FUNCTION

1. Write a C program with Function which takes a number from the user and print the factorial value of it.

   Sample Input: n = 5

   Output: n! = 120

```
#include<stdio.h>

int fact(int n)
{
   int i, fact = 1;
   for(i=2;i<=n;i++)
      fact*=i;
   return fact;
}

int main()
{
   int n;
   printf("n = ");
   scanf("%d",&n);
   printf("n! = %d \n", fact(n));

   return 0;
}
```

2. Write a C program with Function which takes the value of n & r from the user and print the value of nCr.

   Sample Input: 6 4

   Output: 15

```
#include<stdio.h>

int main()
{
   int n, r;
   printf("Enter the value of n & r: ");
   scanf("%d%d", &n,&r);
   printf("\n %d ", nCr(n,r));

   return 0;
}

int fact(int n)
{
   int i, fact = 1;
   for(i=2;i<=n;i++)
      fact*=i;
   return fact;
}
int nCr(int n, int r)
{
   return fact(n)/(fact(r)*fact(n-r));
}
```

3. Write a C program to print Pascal’s Triangle pattern.

Sample Input: 5

Output:

```
1

1    1

1    2    1

1    3    3    1

1    4    6    4    1

1    5    10   10   5    1
```

```c
#include<stdio.h>
int fact(int n);
int nCr(int n, int r);

int main()
{
   int nth, n, r;
   printf("Enter the value of n: ");
   scanf("%d", &nth);

   for(n=0;n<=nth;n++)
   {
      for(r=0;r<=n;r++)
         printf("%d\t", nCr(n,r));
      printf("\n");
   }

   return 0;
}

int fact(int n)
{
   int i, fact = 1;
   for(i=2;i<=n;i++)
      fact*=i;
   return fact;
}
int nCr(int n, int r)
{
   return fact(n)/(fact(r)*fact(n-r));
}
```

4. Write a C program to print nth Fibonacci number.

   Sample Input: 10

   Output: 55

```
#include<stdio.h>
int fibonacci(int n);

int main()
{
   int n;
   printf("Enter the nth: ");
   scanf("%d", &n);
   printf("\n %d ", fibonacci(n));

   return 0;
}

int fibonacci(int n)
{
   int i,f1=1,f2=1,f3=0;
   if(n>2)
   {
   for(i=3;i<=n;i++)
      {
         f3=f1+f2;
         f1=f2;
         f2=f3;
      }
   return f3;
   }
   else
      return 1;
}
```

5. Write a C program to print nth Fibonacci Series with Function.

Sample Input: 10

```
Output:  1        1         2         3         5         8      13      21      34      55
```

```
#include<stdio.h>

int main()
{
   int n,i;
   printf("Enter the nth: ");
   scanf("%d", &n);
   for(i=1;i<=n;i++)
     printf(" %d \t", fibo(i));

   return 0;
}

int fibo(int n)
{
   static int f1=1,f2=1;
   int i, f3=0;
   if(n>2)
   {
      f3=f1+f2;
      f1=f2;
      f2=f3;
      return f3;
   }
   else
      return 1;
}
```

FUNCTION RECURSION

1. Write a C program with Recursion Function which takes a number from the user and print the factorial value of it

   Sample Input: n = 5

   Output: n! = 120

```
#include<stdio.h>
long int factorial(int n);

int main()
{
   int n;
   printf("n =  ");
   scanf("%d",&n);
   printf("n! =  %ld \n", factorial(n));

   return 0;
}

long int factorial(int n)
{
   if(n<=1)
      return 1;
   else
      return (n * factorial(n-1));
}
```

2. Fibonacci Series with Function Recursion.

Sample Input: 10

Output: 1 1 2 3 5 8 13 21 34 55

```
#include<stdio.h>
int fibo(int n);

int main()
{
   int n,i;
   printf("Enter the nth: ");
   scanf("%d", &n);
   for(i=1;i<=n;i++)
     printf(" %d \t", fibo(i));

   return 0;
}

int fibo(int n)
{
   static int f1=1,f2=1;
   if(n<3)
      return 1;
   else
      return fibo(n-1)+fibo(n-2);
}
```

Global, Static, Automatic and Local Variables

1. Example of Global Variables

   Sample Input: 10

   Output: n = 10

   n = 11

```
#include<stdio.h>
int n; //int n is the global variable
int main()
{
   scanf("%d", &n);
   printf("n = %d \n", n); //Before calling function f1
   f1();
   printf("n = %d \n", n);  //After calling function f1
}
void f1()
{
   n++;
}
```

