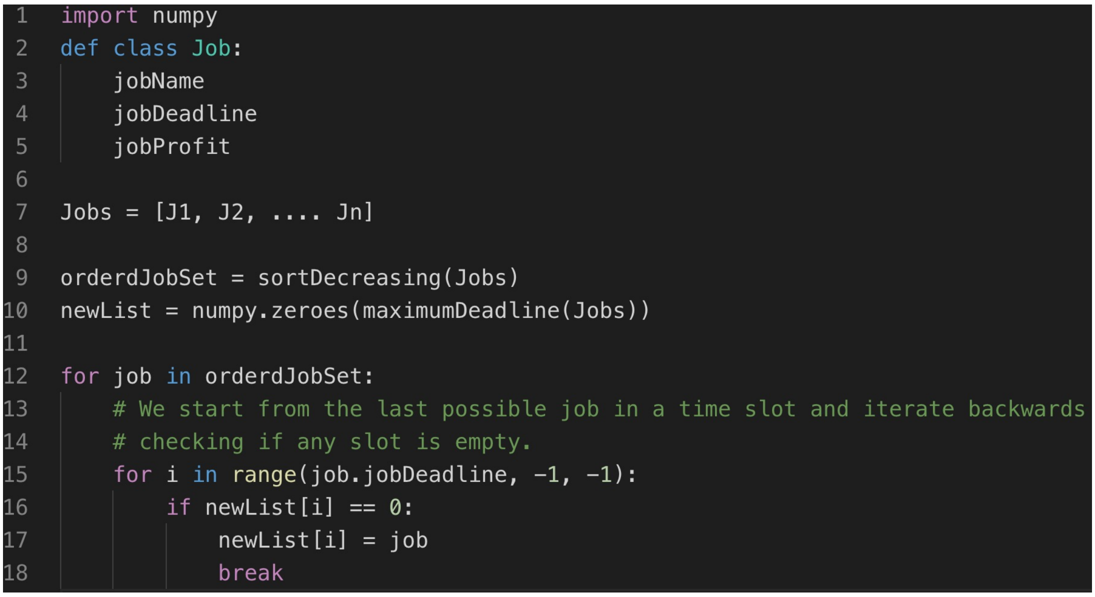
COMP3121 Homework Q4

Arth Sanskar Patel
z5228942

July 16, 2020

1 Answer a

Here in this problem, we have a set $J = [J_1, J_2, \ldots, J_n]$ of n jobs to be completed with each given a deadline d_i and profit p_i. Each job takes 1 unit of time to be completed and we have to select jobs in such a manner that our profit is maximised. First we sort the jobs in decreasing order of maximum profit. We initialise a newList with k 0's where k represents the last deadline. For example if the highest deadline out of the given jobs is 6 then the list would be $[0, 0, 0, 0, 0, 0]$. We now traverse our orderedJobSet and for each job J_i, we find a time slot x where $x < deadline(J_i)$ and x is greatest. If its 0 then its free and we assign that job to the slot where deadline is greatest. If we get a job for which no slot x is free before its deadline, we simply ignore it and continue. This way the newList we get will contain jobs which will maximise the profit. Pseudo code for this is given below



```
import numpy
def class Job:
    jobName
    jobDeadline
    jobProfit

Jobs = [J1, J2, .... Jn]

orderdJobSet = sortDecreasing(Jobs)
newList = numpy.zeroes(maximumDeadline(Jobs))

for job in orderdJobSet:
    # We start from the last possible job in a time slot and iterate backwards
    # checking if any slot is empty.
    for i in range(job.jobDeadline, -1, -1):
        if newList[i] == 0:
            newList[i] = job
            break
```

Since we are iterating through the jobs once and for each job we are iterating through the newList, the time complexity of the algorithm will be $O(n^2)$

We know this solution is optimal because we are selecting jobs in decreasing order of profit. So we are maximising the profit in each step when we select the highest possible profit job available. Lets say we have jobs J_x and J_y each with a same deadline d but profits $p_x > p_y$. In our ordered list J_x will be before J_y and hence we will assign the one available slot to J_x and then when we process J_y, we will see that the slot is occupied and if the slots before deadline d are occupied, they must have been occupied by jobs with profit which is higher than p_y as we are traversing the orderedJobSet. So ignoring job J_y will not matter as choosing it in place of J_x would've only decreased our profit. Reference to this question was taken from here.