Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Department of Computer Science and Engineering (Data Science)

Academic Year 2022-23 (Even)

Name of student: DIVYESH KHUNT **SAP ID:60009210116**

Course : Design and Analysis of Algorithms Lab **Course Code:**

Experimentno1

Aim:

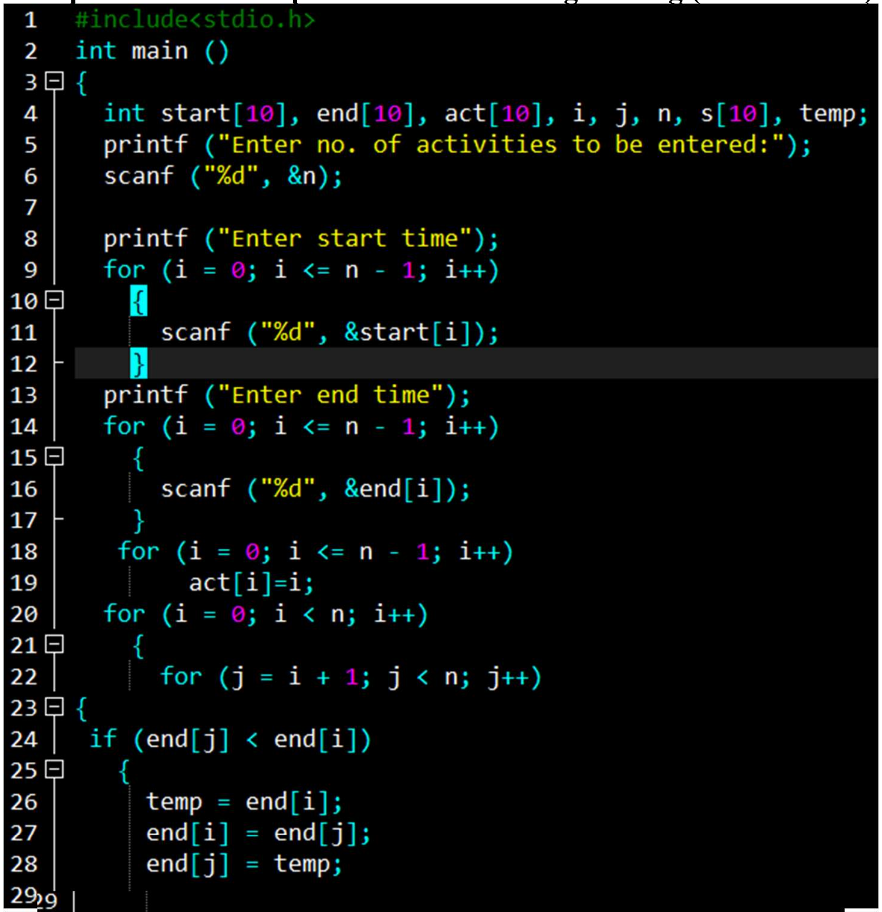
: Implementation of activity selection problem using greedy approach.

Theory:

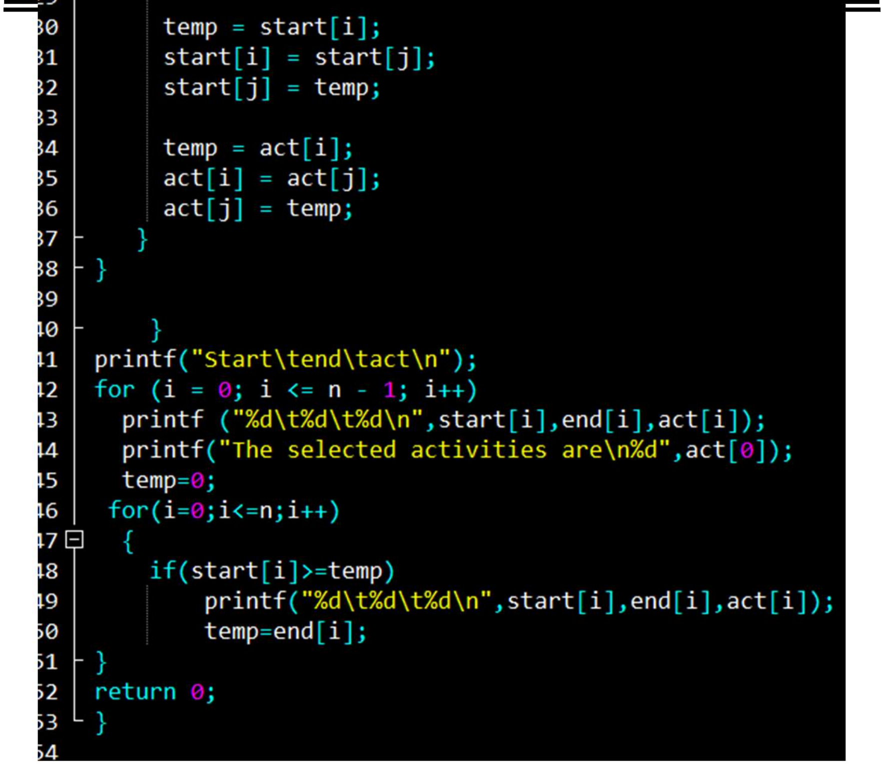
The Activity Selection Problem is an optimization problem which deals with the selection of non-conflicting activities that needs to be executed by a single person or machine in a given time frame. Each activity is marked by a start and finish time. Greedy technique is used for finding the solution since this is an optimization problem.

