

Factorial using Function

void main ()

```
{
    int n,
    printf("Enter the number");
    scanf("%d", &n); 135
    palindrome(n);
}
```

void palindrome (int n)

```
{
    int rev=0, rem, a=m;
    while(a>0)
    {
        rem = a%10;
        rev = rev*10 + rem;
        a = a/10;
    }
```

```
if ( m == rev)
    printf("Palindrom");
else
    printf("not a Palindrom");
}
```

palindrome

n = 252
m = 252
rev = ~~0~~ ~~2~~ 252
a = m = 252 / 10
rem = 2 / 2

0 × 10 + 2
= 2
2 × 10 + 5
25
25 × 10 + 2
252

not a
palindrome

n = 135
m = 135
rev = ~~0~~ ~~5~~ 531
a = m = 135 / 10
rem = 5 / 31

→

0 × 10 + 5	rev 5 × 10 + 3	53 × 10 + 1
5	50 + 3	530 + 1
	53	531

Palindrome using Function

function → int reverse (int);
 declaration void main ()
 {

```

int n, s;
printf("Enter the number");
scanf("%d", &n); 135
s = reverse (n);
if (s == n)
    printf("Palindrome");
else
    printf("not a palindrome");
}
int reverse (int m) ←
{
    int rev = 0, rem, a = m;
    while (a > 0)
    {
        rem = a % 10;
        rev = rev * 10 + rem;
        a = a / 10;
    }
    return (rev);
}
    
```

function definition

Palindrome

n = 252	0 × 10 + 2	= 2
m = 252	rev = 0 252	2 × 10 + 5
a = m = 252 / 10	25	25
rem = 2	25 × 10 + 2	252

not a
 palindrome

n = 135	0 × 10 + 5	5
m = 135	rev = 0 531	53 × 10 + 3
a = m = 135 / 10	53	530 + 1
rem = 5	531	531

Passing Array to a Function

```

void display( int );
void main()
{
    int arr[] = {4, 16, 83, 21, 96, 15, 9};
    int i;
    for( i=0; i < 7; i++)
    {
        display( arr[i] );
    }
}
    
```

Calling function

Called function

```

void display( int m)
{
    printf( "%d", m);
}
    
```

function call

```

i=0
m = arr[0] = 4
i=1
m = arr[1] = 16
i=2
m = arr[2] = 83
i=3
m = arr[3] = 21
i=4
m = arr[4] = 96
i=5
m = arr[5] = 15
i=6
m = arr[6] = 9
    
```

OUTPUT

4 16 83 21 96 15 9

i=0
*r[0]
4

i=1
*r[1]
16

i=2
*r[2]
83

i=3
*r[3]
21

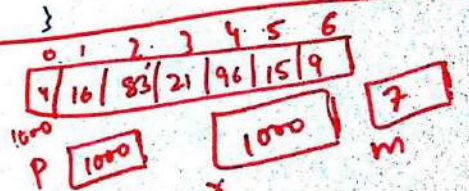
Passing Array using Pointer to a Function

```

void display( int * , int );
void main()
{
    int arr[] = {4, 16, 83, 21, 96, 15, 9};
    int *p;
    p = &arr[0];
    display( p, 7);
}

void display( int *r, int m)
{
    int i;
    for( i=0; i < m; i++)
    {
        printf( "%d", *r[i]);
    }
}
    
```

(1000, 7)



i=2
*r[2]
83

i=3
*r[3]
21