

```

/* This Program takes pairs of people A B in a manner such that A is Smarter than B
after taking N such pairs, it prints the order of smartness of everyone in the
group.
Implemented using topological sorting on a graph maintained via Link Lists.
*/

```

```

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

```

```

/*struct bnode
{
    char C;
    struct bnode* next;
}*S1;

```

```

struct node
{
    char C;
    struct node* next;
};
*/

```

```

int check(int B[50][50],int max)
{
    int i,j;
    for(i=0;i<max;i++)
    {
        for(j=0;j<max;j++)
        {
            if(B[i][j]==1)
                return 1;
        }
    }
    return 0;
}

```

```

int pr(int F[50],int che,int cou)
{
    int i=0;
    for(i=0;i<cou;i++)
    {
        if(F[i]==che)
            return 1;
    }
    return 0;
}

```

```

int main()
{
    int A[50][50];
    int Fin[50];
    int i,j,jmax=0,n,C=0,sum=0,k,flag=1;
    for(i=0;i<50;i++)
        for(j=0;j<50;j++)
        {
            if(i==j)
                A[i][j]=2;
            else

```

```

        A[i][j]=0;

    }
    printf("\n\n Enter the number of pairs  ");
    scanf("%d",&n);
    while(n!=0)
    {
        printf("\n\nEnter the pair in order A B[Where A is smarter than
B(int integer Format)]");
        scanf("%d",&i);
        scanf("%d",&j);
        //--n;
        jmax=j>jmax?j:jmax;
        if(A[i-1][j-1]!=2)
        {
            A[i-1][j-1]=1;
            A[j-1][i-1]=2;
            --n;
        }
        else
            printf("\n\nInvalid pair since %d is already smarter than
%d",j,i);

    }
    while(check(A,jmax))
    {
        flag=1;
        for(i=0;i<jmax;i++)
        {
            for(j=0;j<jmax;j++)
            {
                if(A[j][i]==1)
                    sum++;
            }
            if(sum==0)
            {
                if(!pr(Fin,i,C))
                    Fin[C++]=i;
                for(k=0;k<jmax;k++)
                {
                    if(A[i][k]==1)
                        A[i][k]=0;
                }
                flag=0;
            }
            sum=0;
        }
        if(flag==1)
            break;
        // printf("\n\nWhile runs ");
    }
    // printf("Flag = %d",flag);
    // printf("C: %d,jmax: %d",C,jmax);
    // for(i=0;i<C;i++)
    //     printf("%d",Fin[i]);
    if((C<jmax)|| (flag==1))
        printf("\n\nThe topological ordering is not possible in this

```

```

graph\\n\\n");
    else
    {
        for(i=0;i<jmax-1;i++)
            printf(" %d >",Fin[i]+1);
        //    printf("%d",i);
        printf(" %d ",Fin[i]+1);
    }
    /* for(i=0;i<jmax;i++)
        {for(j=0;j<jmax;j++)
            printf("%d",A[i][j]);
            printf("\\n");
        }
    */
}

```