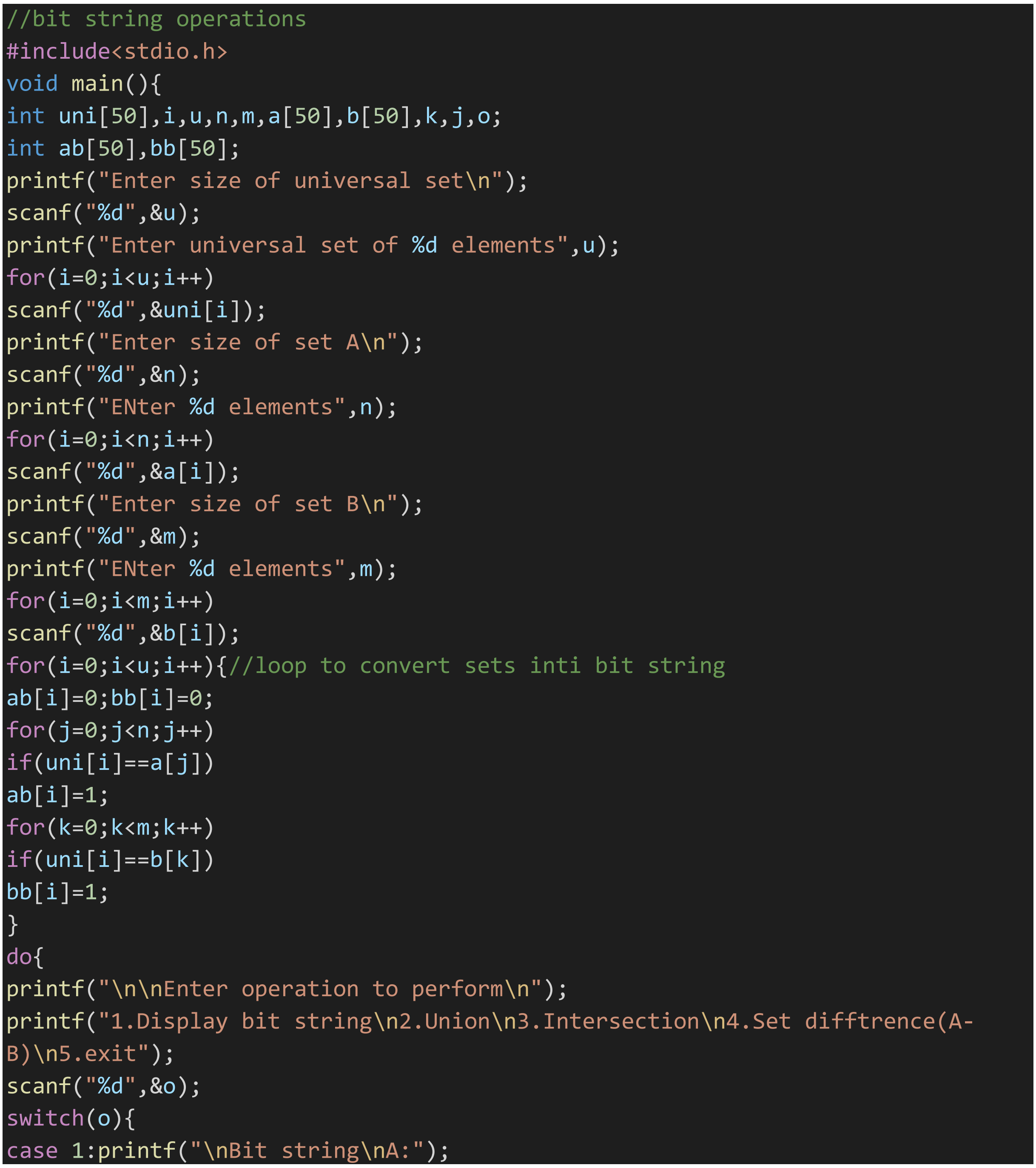
Course outcomes-2

Program 1:

