Assignment No.-4

Title:- Job Scheduling

Problem Statement:- Consider the scheduling problem. n tasks to be scheduled on single processor. Let t1, ..., tn be durations required to execute on single processor is known. The tasks can be executed in any order but one task at a time. Design a greedy algorithm for this problem and find a schedule that minimizes the total time spent by all the tasks in the system. (The time spent by one is the sum of the waiting time of task and the time spent on its execution.)

Learning Objectives:-

1. To study the greedy algorithm to solve the job sequencing problem.
2. To study implementation of greedy algorithm using object oriented concepts.
3. To analyze the greedy algorithm.

Learning Outcomes:- After successfully completing this assignment you should be able to:

1. Get the greedy sequence of job by providing different input.
2. How greedy algorithm works for different inputs.

Software Tools Used:- linux GCC compiler, Eclipse Editor on 64-bit Ubuntu 14.04 operating system.

Hardware Required:- PC configuration- Intel Core2duo 2.93 GHZ CPU, 4GB RAM, 320GB HDD, 17" Monitor, Keyboard and Mouse.

Theory:- This problem consists of n jobs each associated with a deadline and profit and our objective is to earn maximum profit. We will earn profit only when job is completed on or before deadline. We assume that each job will take unit time to complete. In this problem we have n jobs j1, j2, … jn each has an associated deadline d1, d2, … dn and profit p1, p2, ... pn. Profit will only be awarded or earned if the job is completed on or before the deadline. We assume that each job takes unit time to complete. The objective is to earn maximum profit when only one job can be scheduled or processed at any given time.

Consider the following 5 jobs and their associated deadline and profit.

index	1	2	3	4	5
JOB	j1	j2	j3	j4	j5
DEADLINE	2	1	3	2	1
PROFIT	60	100	20	40	20

Sort the jobs according to their profit in descending order

Note! If two or more jobs are having the same profit then sort them as per their entry in the job list.

index	1	2	3	4	5
JOB	j2	j1	j4	j3	j5
DEADLINE	1	2	2	3	1
PROFIT	100	60	40	20	20

Find the maximum deadline value

Looking at the jobs we can say the max deadline value is 3.
So, **dmax** = 3
As dmax = 3 so we will have THREE slots to keep track of free time slots. Set the time slot status to EMPTY

time slot	1	2	3
status	EMPTY	EMPTY	EMPTY

Total number of jobs is 5.
So we can write **n** = 5
Note!
If we look at job j2, it has a deadline 1. This means we have to complete job j2 in time slot 1 if we want to earn its profit.
Similarly, if we look at job j1 it has a deadline 2. This means we have to complete job j1 on or before time slot 2 in order to earn its profit.
Similarly, if we look at job j3 it has a deadline 3. This means we have to complete job j3 on or before time slot 3 in order to earn its profit.
Our objective is to select jobs that will give us higher profit.

Algorithm:-

```
Greedy_Sequence()

   1. Start
   2. Sort the jobs according to descending order of their profits
   3. dmax = empty = maximum deadline , i=0
   4. select the $i^{th}$ job and get its deadline in k
   5. check if $k^{th}$ time slot is empty then
             assign the job to $k^{th}$ slot, i++
             if i<=n and empty>0 then go to 4 otherwise go to 7
      otherwise
             k-- , go to 5.
   6. Print the job sequence
   7. Stop
```

Program Code with Sample Output:-

Analysis of Algorithm:-

Time Complexity of this algorithm is $O(n^2)$

Applications:-

1. Graph coloring
2. Finding minimum spanning tree using prims and kruskal
3. Network routing.

Conclusion:-

References:-

1. Horowitz, Sahani, Dinesh Mehata, ―Fundamentals of Data Structures in C++, Galgotia Publisher, ISBN: 8175152788, 9788175152786.
2. Peter Brass, ―Advanced Data Structures‖, Cambridge University Press, ISBN: 978-1-107-43982-5.

Date of Completion:-

FAQ's:-

1. What is the greedy strategy to solve problem?
2. What are different problems solved by greedy technique?
3. What is benefit of greedy strategy?