2. Example of Static Variables

Output with static variable: 11 12 13	Output without static variable: 11 11 11

#include<stdio.h>
 	
int main()
{
   f1();
   f1();
   f1();
}
void f1()
{
   static int n = 10;
   printf("%d \n", ++n);
}
	#include<stdio.h>
	
	int main()
	{
	   f1();
	   f1();
	   f1();
	}
	void f1()
	{
	   int n = 10;
	   printf("%d \n", ++n);
	}

3. Another Example of Static Variables

Sample Input: 10

Output: 1 1 2 3 5 8 13 21 34 55

(Fibonacci series)

```
#include<stdio.h>
long int fibonacci(int n);

int main()
{
   int n,i;
   printf("Enter the nth: ");
   scanf("%d", &n);
   for(i=1;i<=n;i++)
     printf(" %ld \t", fibonacci(i));

   return 0;
}

long int fibonacci(int n)
{
   static int f1=1,f2=1;
   long int f;
   f = (n<3)?1: f1+f2;
   f1=f2;
   f2=f;
   return f;
}
```

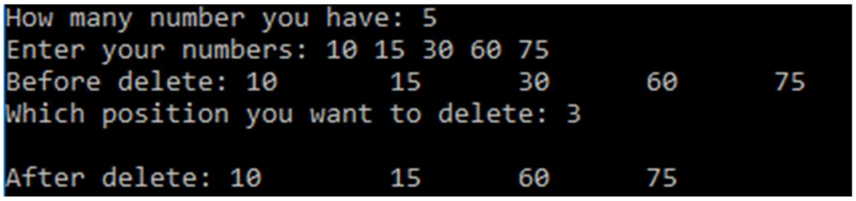
ARRAY

1. Bubble sorting (sorting minimum to maximum)

   Output: 35   50   60   60   80   90

```
#include<stdio.h>

int main()
{
   int mark[]={50,60,35,90,60,80},i,j,temp,size;
   size = sizeof(mark)/sizeof(mark[0]);

   for(i=0;i<size-1;i++)
      for(j=i+1;j<size;j++)
      {
        if(mark[i]>mark[j]) //revers the condition for max-min
            {
               temp=mark[i];
               mark[i]=mark[j];
               mark[j]=temp;
            }
      }

      for(i=0;i<size;i++)
         printf("%d\t", mark[i]);
}
```

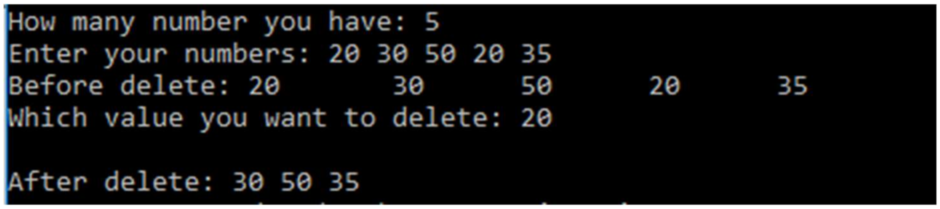
2. Removing element from an array by position.



```
How many number you have: 5
Enter your numbers: 10 15 30 60 75
Before delete: 10       15      30      60      75
Which position you want to delete: 3

After delete: 10        15      60      75
```

```
#include<stdio.h>

int main()
{
    int number[20],n,i,pos;

    printf("How many number you have: ");
    scanf("%d",&n);
    printf("Enter your numbers: ");
    for(i=0;i<n;i++)
        scanf("%d",&number[i]);

    printf("Before delete: ");
    for(i=0;i<n;i++)
        printf("%d\t",number[i]);

    printf("\nWhich position you want to delete: ");
    scanf("%d",&pos);
      for(i=pos-1;i<n-1;i++)
        {
            number[i]=number[i+1];
        }
    printf("\nAfter delete: ");
    for(i=0;i<n-1;i++)
        printf("%d\t",number[i]);
}
```

