



Class Notes

Published Date: November, 2022

PPS: UNIT-1

Introduction to Components of a Computer System

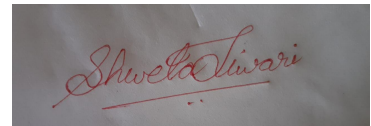
FALL SEMESTER, YEAR (I/II sem, 1st yr)

FALL SESSION (2022-23)

(PPS)

MS. SHWETA TIWARI

Published Date: November, 2022



shwetatiwario8@recabn.ac.in
shwetatiwario8aug@gmail.com

TOPIC On: UNIT-1: Type Conversion in **C**

By SHWETA TIWARI

Under On: Introduction to Components of a Computer System

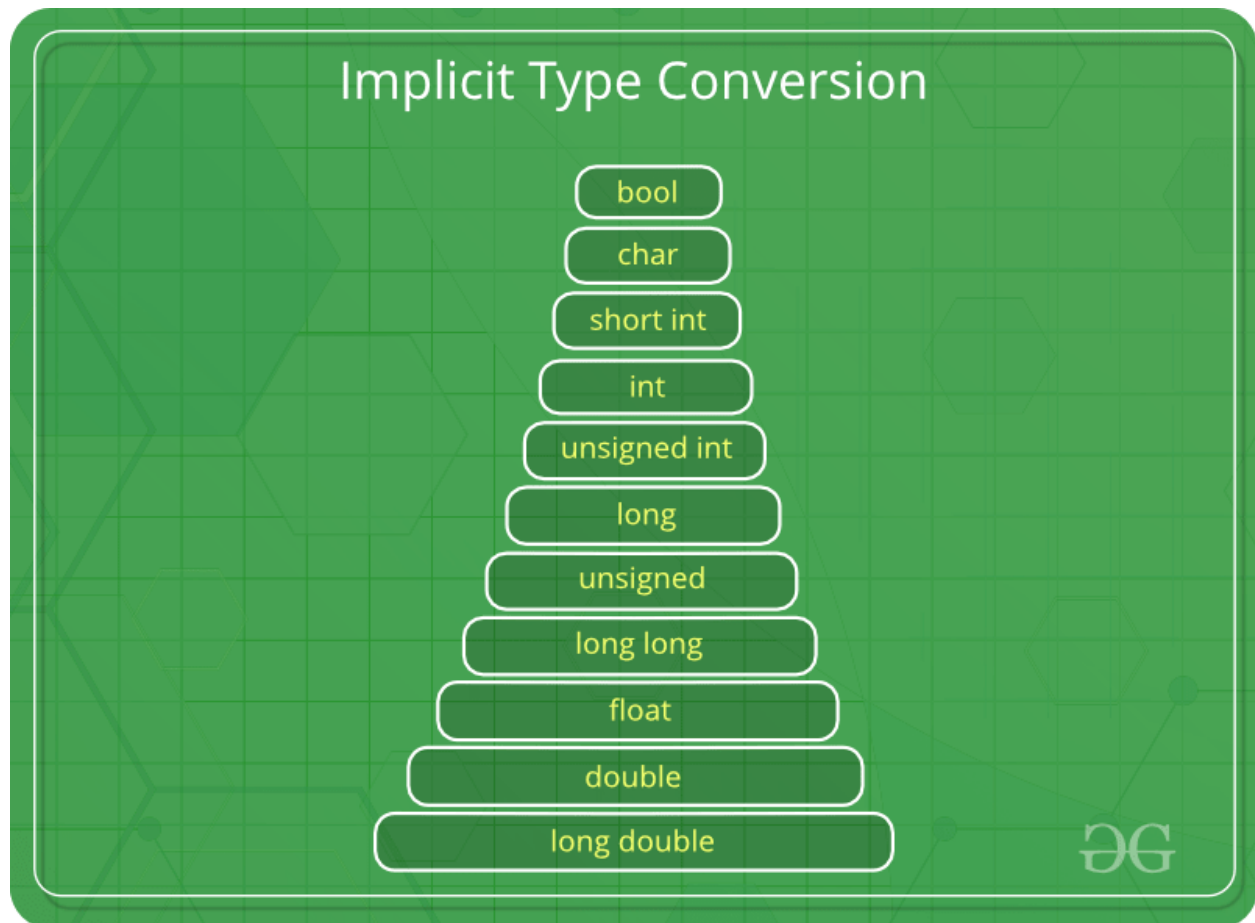
PREPARED FOR
Engineering Students
All Engineering College

PREPARED BY
SHWETA TIWARI

Type Conversion in C

Type conversion in C is the process of converting one data type to another. The type conversion is only performed to those data types where conversion is possible. Type conversion is performed by a compiler. In type conversion, the destination data type can't be smaller than the source data type. Type conversion is done at compile time and it is also called widening conversion because the destination data type can't be smaller than the source data type. ***There are two types of Conversion:***

1. Implicit Type Conversion



Also known as ‘automatic type conversion’.

