1.Program to implement data structure set operation union intersection and difference using bit String

Program

```
#include<stdio.h>
#include<stdlib.h>
Void main()
{
 Int ch,A[50],B[50],C[50],m,n,I;
 Do
 {
  Printf("\nInput choice to perform: ");
  Printf("\n1.Union\t2.Intersection\t3.Difference\t4.Exit");
  Printf("\nChoice: ");
  Scanf("%d",&ch);
 Switch(ch)
 {
  Case 1:printf("\nEnter cardinality of first set: ");
        Scanf("%d",&m);
         Printf("\nEnter cardinality of second set: ");
    Scanf("%d",&n);
    If(m!=n)
    {
    Printf("\nCannot perform union!");
    Break;
      }
     Printf("\nEnter elements of first set☹0/1) ");
     For(i=0;i<m;i++)
    {
      Scanf("%d",&A[i]);
```

```
 }
Printf(“\nEnter elements of second set: “);
  For(i=0;i<n;i++)
{
   Scanf(“%d”,&B[i]);
 }
Printf(“\nElements of set1 union set2☹0/1) “);
For(i=0;i<m;i++)
{
 C[i]=A[i]|B[i];
 Printf(“%d “,C[i]);
 }
Break;
Case 2:printf(“\nEnter cardinality of first set: “);
    Scanf(“%d”,&m);
     Printf(“\nEnter cardinality of second set: “);
Scanf(“%d”,&n);
If(m!=n)
{
Printf(“\nCannot perform intersection!”);
Break;
  }
  Printf(“\nEnter elements of first set☹0/1) “);
  For(i=0;i<m;i++)
{
   Scanf(“%d”,&A[i]);
 }
Printf(“\nEnter elements of second set: “);
  For(i=0;i<n;i++)
```

```
{
  Scanf("%d",&B[i]);
 }
Printf("\nElements of set1 intersection set2: (0/1)");
For(i=0;i<m;i++)
{
 C[i]=A[i]&B[i];
 Printf("%d ",C[i]);
 }
    Break;
Case 3:printf("\nEnter cardinality of first set: ");
    Scanf("%d",&m);
     Printf("\nEnter cardinality of second set: ");
Scanf("%d",&n);
If(m!=n)
{
Printf("\nCannot perform difference!");
Break;
  }
 Printf("\nEnter elements of first set☹0/1) ");
 For(i=0;i<m;i++)
{
  Scanf("%d",&A[i]);
 }
Printf("\nEnter elements of second set☹0/1) ");
 For(i=0;i<n;i++)
{
  Scanf("%d",&B[i]);
 }
```

```
    For(i=0;i<n;i++)
    {
     If(A[i]==0)
     C[i]=0;
     Else
     {
      If(B[i]==1)
    C[i]=0;
    Else
    C[i]=1;
      }
     }
    Printf("\nElements of set1 – set2: ");
    For(i=0;i<m;i++)
    {
     Printf("%d ",C[i]);
     }
        Break;
  Case 4:printf("\nProgram exit successfully!");
        Exit(0);
         Break;
  Default:printf("\nInvalid choice!");
  };
 }while(1);

}
```

OUTPUT

```
Input choice to perform:
1.Union 2.Intersection 3.Difference 4.Exit
Choice: 1

Enter cardinality of first set: 3

Enter cardinality of second set: 3

Enter elements of first set:(0/1) 1
0
1

Enter elements of second set: 1
0
0

Elements of set1 union set2:(0/1) 1 0 1
Input choice to perform:
1.Union 2.Intersection 3.Difference 4.Exit
Choice: 2

Enter cardinality of first set: 3

Enter cardinality of second set: 3

Enter elements of first set:(0/1) 1
0
1

Enter elements of second set: 1
0
0

Elements of set1 intersection set2: (0/1)1 0 0
Input choice to perform:
1.Union 2.Intersection 3.Difference 4.Exit
Choice: 3

Enter cardinality of first set: 3

Enter cardinality of second set: 3

Enter elements of first set:(0/1) 1
0
1

Enter elements of second set:(0/1) 1
0
0

Elements of set1 - set2: 0 0 1
```