Sameer Sir Classes, Jabalpur
Auth Exam Center Oracle, Microsoft
9407077858

POINTER VARIABLE

POINTER IS A VARIABLE WHICH STORED ADDRESS OF THE ANOTHER VARIABLE.

TYPES OF VARIABLES

1. NORMAL VARIABLE
2. POINTER VARIABLE

(VALUE)	(ADDRESS)
int a ;	int* p ; // int *p;
a = value	p = address
&a = self - address	*p = value
	&p = self - address

Var type int

Var type Pointer

a = 5 value
&a = 100 address

1. int* a, p;
type of a = int *
type of p = int *

2 int a, *p;
type of a = int = Value
type of p = int * = Address

FEATURES OF POINTERS

---

1. **SIZE OF THE POINTER = 2 BYTE**

   (int , float , char,...)

   Pointer var size = int size

2. **format = %u // unsigned integer**

   = %x // hexadecimal

   internal work ON HEXA

   a) **signed int a ; // int a;**

   size = 4 byte or 32 bit
   format = %d

   Signed + / −
   first bit + / −

   range = -2,147,483,647 to 2,147,483,647

   b) **unsigned int a;**

   ONLY +

   size = 4 byte
   format = %u

   range = 0 to 4, 294, 967, 295
   0 TO FFFF (HEXA DECIMAL)

3. USE OF POINTER :- DMA (dynamic memory allocation)

4. a) DECLARATION // imp

b) INITIAZATION

c) PROCESSING

5. int *p;

p = address

*p = value

6. int **p;

*p = address

**p = value

7. int ***p;

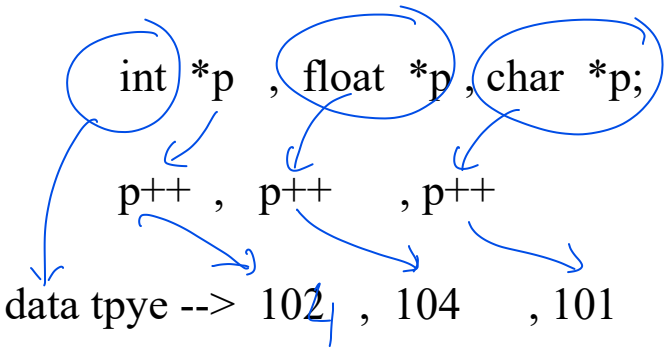
**p = address

***p = value

8. airthmatic operation

- , ++ , -- (address)