A. Done by the compiler on its own, without any external trigger from the user.

B. Generally takes place when in an expression more than one data type is present. In such conditions type conversion (type promotion) takes place to avoid loss of data.

C. All the data types of the variables are upgraded to the data type of the variable with the largest data type.

bool -> char -> short int -> int ->
unsigned int -> long -> unsigned ->
long long -> float -> double -> long double

D. It is possible for implicit conversions to lose information, signs can be lost (when signed is implicitly converted to unsigned), and overflow can occur (when long long is implicitly converted to float).

Occurrences of Implicit Type Conversion in C

Implicit type conversion is also called automatic type conversion. Some of its few occurrences are mentioned below:

- Conversion Rank
- Conversions in Assignment Expressions
- Conversion in other Binary Expressions
- Promotion
- Demotion

Example of Type Implicit Conversion

```
// An example of implicit conversion
#include <stdio.h>
int main()
{
    int x = 10; // integer x
    char y = 'a'; // character c

    // y implicitly converted to int. ASCII
    // value of 'a' is 97
    x = x + y;

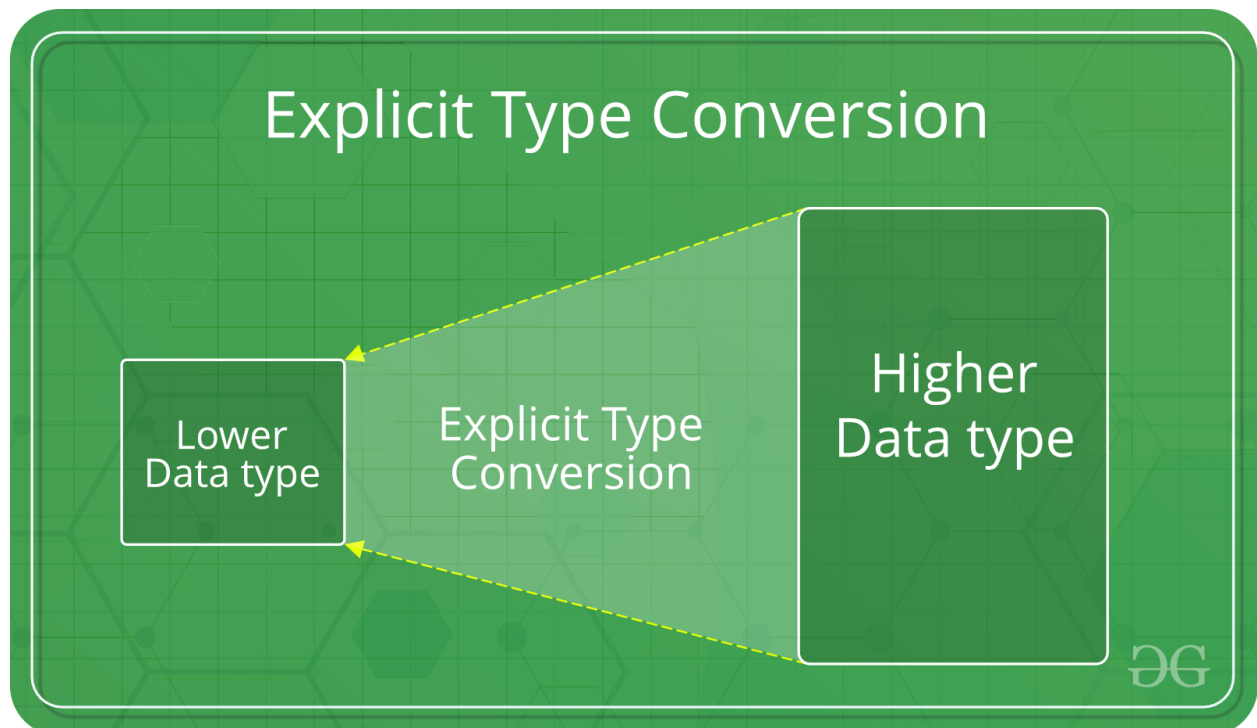
    // x is implicitly converted to float
    float z = x + 1.0;

    printf("x = %d, z = %f", x, z);
    return 0;
}
```

Output

x = 107, z = 108.000000

2. Explicit Type Conversion



This process is also called type casting and it is user-defined. Here the user can typecast the result to make it of a particular data type. The syntax in C Programming:

(type) expression

Type indicates the data type to which the final result is converted.

```
// C program to demonstrate explicit type casting
#include<stdio.h>
```

```
int main()
{
    double x = 1.2;

    // Explicit conversion from double to int
    int sum = (int)x + 1;

    printf("sum = %d", sum);

    return 0;
}
```

Output

sum = 2

Advantages of Type Conversion

- This is done to take advantage of certain features of type hierarchies or type representations.
- It helps us to compute expressions containing variables of different data types.
- The accuracy of our result increases with the help of type conversion.
- Before performing operations, we can refer to the conversion rank hierarchy to get better results.