Problem 1

HIGHEST-PENALTY-FIRST (n, p1, p2, …, pn , d1, d2, …, dn)

SORT jobs by highest penalties and renumber so that p1 ≥ p2 ≥ … ≥ pn.

t[n] = {0} Make n slots of 1 minute each of t such that t[4] starts at minute 4 and end at minute 5

FOR j = 1 TO n

    FOR k=1 to n

        IF dj-k <= 0

            Assign job j to interval [dj, dj+1].

            T[dj] ← pj

        IF t[dj-k] is empty

            Assign job j to interval [dj-k, dj].

            T[dj-1] ← pj

RETURN t

Prove of optimal solution:

Proving by induction that highest penalty first schedule S is optimal

Let S* to be an optimal schedule with same choices as S up till job i

Base case: P(0) is true

Assume P(i-1). We have to prove P(i) is also true

This means that the i-1 choices that greedy made are all optimal because they match with the i-1 jobs that are in optimal solution.

Now either the greedy schedules the job i or not

If it did not schedule the job then there must be no place before deadline and if it has same schedule up till i-1 jobs with optimal solution so optimal solution also cant schedule the job either so both optimal and greedy ends up paying the penalty.

So greedy is as good as optimal