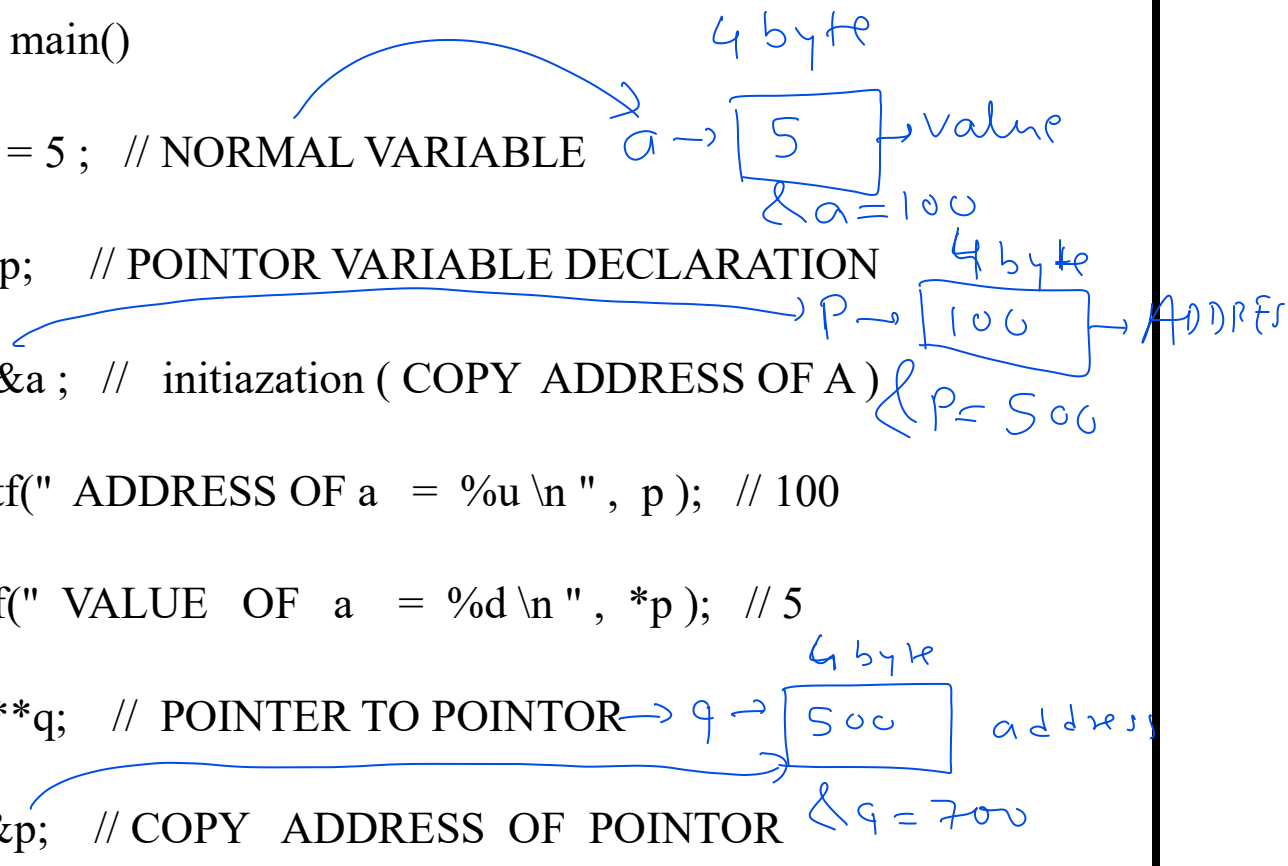
+, * , / X



// WAP TO PRINT VALUE AND ADDRESS OF NORMAL VARIABLE

```
#include<stdio.h>

    int     main()
    {
        int  a = 5 ;

        printf(" VALUE   OF  a  = %d \n " , a);    // 5

        printf(" ADDRESS  OF  a  = %u \n",   &a);  // 65524
    }
```

```
// WAP TO PRINT  VALUE AND ADDRESS OF NORMAL VARIABLE
#include<stdio.h>
int     main()
{
int  a = 5 ;   // NORMAL VARIABLE

int  *p;     // POINTOR VARIABLE DECLARATION

p =  &a ;   //   initiazation ( COPY  ADDRESS OF A )

 printf("  ADDRESS OF a   =  %u \n " ,  p );   // 100

printf("  VALUE   OF   a    =  %d \n " ,  *p );   // 5

 int  **q;    //  POINTER TO POINTOR

q = &p;    // COPY   ADDRESS  OF  POINTOR

 printf("  ADDRESS OF p  = %u \n " ,   q );   // 500
printf("  ADDRESS OF a  = %u \n " ,  *q );   // 100
printf("  VALUE   OF a  = %d \n " , **q );   // 5
 printf("  ADDRESS OF q  = %u \n " ,  &q );   // 700
 }
```



```
// ADDITION OF TWO NOS USING POINTERS

#include<stdio.h>
int   main()
{
              int  a = 5 , b = 2 , c ;
             int  *p , *q ;  // DECLARATION
              p = &a;INITIALIAZATION
              q = &b;

              c = *p + *q ;  // c = a + b;   // processing
              printf(" SUM = %d \n ", c);
}
// DIVISION OF TWO FLOATS NOS  USING POINTER
VARIABLES
 #include<stdio.h>
      int     main()
      {
              float  a = 5 , b = 2 , c;
              float  *p , *q ;       // decalration

          p = &a ;          // initialize
          p= &b ;

          c = *p / (*q) ;    //  c = a / b ;  // processing
          printf(" DIV = %f\n ", c );
      }
```

```
c = *p/*q ;

    /* --> commment
c = *p/ (*q) ;
```

// FACTORIAL USING POINTER VARIABLES

```
#include<stdio.h>
int   main()
{
        int   n, i , f = 1 ;
         int  *p ,*q  ;          // decalration

         p  = &n;                //   initialize
         q   = &f;
        printf(" ENTER NO. \n");
         scanf( "%d" , p ) ;

        for( i = 1; i <= *p ; i++ )
         {
              *q = *q * i ;     //     f = f * i;
         }
         printf(" FACTROIAL = %d\n", *q);
 }
```

// WAP TO FIND EXPONENTIAL SERIES USING POINTOR

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$$

```
#include<stdio.h>
#include<math.h>
   int   main()
  {

            int   n , i , f = 1;                    //  x, n ,f, p, s

            float  x , s = 1 , p = 1;

             printf(" enter x and n \n ");

            scanf("%f%d", &x , &n);

            for( i = 1 ; i <= n ; i++)
             {
                    f = f * i;  // fact

                  p = p * x;

                    s = s + p / f;
             }

      printf(" ans = %f \n" , s);

  }
```