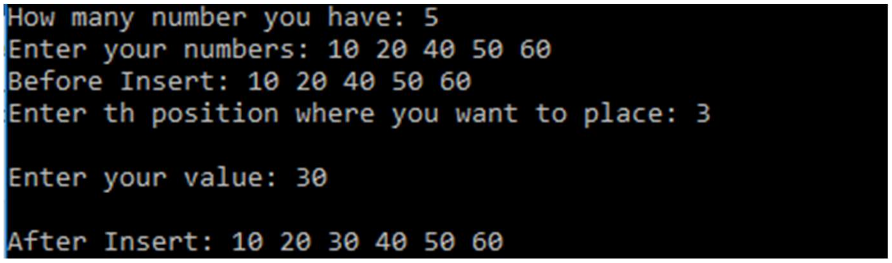
3. Removing elements from an array by value.



```
How many number you have: 5
Enter your numbers: 20 30 50 20 35
Before delete: 20        30        50        20        35
Which value you want to delete: 20

After delete: 30 50 35
```

```
#include<stdio.h>

int main()
{
   int number[20],n,i,j,c=0,value;

   printf("How many number you have: ");
   scanf("%d",&n);
   printf("Enter your numbers: ");
   for(i=0;i<n;i++)
      scanf("%d",&number[i]);

   printf("Before delete: ");
   for(i=0;i<n;i++)
      printf("%d ",number[i]);

   printf("\nWhich value you want to delete: ");
   scanf("%d",&value);

     for(i=0;i<n-1;i++)
         {
            if(number[i]==value)
            {
               for(j=i;j<n;j++)
                  number[j]=number[j+1];
                  c++;
                  i=0;  //If you want to delete every match
            }
         }

   printf("\nAfter delete: ");
   for(i=0;i<n-c;i++)
      printf("%d ",number[i]);

return 0;
}
```

4. Inserting element in an array by position.



```
How many number you have: 5
Enter your numbers: 10 20 40 50 60
Before Insert: 10 20 40 50 60
Enter th position where you want to place: 3

Enter your value: 30

After Insert: 10 20 30 40 50 60
```

```
#include<stdio.h>

int main()
{
  int number[20],n,i,pos,value,temp;

  printf("How many number you have: ");
  scanf("%d",&n);
  printf("Enter your numbers: ");
  for(i=0;i<n;i++)
    scanf("%d",&number[i]);

  printf("Before Insert: ");
  for(i=0;i<n;i++)
    printf("%d ",number[i]);

  printf("\nEnter the position where you want to place: ");
  scanf("%d",&pos);
  printf("\nEnter your value: ");
  scanf("%d",&value);
  number[n]=value;

    for(i=n;i>pos-1;i--)
    {
      temp = number[i];
      number[i]=number[i-1];
      number[i-1]=temp;
    }

  printf("\nAfter Insert: ");
  for(i=0;i<n+1;i++)
    printf("%d ",number[i]);

  return 0;
}
```

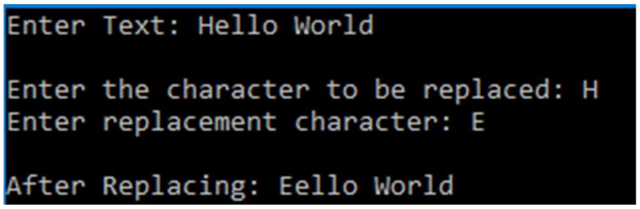
STRING

1. Find the string length.
   Sample Input: Tarikul
   Output: 7

```
#include<stdio.h>
void main()
{
   char name[20];
   int size;
   gets(name);
   size = stringlentgh(name);
   printf("%d",size);
}
int stringlentgh(char str[])
{
   int i=0;
   while(str[i]!='\0')
      i++;
   return i;
}
```

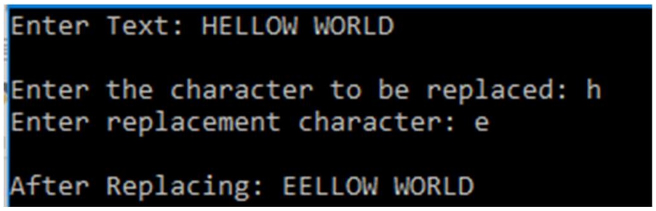
2. Replacing a character from string(case sensitive).

```
#include<stdio.h>
#include<string.h>
void main()
{
   char text[20],ltr,rpltr;
   int i,w;
   printf("Enter Text: ");
   gets(text);
   w = strlen(text); //built-in function
   printf("\nEnter the character to be replaced: ");
   scanf("%c", &ltr);
   getchar();
   printf("Enter replacement character: ");
   scanf("%c", &rpltr);
   for(i=0;i<w;i++)
   {
      if(text[i]==ltr)
         text[i]=rpltr;
   }
   printf("\nAfter Replacement: %s \n", text);
}
```



```
Enter Text: Hello World

Enter the character to be replaced: H
Enter replacement character: E

After Replacing: Eello World
```

