Subject
Month: Year Date

Problem #4

1. Activity Selection : Dp vs Greedy

a. optimal substructure?

yes. if ~~ak~~ Ak is in an~~y~~ optimal soulution for Ŝ-ap, then the subproblems c[a,k] and c[k,p] must also be optimal (cutting an activity from an optimal solution leaves an optimal sub-solution.

b. RecuSelect (s, f, k, n)

```
RecuSelect (s, f, a, p):
    if Ŝ-ap = ∅ : return []
    best = []
    for each ak in Ŝ-ap:
    left = RecuSelect (s, f, a, k)
    right = RecuSelect (s, f, a, k)
    candidate = left + [ak] + right
    if len (candidate) > len (~~left + [ak] + right~~ best) = candidate
return best.
```

C. Bottom-Up Tabulation

```
for lenght l = 2 to n+1:
    for all pairs (i,j) with j-i =l:
        c[i][j] = max over ak in
    Ŝ-ij of (c[i][k] + c[k)[j]+1)
        if no such ak: c[i][j] = 0

Answer = c[0][n]
```

Greedy approach

→ always pick the activity with the earlier finish time that is compitable with the last selected activity

```
        GreedySchedule (s, f, n):
→          sort activities by finish time f
           A = {a1}
           last = 1
               for i = 2 to n:
                   if s[i] >= f[last]:
                       A = A ∪ {ai}
                       last = i
                   return A
```