Assignment – 11

1. Accept N numbers from user and accept one another number as NO,check whether NO is Present or not.

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

#define TRUE 1
#define FALSE 0

typedef int BOOL;


BOOL Check(int Arr[],int iLength ,int iNO)
{
   int iCnt = 0;
   int iCount = 0;
   for(iCnt = 0; iCnt < iLength; iCnt++)
   {
      if(Arr[iCnt] == iNO)
      {
         break;
         iCount ++;
      }

   }
    if(Arr[iCnt] == iNO)
    {
       return TRUE;
    }
    else
    {
        return FALSE;
    }


}

int main()
{

   int iSize = 0;
   int *p = NULL;
   int iCnt = 0;
   BOOL bRet = 0;
   int iValue = 0;

   printf("Enter number of Elements :\n");
```

```
    scanf("%d",&iSize);

    printf("Enter the number: \n ");
    scanf("%d",&iValue);

    p = (int*) malloc (iSize * sizeof(int));

    printf("Enter the Elements :\n");

    for(iCnt = 0; iCnt < iSize; iCnt++)
    {
        scanf("%d",&p[iCnt]);

    }

    bRet = Check(p , iSize, iValue);

   if(bRet == TRUE)
   {
     printf("TRUE \n");
   }
   else
   {
     printf("FALSE \n");
   }




    free(p);


    return 0;

}
```

OUTPUT :

```
gcc A11Program1.c -o Myexe
```

1 ./Myexe

```
Enter number of Elements :
6
Enter the number:
 66
Enter the Elements :
85 66 3 66 93 88
TRUE
```

2 ./Myexe

```
Enter number of Elements :
6
Enter the number:
 12
Enter the Elements :
85 11 3 15 11 111
FALSE
```

2. Accept N number from user and accept one another number as NO, return index of first occurrence of that NO.

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



int FirstOcc(int Arr[],int iLength,int iNo)
{
   int iCnt = 0;
   int iIndex = -1;

   for(iCnt = 0; iCnt < iLength; iCnt++)
   {
      if(Arr[iCnt] == iNo)
      {

        iIndex =  iCnt;
        break;


      }

   }
    return iIndex;

}

int main()
{

   int iSize = 0;
   int *p = NULL;
   int iValue = 0;
   int iCnt = 0;
   int iRet = 0;

   printf("Enter number of Elements : \n");
   scanf("%d",&iSize);

   printf("Enter the number : \n");
   scanf("%d",&iValue);

   p = (int*) malloc (iSize * sizeof(int));

   printf("Enter the Elements :\n");

   for(iCnt = 0; iCnt < iSize; iCnt++)
```

```
    {
        scanf("%d",&p[iCnt]);

    }

    iRet = FirstOcc(p , iSize, iValue);

    if(iRet == -1)
    {
        printf("There is no such number %d \n",iRet);
    }
    else
    {
        printf("First occurrence of number is %d \n",iRet);
    }



    free(p);


    return 0;

}
```

OUTPUT :
gcc A11Program2.c -o Myexe

1./Myexe

Enter number of Elements :
6
Enter the number :
66
Enter the Elements :
85 66 3 66 93 88
First occurrence of number is

2 ./Myexe
Enter number of Elements :
6
Enter the number :
12
Enter the Elements :
85 11 3 15 11 111
There is no such number -1

3. Accept N number from user and accept one another number as NO,return index of last occurrence of that No.

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



int FirstOcc(int Arr[],int iLength,int iNo)
{
   int iCnt = 0;
   int iIndex = -1;

   for(iCnt = 0; iCnt < iLength; iCnt++)
   {
      if(Arr[iCnt] == iNo)
      {

        iIndex =  iCnt;

      }

   }
     return iIndex;

}

int main()
{

   int iSize = 0;
   int *p = NULL;
   int iValue = 0;
   int iCnt = 0;
   int iRet = 0;

   printf("Enter number of Elements : \n");
   scanf("%d",&iSize);

   printf("Enter the number : \n");
   scanf("%d",&iValue);

   p = (int*) malloc (iSize * sizeof(int));

   printf("Enter the Elements :\n");

   for(iCnt = 0; iCnt < iSize; iCnt++)
   {
      scanf("%d",&p[iCnt]);

