

```

#include <stdio.h>

struct DisjSet {
    int parent[10];
    int rank[10];
    int n;
}dis;

void makeSet()
{
    for (int l = 0; l < dis.n; l++) {
        dis.parent[l] = l;
        dis.rank[l]=0;
    }
}

void displaySet()
{ printf("\nParent Array : \n");
  for (int i = 0; i < dis.n; i++) {
      printf("%d ",dis.parent[i]); }
  printf("\nRank Array : \n");
  for (int i = 0; i < dis.n; i++)
      printf("%d ",dis.rank[i]);
  }
  Printf("\n");
}

int find(int x)
{
    If (dis.parent[x] != x)
    {
        Dis.parent[x] = find(dis.parent[x]);
    }

    return dis.parent[x];
}

```

```

void Union(int x, int y)
{
    int xset = find(x);
    int yset = find(y);
    If (xset == yset)
        return;
    If (dis.rank[xset] < dis.rank[yset]) {
        dis.parent[xset] = yset;
        dis.rank[xset]=-1;
    }
    else if (dis.rank[xset] > dis.rank[yset]) {
        dis.parent[yset] = xset;
        dis.rank[yset]=-1;
    }
    else {
        dis.parent[yset] = xset;
        dis.rank[xset] = dis.rank[xset] + 1;
        dis.rank[yset]=-1;
    }
}

```

```

Int main()

```

```

{ int n,x,y;
    Printf("Enter the no: of elements : ");
    Scanf("%d",&dis.n);
    makeSet();
    int ch,wish;
do
{
    Printf("\n  MENU\n");
    Printf("\n*****\n");
    Printf("1. Union \n2.Find\n3.Display\n");
    Printf("Enter choice : ");

```

```

Scanf("%d",&ch);
Switch(ch)
{
    Case 1: printf("Enter elements to perform union : ");
        Scanf("%d %d",&x,&y);
        Union(x, y);
        Break;
    Case 2: printf("Enter elements to check if connected components : ");
        Scanf("%d %d",&x,&y);
        If (find(x) == find(y))
            Printf("Connected components\n") ;
        Else
            Printf("Not onnected components \n") ;
        Break;
    Case 3: displaySet();
        Break;
}
Printf("\nDo you wish to continue?(1/0) \n");
Scanf("%d",&wish);
}while(wish==1);
return 0;
}

```

## Output:-

Enter the no: of elements : 7

MENU

\*\*\*\*\*

1. Union

2.Find

3.Display

Enter choice : 1

Enter elements to perform union : 3

5

Do you wish to continue ?(1/0)

1

MENU

\*\*\*\*\*

1. Union

2.Find

3.Display

Enter choice : 2

Enter elements to check if connected components : 3

5

Connected components

Do you wish to continue ?(1/0)

1

MENU

\*\*\*\*\*

1. Union

2.Find

3.Display

Enter choice : 3

Parent Array :

0 1 2 3 4 3 6

Rank Array :

0 0 0 1 0 -1 0

Do you wish to continue ?(1/0)

0

[Program finished]