Algorthm:

```
Activity-Selection(Activity, start, finish)
    Sort Activity by finish times
    stored in finish Selected =
    {Activity[1]}
    n =
    Activity
    j = 1

    for i = 2 to n:
      if start[i] ≥ finish[j]:
         Selected = Selected U
         {Activity[i]} j = i
 return Selected
```

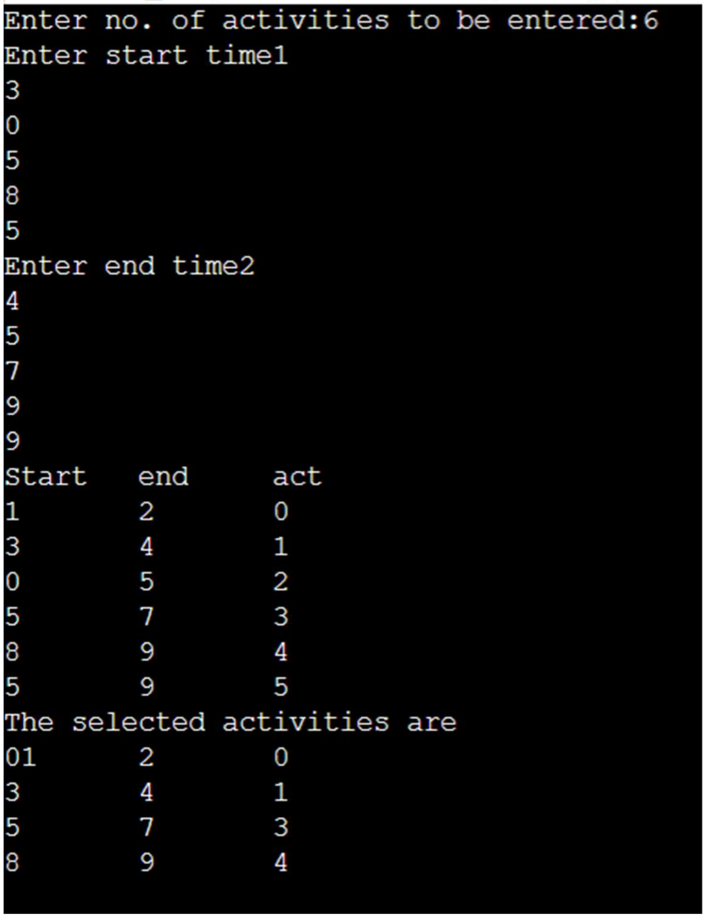


```
#include<stdio.h>
int main ()
{
  int start[10], end[10], act[10], i, j, n, s[10], temp;
  printf ("Enter no. of activities to be entered:");
  scanf ("%d", &n);

  printf ("Enter start time");
  for (i = 0; i <= n - 1; i++)
    {
      scanf ("%d", &start[i]);
    }
  printf ("Enter end time");
  for (i = 0; i <= n - 1; i++)
    {
      scanf ("%d", &end[i]);
    }
   for (i = 0; i <= n - 1; i++)
       act[i]=i;
  for (i = 0; i < n; i++)
    {
      for (j = i + 1; j < n; j++)
{
 if (end[j] < end[i])
   {
     temp = end[i];
     end[i] = end[j];
     end[j] = temp;
```



```

      temp = start[i];
      start[i] = start[j];
      start[j] = temp;

      temp = act[i];
      act[i] = act[j];
      act[j] = temp;
    }
 }


      }
 printf("Start\tend\tact\n");
 for (i = 0; i <= n - 1; i++)
   printf ("%d\t%d\t%d\n",start[i],end[i],act[i]);
   printf("The selected activities are\n%d",act[0]);
   temp=0;
  for(i=0;i<=n;i++)
   {
     if(start[i]>=temp)
         printf("%d\t%d\t%d\n",start[i],end[i],act[i]);
         temp=end[i];
 }
 return 0;
 }

```

Example



```
Enter no. of activities to be entered:6
Enter start time1
3
0
5
8
5
Enter end time2
4
5
7
9
9
Start   end     act
1       2       0
3       4       1
0       5       2
5       7       3
8       9       4
5       9       5
The selected activities are
01      2       0
3       4       1
5       7       3
8       9       4
```

Complexity :

Time Complexity: **O(N * log N)**
Auxiliary Space: **O(N)**

Conclusion:

Thus the code of ACTIVITY SELECTION PROBLEM was successfully implemented