   }
```

```
    iRet = FirstOcc(p , iSize, iValue);

    if(iRet == -1)
    {
       printf("There is no such number %d \n",iRet);
    }
    else
    {
       printf("First occurrence of number is %d \n",iRet);
    }



    free(p);


    return 0;

}
```

OUTPUT :

gcc A11Program3.c -o Myexe

1 ./Myexe
Enter number of Elements :
6
Enter the number :
66
Enter the Elements :
85 66 3 66 93 88
First occurrence of number is 3

2 ./Myexe
Enter number of Elements :
6
Enter the number :
93
Enter the Elements :
85 66 3 66 93 88
First occurrence of number is 4

3 ./Myexe
Enter number of Elements :
6
Enter the number :
12
Enter the Elements :
85 11 3 15 11 111
There is no such number -1

4. Accept N numbers from user and accept Range, Display all elements from that range.

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

void Range(int Arr[],int iLength,int iStart, int iEnd)
{
   int iCnt = 0;


   for(iCnt = 0; iCnt < iLength; iCnt++)
   {
      if((Arr[iCnt] > iStart) && (Arr[iCnt] < iEnd))
      {
         printf("%d \n",Arr[iCnt]);
      }

   }


}
int main()
{

   int iSize = 0;
   int *p = NULL;
   int iValue1 = 0;
   int iValue2 = 0;
   int iCnt = 0;


   printf("Enter number of Elements : \n");
   scanf("%d",&iSize);

   printf("Enter The Starting Point : \n");
   scanf("%d",&iValue1);

    printf("Enter The Ending Point : \n");
   scanf("%d",&iValue2);

   p = (int*) malloc (iSize * sizeof(int));

   printf("Enter the Elements :\n");

   for(iCnt = 0; iCnt < iSize; iCnt++)
   {
```

```
        scanf("%d",&p[iCnt]);
    }
    Range(p , iSize, iValue1,iValue2);

    free(p);

    return 0;
}
```

OUTPUT :

gcc A11Program4.c -o Myexe

1 ./Myexe
Enter number of Elements :
6
Enter The Starting Point :
60
Enter The Ending Point :
90
Enter the Elements :
85 66 3 76 93 88
85 66 76 88

2 ./Myexe
Enter number of Elements :
6
Enter The Starting Point :
30
Enter The Ending Point :
50
Enter the Elements :
85 66 3 76 93 88

5. Accept N number from user and return product of all odd elements.

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

int Product(int Arr[],int iLength)
{
   int iCnt = 0;
   int iProduct = 1;
   if((Arr[iCnt] % 2 )== 0)
   {
     return 0;
   }

   for(iCnt = 0 ; iCnt < iLength; iCnt++)
   {

      if((Arr[iCnt] % 2) != 0)
      {
        printf("%d \n",Arr[iCnt]);

        iProduct = iProduct *  Arr[iCnt];

      }


   }

      return iProduct;
}
int main()
{
    int iSize = 0;
    int *p = NULL;
    int iCnt = 0;
    int iRet = 0;

    printf("Enter number of Elements : \n");
    scanf("%d",&iSize);

    p = (int*)malloc(iSize * sizeof(int));

    if(p == NULL)
    {
        printf("Unable to allocate memory");
```

```
        return -1;
    }
    printf("Enter the Elements : \n");

    for(iCnt = 0; iCnt < iSize; iCnt++)
    {
        scanf("%d",&p[iCnt]);
    }



   printf("Elements of Array are : \n");

    for(iCnt = 0; iCnt < iSize; iCnt++)
    {
        printf("%d\t",p[iCnt]);
    }
    printf("\n");

     iRet = Product(p, iSize);

     printf("Result is %d \n",iRet);

     free(p);




    return 0;
}
```

OUTPUT :

gcc A11Program5.c -o Myexe

1 ./Myexe
Enter number of Elements :
6
Enter the Elements :
15 66 3 70 10 88
Elements of Array are :
15 66 3 70 10 88
15
3
Result is 45

2 ./Myexe
Enter number of Elements :
6
Enter the Elements :
44 66 72 70 10 88
Elements of Array are :
44 66 72 70 10 88
Result is 0