INTRIDUCTION TO ALGORITHM DESIGN AND ANALYSIS

HOMEWORK 5 REPORT

PART 1

An optimal algorithm to schedule the jobs in decreasing order of w_i/t_i . So, jobs are sorted by decreasing order w_i/t_i . In this algorithm selection sort is used. Suppose that n is number of jobs, time complexity is $O(n^2)$.

PART 2

a) Suppose that M = 20, {N1,N2,N3} = {2, 5, 3} and {S1,S2,S3} = {30,4,50}. Then the optimal plan would be [NY,NY,NY] and cost of this plan is 2 + 5 + 3 = 10, while this greedy algorithm would return [NY, SF, NY] and cost of this plan is = 2 + 20 + 4 + 20 + 3 = 49. So, given algorith is not optimal.

b) The optimal plan either ends in NY, or in SF. If it ends in NY, it will pay N_n plus one of the following two equaitons.

- The cost of the optimal plan on n-1 months, ending in NY, or
- The cost of the optimal on n - 1 months, ending in SF.

Thus, if $OPT_N(j)$ denotes the minimum cost of a plan on months 1,…,j ending in NY, and

$OPT_S(j)$ denotes the minimum cost of a plan on months 1,…,j ending in SF, then

$$OPT_n(n) = N_n + min(OPT_N(n-1), M + OPT_S(n-1))$$

$$OPT_S(n) = S_n + min(OPT_s(n-1), M + OPT_N(n-1))$$

This algorithm is implemeneted in python. The algorithm has n iterations, and each takes constant time. Thus the running time is O(n).