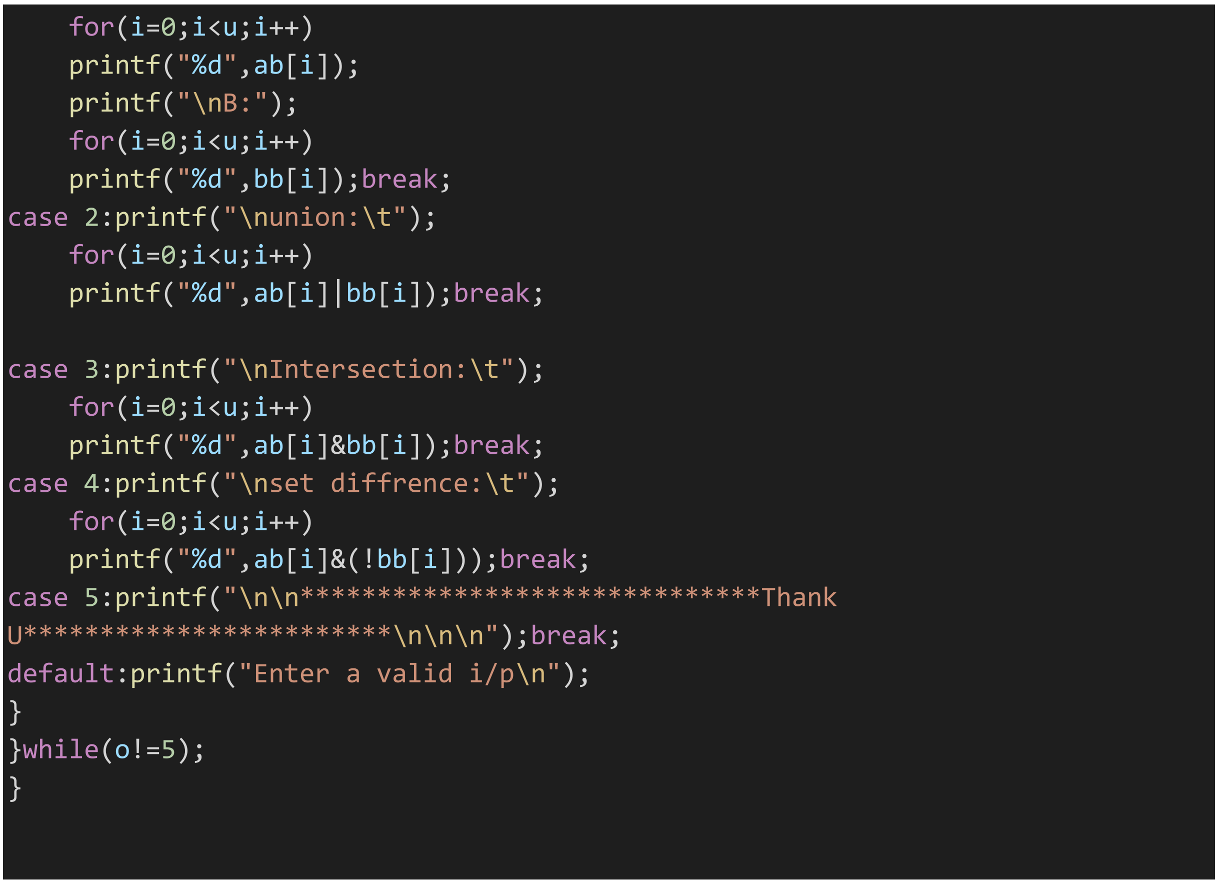
Aim:-

Implementation of BitString Operations using C

Source code:-



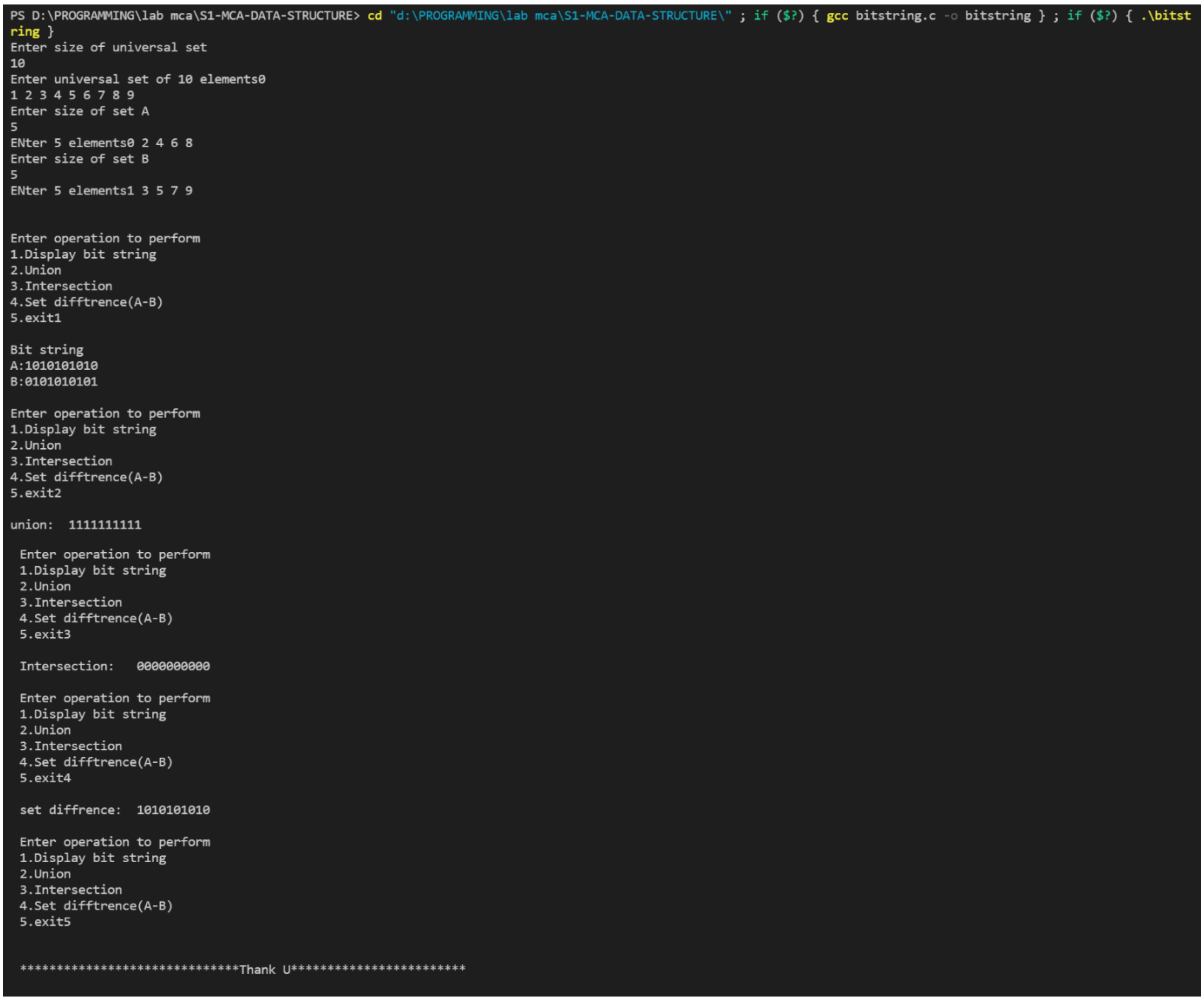
```
//bit string operations
#include<stdio.h>
void main(){
int uni[50],i,u,n,m,a[50],b[50],k,j,o;
int ab[50],bb[50];
printf("Enter size of universal set\n");
scanf("%d",&u);
printf("Enter universal set of %d elements",u);
for(i=0;i<u;i++)
scanf("%d",&uni[i]);
printf("Enter size of set A\n");
scanf("%d",&n);
printf("ENter %d elements",n);
for(i=0;i<n;i++)
scanf("%d",&a[i]);
printf("Enter size of set B\n");
scanf("%d",&m);
printf("ENter %d elements",m);
for(i=0;i<m;i++)
scanf("%d",&b[i]);
for(i=0;i<u;i++){//loop to convert sets inti bit string
ab[i]=0;bb[i]=0;
for(j=0;j<n;j++)
if(uni[i]==a[j])
ab[i]=1;
for(k=0;k<m;k++)
if(uni[i]==b[k])
bb[i]=1;
}
do{
printf("\n\nEnter operation to perform\n");
printf("1.Display bit string\n2.Union\n3.Intersection\n4.Set difftrence(A-B)\n5.exit");
scanf("%d",&o);
switch(o){
case 1:printf("\nBit string\nA:");
```



```
    for(i=0;i<u;i++)
    printf("%d",ab[i]);
    printf("\nB:");
    for(i=0;i<u;i++)
    printf("%d",bb[i]);break;
case 2:printf("\nunion:\t");
    for(i=0;i<u;i++)
    printf("%d",ab[i]|bb[i]);break;

case 3:printf("\nIntersection:\t");
    for(i=0;i<u;i++)
    printf("%d",ab[i]&bb[i]);break;
