- Increment And Decrement Operators in C:-

→ Increment operator in C:-

Syntax:-

```
// AS   PREFIX
++m

// AS   POSTFIX
m++
```

where m is variable.

1. Pre-increment:-

→ Also known as prefix increment, the value is incremented first according to the precedence and then less priority options are done.

```
result = ++ var1
```

The above expression can be expanded as:-

```
var = var+1;
result = var;
```

2. Post-increment:-

→ The increment operations is performed after all the other operations are done. It is also known as postfix increment.

```
result = var1 ++;
```

The above expression is equivalent;-

```
result = var;
var = var+1;
```

```
/*C program to illustrate the increment of both
      types*/
#include<stdio.h>
void  increment ()
{
    int a=5;
    int b =5;

    // PREFIX
    int  prefix = ++a;
    printf("Prefix Increment : %d \n", prefix);

    //POSTFIX
    int postfix = b++;
    printf ("Postfix Increment : %d", postfix);
 }
//Driver code
int main ()
{
    increment();

    return 0;
 }
```

Output:-

```
Prefix  Increment:6

Postfix  Increment:5
```

* The post - ~~incre~~ increment have higher precedence that pre-increment as it is postfix operator while pre-increment comes in unary operator category.

→ Decrement operator in C:-

Syntax:-

```
// AS   PREFIX
--m

// AS  POSTFIX
m--                    where m is variable
```

1Y. Pre- Decrement operator:-

The pre-decrement operator decreases the value of the variable immediately when encontered. It is also known as prefix decrement.

```
result =--m;
```

which can be expanded to

```
m=m-1;
result m;
```

2Y. Post decrement operator:-

In this case, the decrement operation is perforedmed after all the other operators are evaluated.

```
result = m--;
```

The above expression can be expanded as:-

```
 result = m;
 m = m-1;

/*C program to illustrate the decrement operator
  of both types */

#include <stdio.h>

void decrement ()
{
   int a=5;
   int b=5;

   // PREFIX
   int    prefix = --a;
   printf ("Prefix = %d \n", prefix);

    // POSTFIX
    int postfix = b--;
    printf(" Postfix = %d", postfix);
  }

// Driver code

int main( )

{
  decrement ();

   return 0;
  }
```

Output :-

```
Prefix = 4
Postfix = 5
```

37.

* Swap two number without using 3rd variable.

```
# include<stdio.h>
int main()
{
  int a,b;
  printf(" Please enter any number:");
  scanf ("%d", &a);
  printf (" Please enter any number:");
  scanf("%d", &b);
  a=a+b;
  a=a-b;
  a=a-b;
  printf("After swapping, the numbers are %d %d", a,b);
  return 0;
}
```

- Nested conditional operator:-

→ Syntax:-

Condition ? | cond 2 ?__ : __ : | | cond 3 ?__ : __; |

TRUE FALSE

38.

* Find greatest among three no.

```
#include <stdio.h>
int main ()
{
  int a,b,c,d;
  printf ("Enter three no.:");
  scanf ("%d %d %d", &a, &b, &c);
  d= (a>b) ? a>c ? a : c : b>c ? b : c;
  printf(" max is %d", d);
  return 0;
```