**Type Casting in C:**

Type Casting in C is used to convert a variable from one data type to another data type, and after type casting compiler treats the variable as of the new data type.

There are two types of type conversion:

1. Implicit Type Conversion

Also known as ‘automatic type conversion’.

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

#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;

}

**Without type casting:**

#include <stdio.h>

void main ()

{

int a;

a = 15/6;

printf("%d",a);

}

**output:2**

**Withtype casting:**

#include <stdio.h>

void main ()

{

float a;

a = (float) 15/6;

printf("%f",a);

}

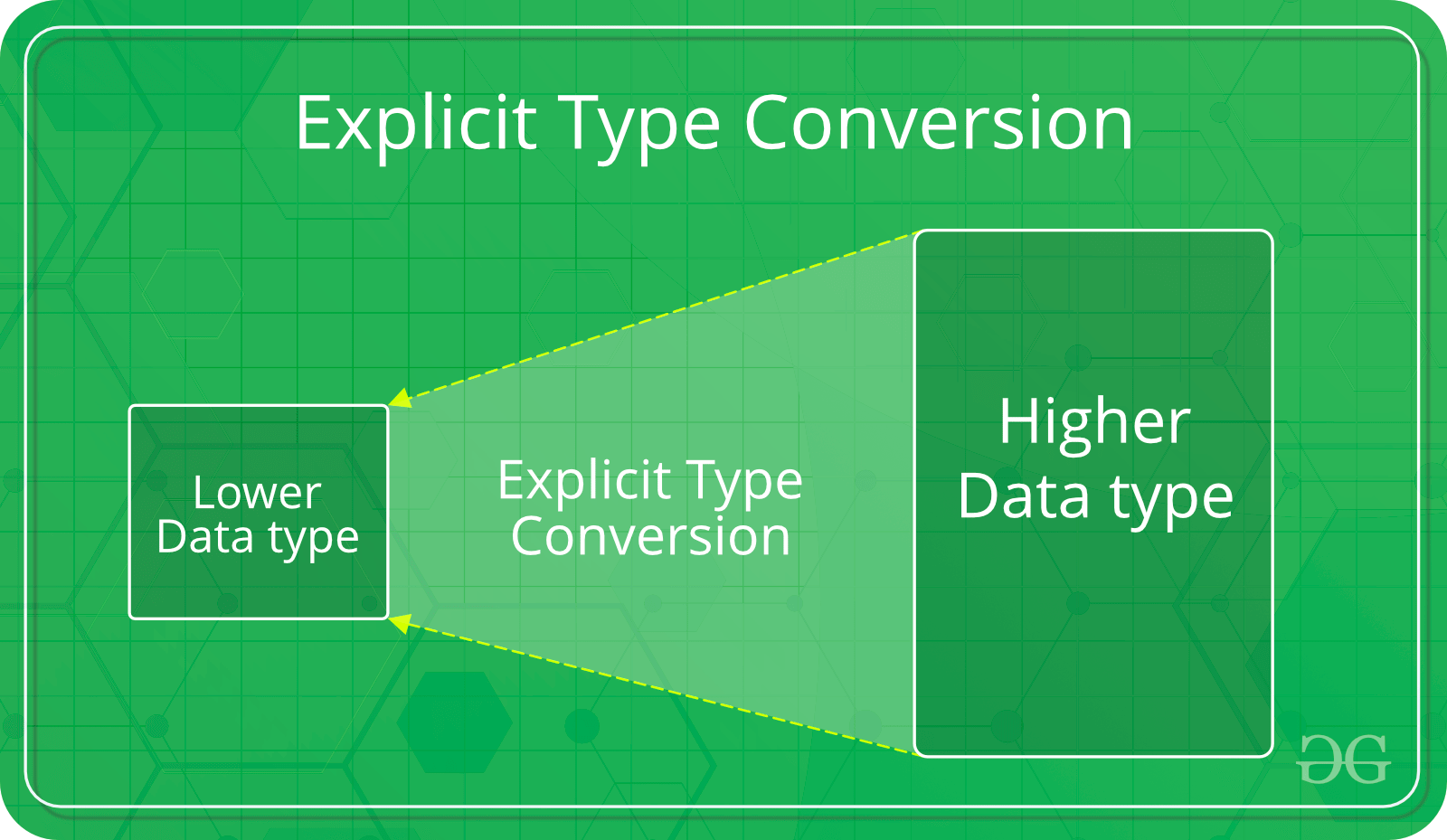
**output:2.500000**

**conversion order:**

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

Explicit Type Conversion:

syntax:(type) expression



#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:2**