Difference Between Unsigned Int and Signed Int in C

Difficulty Level : Easy • Last Updated : 09 Oct, 2022

Integers are typically stored as 32-bit values, but in some environments, they may contain 16-bit values (or even a different number, usually the product of two powers). For example, let's examine 4-bit integers. They are small but can help illustrate a point. signed int can represent negative values, and unsigned int can only represent non-negative integer values.

For every value which is greater than **INT_MAX** and less than **INT_MIN** we can encounter discontinuity i.e, we can get unexpected results, if we use a signed integer. But for unsigned integer types discontinuity will only be a very large value than **INT_MAX** and a very less value than **INT_MIN**.

We can see the discontinuity of signed integer values i.e, signed integer values get int overflow error as shown in the below program.

C

```c
// C program to show integer
// overflow error signed
// integer
#include <stdio.h>

// Driver code
int main()
{
  int x = 4294967295;
  printf("%d", x);
  return 0;
}
```

```
-1
```

The above program will print -1 as its output because it will be out of range.

But the same above program will run for unsigned values:

C

```c
// C program for unsigned integer
#include <stdio.h>
int main()
{
  unsigned int x;
  x = 4294967295;

  printf("%u", x);
  return 0;
}
```

Output

```
4294967295
```

Difference between Signed Int and Unsigned Int

Signed Int	Unsigned Int
A signed int can store negative values.	Unsigned integer values can only store positive values.
A signed integer can hold values from $-2^{32}/2 - 1$ (	A 32-bit unsigned integer can store only positive values from 0 to $2^{32}-1$ (4294967295)

Signed Int	Unsigned Int
-2147483648) to 2^{32} /2 – 1 (2147483647)	
A signed integer can get an overflow error while used in a program with larger values.	An unsigned integer never gets an overflow error because if the result is greater than the largest value of the unsigned integer type, the value is reduced to the modulo of the largest number plus one which can be represented as an unsigned integer.

Related Articles

1. What happens if loop till Maximum of Signed and Unsigned in C/C++?
2. Difference between const int*, const int * const, and int const *
3. Difference between Signed magnitude and 2's complement
4. Difference between "int main()" and "int main(void)" in C/C++?
5. Difference between sizeof(int *) and sizeof(int) in C/C++
6. Difference between int (*p)[3] and int* p[3]?
7. Difference between int* p() and int (*p)()?
8. Maximum value of signed char in C++
9. Maximum value of unsigned char in C++
10. What's difference between "array" and "&array" for "int array[5]" ?

Previous Next

Article Contributed By :

kamleshjoshi18
@kamleshjoshi18

Vote for difficulty

Current difficulty : Easy

Easy | Normal | Medium | Hard | Expert

Article Tags : C-Data Types, C Language, Difference Between

Improve Article | Report Issue

GeeksforGeeks
A-143, 9th Floor, Sovereign Corporate Tower,
Sector-136, Noida, Uttar Pradesh - 201305

feedback@geeksforgeeks.org