Format Specifiers for Floating Point Number

In C programming, floating point numbers are represented using the float, double, and long double data types. When it comes to printing these floating-point values, we have three main format specifiers for printing floating-point numbers: **%f**, **%e**, and **%g**. These specifiers help present numbers in different notations based on the magnitude and precision required.

1. %f: Fixed-Point Notation

The %f format specifier is used to print floating-point numbers in fixed-point notation. This format ensures a fixed number of digits after the decimal point.

- By default, %f prints 6 digits after the decimal point.
- If you enter a number with fewer digits after the decimal point, the remaining spaces are filled with zeros.
- If more than six digits are provided, the extra digits are rounded off, and only the first six digits are printed.

In the case of **long double** values, the format specifier becomes **%Lf** for fixed-point representation.

Example:

C

```c
#include <stdio.h>
int main()
{
    float x = 10.25;
    double y = 125.387648;
    double z = 15E+6;

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

Output

```
10.250000 125.387648 15000000.000000
```

2. %e: Exponential Notation

The **%e** format specifier is used to print numbers in exponential (scientific) notation. In this notation, numbers are represented in the form: 1.2×10^{24}, -2.11×10^{-11},

- The output is always printed with one digit before the decimal and six digits after it.
- The exponent is always printed with at least two digits, including a leading zero if necessary.

The %e notation is used for very large or very small numbers, making them easier to read and understand.

Example:

C

```c
#include <stdio.h>
int main()
{
    float x = 10.25;
    double y = 125.387648;
    double z = 15E+6;

    printf("%e %e %e", x, y, z);
    return 0;
}
```

Output

```
1.025000e+01 1.253876e+02 1.500000e+07
```

3. %g: Mixed Notation

The %g format specifier is a mix between %f and %e. It chooses the most compact representation between fixed-point and exponential notation based on the value of the floating-point number.

- If the number is very small or very large, %g will switch to exponential notation.
- Otherwise, it will print in fixed-point notation.
- %g ensures that the total number of digits printed (before and after the decimal) does not exceed six.
- It eliminates trailing zeros in the fixed-point representation.

Example

C

```c
#include <stdio.h>
int main()
{
    float x = 10.25;
    double y = 125.387648;
```

```
    double z = 15E+6;

    printf("%g %g %g", x, y, z);
    return 0;
}
```

Output

```
10.25 125.388 1.5e+07
```

Important Points

- When printing floating-point numbers, C automatically converts float values to double for output. This means that the format specifiers %f, %e, and %g are used primarily for double values.
- For long double numbers, you need to use %Lf, %Le, and %Lg for fixed-point, exponential, and mixed notations, respectively.