba.com/clock-in-c/)

→ (https://www.educba.com/variables-in-c/)



Introduction to Data Types in C

C is a compact, general-purpose computer programming language that Dennis Ritchie developed for Unix operating system at bell laboratories. C is a structured programming language that is machine-independent. Many organizations have used C for developing operating systems, interpreters, device drivers; also, database oracle is written in C and in the modern era, the embedded system designs and IoT development also use C language. C is a compiled language in which the compiler takes responsibility to convert the source code into machine-readable object code. There are various compilers available like – TurboC, Clang, etc.

Types of Data Types in C

1. Whenever a variable is defined in C, it has to be associated with a certain data type.
2. This gives an indication about the amount of memory to be allocated to that variable, and each variable will hold its own unique memory location, except for some cases where the variables point to the same memory location only
3. C has categorized the data types into:

- a. Primary data types
- b. Derived data types

a. The primary data types are also called as primitive data types, and they include the following :

EDUCBA
(https://www.educba.com/software-development/)

- Float
- Char
- Void

b. The derived data types can be among the following :

- Array
- Structure
- Union
- Pointer

Lets now Describe all of them with examples

1. The integer data type (int) : If you have to store the whole numbers, then int can be used as a data type, it can have a range of numbers based upon the size you choose in memory, and it can have either all positive or from negative to positive range of numbers based upon user choice of code design.

Int type	**Size(in bytes)**	**Range allowed**
int or signed int	2	-32,768 - 32767
unsigned int	2	0 to 65535
short int or signed short int	1	-128 - 127
unsigned short int	1	0 to 255
long int or signed long int	4	-2,147,483,648 - 2,147,483,647
unsigned long int	4	0 to 4,294,967,295

For example

Code:

```
#include <stdio.h>
void main()
{
int a = 1;
printf(" %d is the integer value ",a);
unsigned short int x = -3278989;
printf(" %hu is the integer value ",x);
}
```

2. Float data type: Any real number can be stored in the float data type, and here also we can specify the range; based on data type and size selection, a range of numbers is allo

Float Type	Size(in bytes)	Range of Float
Float	4	3.4E-38 to 3.4E+38
double	8	1.7E-308 to 1.7E+308
long double	10	3.4E-4932 to 1.1E+4932

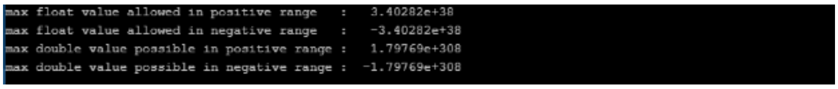
For example

Code:

```
#include <stdio.h>
#include <limits.h>
#include <float.h>
void main() {

printf("max float value allowed in positive range   :   %g\n",
(float) FLT_MAX);
printf("max float value allowed in negative range   :   %g\n",
(double) -DBL_MAX);
}
```

Output :



```
max float value allowed in positive range   :   3.40282e+38
max float value allowed in negative range   :   -3.40282e+38
max double value possible in positive range :   1.79769e+308
max double value possible in negative range :  -1.79769e+308
```

3. Char type: This represents the character data type, and it can be either signed or unsigned with a constant size of 1 byte for both cases.

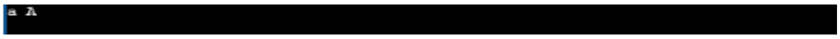
Char Type	Size(in bytes)	Range of char
char or signed char	1	-128 - 127
unsigned char	1	0 to 255

For example

Code:

```
#include <stdio.h>
void main() {
char c ='a';
char f = 65; // represents ASCII char value, refer to ASCII table
printf("%c %c ", c, f);
}
```

Output:



```
a A
```

4. Void type: If you don't want to assign any type to a function (i.e. it won't return anything like you saw the main function prefixed with void type in the above snippets), then you can mark it as void type.

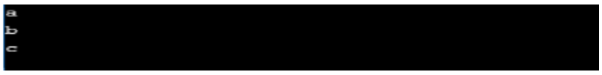
The above snippets can be referred to as examples for the same.

5. Arrays: When any homogenous set of data has to be stored in contiguous memory locations, then this data type is chosen, the use case is that there may be times when your code would return more than one result, and that has to be returned from functions cumulatively, like if we have to find a list of all months in a year, then they will be 12, hence we can't place 12 months discretely in a single variable, so we use arrays for the same.