3. Replacing a character from string(no case sensitive).

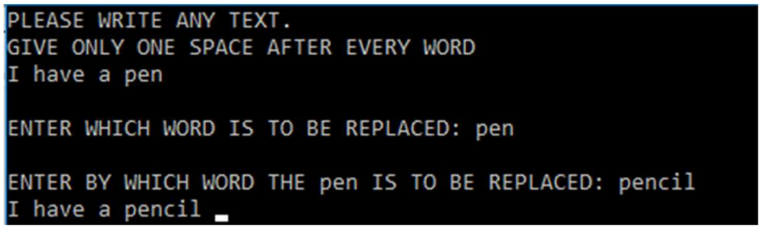


```
Enter Text: HELLOW WORLD

Enter the character to be replaced: h
Enter replacement character: e

After Replacing: EELLOW WORLD
```

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

void main()
{
   char text[20],ltr,rpltr;
   int i,w;
   printf("Enter Text: ");
   gets(text);
   w = strlen(text);
   printf("\nEnter the character to be replaced: ");
   scanf("%c", &ltr);
   getchar();
   printf("Enter replacement character: ");
   scanf("%c", &rpltr);
   for(i=0;i<w;i++)
   {
      if(text[i]==ltr)
         text[i]=rpltr;
      else if(text[i]==ltr-32)
         text[i]=rpltr-32;
      else if(text[i]==ltr+32)
         text[i]=rpltr+32;
   }
   printf("\nAfter Replacing: %s \n", text);

}
```

4. Replacing a word from string(case sensitive).

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

void main()
{
  char text[100],word[10],rpwrd[10],str[10][10];
  int i=0,j=0,k=0,w,p;

      printf("PLEASE WRITE ANY TEXT.\n");
      printf("GIVE ONLY ONE SPACE AFTER EVERY WORD\n");
      gets(text);
      printf("\nENTER WHICH WORD IS TO BE REPLACED: ");
      scanf("%s",word);
      printf("\nENTER BY WHICH WORD THE %s IS TO BE REPLACED: ",word);
      scanf("%s",rpwrd);
      p=strlen(text);

for (k=0; k<p; k++)
{
  if(text[k]!=' ')
    {
      str[i][j] = text[k];
      j++;
    }
  else
    {
      str[i][j]='\0';
      j=0; i++;
    }
}
str[i][j]='\0';
w=i;
    for (i=0; i<=w; i++)
    {
   if(strcmp(str[i],word)==0)
      strcpy(str[i],rpwrd);
        printf("%s ",str[i]);
    }
   getch();
}
```



```
PLEASE WRITE ANY TEXT.
GIVE ONLY ONE SPACE AFTER EVERY WORD
I have a pen

ENTER WHICH WORD IS TO BE REPLACED: pen

ENTER BY WHICH WORD THE pen IS TO BE REPLACED: pencil
I have a pencil _
```

POINTER

1. Explain the output of the flowing code.

   Output: 25

               5

```
#include<stdio.h>

int main()
{
   int n=25;

   f1(n); printf("%d", n);
   f2(&n); printf("\n%d", n);
}

void f1(int n)
{
   n++;
}
void f2(int *p)
{
   *p=5;
}
```

2. Dynamic Array declaration with Memory Allocation - malloc().

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

void main()
{
   char *name;
   int n,i;
   printf("How many characters you have: ");
   scanf("%d",&n);
   name = (char *) malloc(n*sizeof(char));
   printf("Enter Your characters: ");
   for(i=0;i<=n;i++)
      scanf("%c", (name+i));
   for(i=0;i<=n;i++)
      printf("%c", *(name+i));

   free(name);
}
```

STRUCTURE

1. Write a C Program to Store Information of Students Using Structure.

```
#include <stdio.h>
struct student
{
   char name[20];
   long int ID;
   float marks;
};

int main()
{
   int i;
   struct student s[3];

   printf("Enter information of students:\n");

   // storing information
   for(i=0; i<3; ++i)
   {
      printf("\nEnter id number: ");
      scanf("%ld",&s[i].ID);

      printf("Enter name: ");
      scanf("%s",s[i].name);

      printf("Enter marks: ");
      scanf("%f",&s[i].marks);

      printf("\n");
   }

   printf("Displaying Information:\n\n");
   // displaying information
   for(i=0; i<3; ++i)
   {
      printf("\nID number: %ld\n",s[i].ID);
      printf("Name: ");
      puts(s[i].name);
      printf("Marks: %.1f",s[i].marks);
      printf("\n");
   }
   return 0;
}
```