case 4:printf("\nset diffrence:\t");
    for(i=0;i<u;i++)
    printf("%d",ab[i]&(!bb[i]));break;
case 5:printf("\n\n******************************Thank
U************************\n\n\n");break;
default:printf("Enter a valid i/p\n");
}
}while(o!=5);
}
```

Output:-



```
PS D:\PROGRAMMING\lab mca\S1-MCA-DATA-STRUCTURE> cd "d:\PROGRAMMING\lab mca\S1-MCA-DATA-STRUCTURE\" ; if ($?) { gcc bitstring.c -o bitstring } ; if ($?) { .\bitst
ring }
Enter size of universal set
10
Enter universal set of 10 elements0
1 2 3 4 5 6 7 8 9
Enter size of set A
5
ENter 5 elements0 2 4 6 8
Enter size of set B
5
ENter 5 elements1 3 5 7 9


Enter operation to perform
1.Display bit string
2.Union
3.Intersection
4.Set difftrence(A-B)
5.exit1

Bit string
A:1010101010
B:0101010101

Enter operation to perform
1.Display bit string
2.Union
3.Intersection
4.Set difftrence(A-B)
5.exit2

union:  1111111111

 Enter operation to perform
 1.Display bit string
 2.Union
 3.Intersection
 4.Set difftrence(A-B)
 5.exit3

 Intersection:   0000000000

 Enter operation to perform
 1.Display bit string
 2.Union
 3.Intersection
 4.Set difftrence(A-B)
 5.exit4

 set diffrence:  1010101010

 Enter operation to perform
 1.Display bit string
 2.Union
 3.Intersection
 4.Set difftrence(A-B)
 5.exit5


 ******************************Thank U************************
```