Let's see a simple snippet to understand the declaration and use of arrays.

```
#include <stdio.h>
void main() {
int i;
char arr[] = {'a', 'b', 'c'};
for(i = 0 ; i < 3 ; i++)
{
printf("%c\n",arr[i]);
}
}
```

Output:



```
a
b
c
```

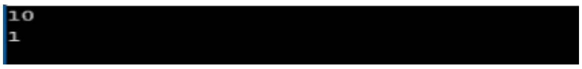
6. Structures: If there is a requirement where you need to represent any physical world structure into the coding world (https://www.educba.com/what-is-coding/), then this type could come in handy, like the class of students can be defined as a structure and student marks and student roll number can be used as variables inside it, an array can be introduced which could hold data related to such structure for many students.

For example

Code:

```
#include <stdio.h&gt
struct class{
int marks;
int rollNo;};
void main() {
struct class c;
c.marks=10;
c.rollNo=1;
printf("%d\n", c.marks);
printf("%d", c.rollNo);
}
```

Output:



```
10
1
```

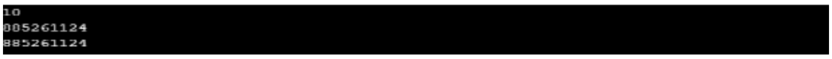
7. Pointer: This is one of the most important data types (https://www.educba.com/c-plus-pluslocation for reading and write operations, that memory location can be the address of a function, can be the address of a variable, etc. Pointers get necessary for Array and structure handling in C language and also provides dynamic memory management.

For example

Code:

```
#include <stdio.h>
void main() {
int a, *p;  // variable and pointer declaration
a = 10;
p = &a;
printf("%d", *p);    // print the value of 'a'
printf("%u", &a);    //print the address of 'a'
printf("%u", p);     // print the address of 'a' in different way
// remember & represents address of variable
}
```

Output:



```
10
005261124
885261124
```

Conclusion

Hence we saw various data types in C and how they work along with C language to handle coding scenarios. You can do embedded programming also with C, as utilities for the same have been developed too. So C is a versatile language, but with real-world scenarios, coding gets complex and more involved.

Recommended articles

This is a guide to Data type in C. Here we discuss the basic concept, different types of data with respective examples, and code implementation. You can also go through our other suggested articles to learn more –

1. Types of Computer Language (https://www.educba.com/types-of-computer-language/)
2. Arrays in C Programming (https://www.educba.com/arrays-in-c-programming/)
3. Patterns in C Programming (https://www.educba.com/patterns-in-c-programming/)

EDUCBA
(https://www.educba.com/software-development/)

Lifetime Access

About Us

Blog (https://www.educba.com/blog/?source=footer)

Who is EDUCBA? (https://www.educba.com/about-us/?source=footer)

Sign Up (https://www.educba.com/software-development/signup/?source=footer)

Corporate Training (https://www.educba.com/corporate/?source=footer)

Certificate from Top Institutions (https://www.educba.com/educbalive/?source=footer)

Contact Us (https://www.educba.com/contact-us/?source=footer)

Verifiable Certificate (https://www.educba.com/software-development/verifiable-certificate/?source=footer)

Reviews (https://www.educba.com/software-development/reviews/?source=footer)

Terms and Conditions (https://www.educba.com/terms-and-conditions/?source=footer)

Privacy Policy (https://www.educba.com/privacy-policy/?source=footer)

Apps

iPhone & iPad (https://itunes.apple.com/in/app/educba-learning-app/id1341654580?mt=8)

Android (https://play.google.com/store/apps/details?id=com.educba.www)

Resources

Free Courses (https://www.educba.com/software-development/free-courses/?source=footer)

Java Tutorials (https://www.educba.com/software-development/software-development-tutorials/java-tutorial/?source=footer)

Python Tutorials (https://www.educba.com/software-development/software-development-tutorials/python-tutorial/?source=footer)

QUIZ

EDUCBA
(https://www.educba.com/software-development/)

All Courses (https://www.educba.com/software-development/courses/?source=footer)

Software Development Course - All in One Bundle (https://www.educba.com/software-development/courses/software-development-course/?source=footer)

Become a Python Developer (https://www.educba.com/software-development/courses/python-certification-course/?source=footer)

Java Course (https://www.educba.com/software-development/courses/java-course/?source=footer)

Become a Selenium Automation Tester (https://www.educba.com/software-development/courses/selenium-training-certification/?source=footer)

Become an IoT Developer (https://www.educba.com/software-development/courses/iot-course/?source=footer)

ASP.NET Course (https://www.educba.com/software-development/courses/asp-net-course/?source=footer)

VB.NET Course (https://www.educba.com/software-development/courses/vb-net-course/?source=footer)

PHP Course (https://www.educba.com/software-development/courses/php-course/?source=footer)

QUIZ