Constants and related programs in c

Constants in C

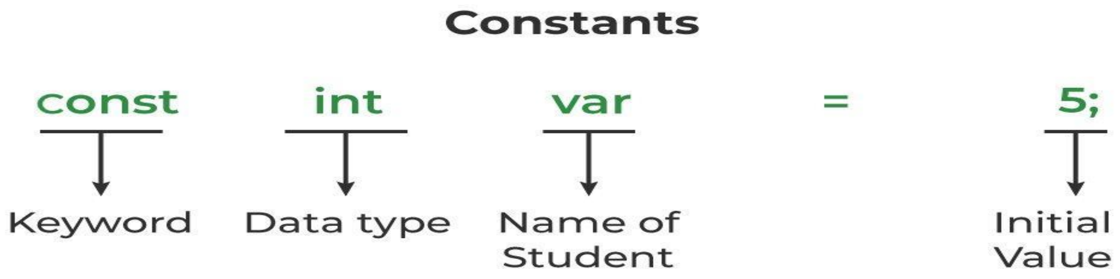
In C programming, constants are read-only values that cannot be modified during the execution of a program. These constants can be of various types, such as integer, floating-point, string, or character constants. They are initialized with the declaration and remain same till the end of the program.

```
#include <stdio.h>

int main() {

       // Defining constant variable

      const a = 10;

     printf("%d", a);

    return 0;

}
```

Syntax

We define a constant in C using the **const** keyword. Also known as a const type qualifier, the const keyword is placed at the start of the variable declaration to declare that variable as a constant.

const *data_type var_name = value;*



2 ways to define constant in C

There are two ways to define constant in C programming.

1. const keyword
2. #define preprocessor

1) C const keyword

The const keyword is used to define constant in C programming.

```
const float PI=3.14;
```

Now, the value of PI variable can't be changed.

```
#include<stdio.h>
int main(){
    const float PI=3.14;
    printf("The value of PI is: %f",PI);
    return 0;
}
```

Output:

```
The value of PI is: 3.140000
```

If you try to change the the value of PI, it will render compile time error.

```
#include<stdio.h>
int main(){
const float PI=3.14;
PI=4.5;
printf("The value of PI is: %f",PI);
    return 0;
}
```

Output:

Compile Time Error: Cannot modify a const object

2) C #define preprocessor

The #define preprocessor is also used to define constant. We will learn about [#define preprocessor directive].

Types of constant:

There are different types of Constants in C. Some of them are as follows:

Decimal Constant

A whole number represented in ***base 10*** is known as a ***decimal constant***. It has digits that range from ***0*** to ***9***. Declaring a ***decimal constant*** has a simple syntax that just requires the value to be written.

Example:

```
#include <stdio.h>

int main() {
int decimal = 42;
printf("The decimal constant is: %d\n", decimal);
return 0;
}
```

Output:

```
The decimal constant is: 42
```

Real or Floating-Point Constant:

A ***fractional component*** or ***exponentiation*** of a number is represented by ***a real or floating-point constant***. It can be expressed with a decimal point, the letter ***"E"***, or the symbol ***"e"*** in exponential or decimal notation.

Example:

```
#include <stdio.h>

int main() {
float real = 3.14;
printf("The real constant is: %f\n", real);
return 0;
}
```

Output:

```
The real constant is: 3.140000
```

Octal Constant:

A ***base 8*** value is represented by an ***octal constant***. It is prefixed with a ***'0' (zero)*** to show that it is an octal constant and has digits ranging from ***0*** to *7*.

Example:

```
#include <stdio.h>
```

```

int main() {
int octal = 052; // Octal representation of decimal 42
printf("The octal constant is: %o\n", octal);
return 0;
}
```

Output:

The octal constant is: 52

Hexadecimal Constant:

A ***base-16*** value is represented by a ***hexadecimal constant***. It uses letters ***A to F*** (or ***a to f***) and numbers ***0*** to ***9*** to represent values from ***10*** to ***15***. It is prefixed with ***'0x'*** or ***'0X'*** to identify it as a hexadecimal constant.

Example:

```
#include <stdio.h>

int main() {
int hexadecimal = 0x2A; // Hexadecimal representation of decimal 42
printf("The hexadecimal constant is: %x\n", hexadecimal);
return 0;
}
```

Output:

```
The hexadecimal constant is: 2a
```

Character Constant

A ***character constant*** represents a ***single character*** that is enclosed in ***single quotes***.

Example:

```
#include <stdio.h>

int main() {
char character = 'A';
printf("The character constant is: %c\n", character);
return 0;
}
```

Output:

```
The character constant is: A
```

String Constant:

A ***series of characters*** wrapped in ***double quotes*** is represented by a ***string constant***. It is a character array that ends with the null character ***\0***.

Example:

```
#include <stdio.h>

int main() {
char string[] = "Hello, World!";
printf("The string constant is: %s\n", string);
return 0;
}
```

Output:

```
The string constant is: Hello, World!
```