## Activity Selection Problem:

```
ASP(A,n)
{
  Sort according to early finish times.
    B= { A[i] }
    for(i<-2 To n)
   {
      if(Si >= fj)
     { B = B U { A[i] }
        j=i }
   }
}
```

To prove any greedy algo correct, we've to prove 2 things:

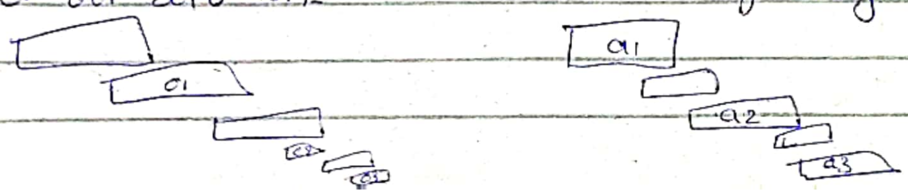
1) Greedy choice property
2) Optimal substructure

Proof (Greedy choice property):

Let there is an optimal solution $O\{O_1, O_2, O_3\}$

& greedy solution is $G_S = \{a_1, a_2, a_3\}$

Let our solution:  But in greedy:



Case 1:

If $(o_1 == a_1)$

we are done (we've same solution)

Case 2:

if $(o_1 \neq a_1)$

Just replace $o_1$ by $a_1$

$O = \{a_1, o_2, o_3\}$

- Solution is feasible (all conditions satisfied)

So, no cost is used/added so greedy choice made by our algo is correct.

# Activity Selection Problem:

## Proof of Correctness:

Optimal substructure proof,

Proof by induction:

Let there are $n$ activities,

$S = \{a_1, a_2, a_3, \dots, a_n\}$

assume it is sorted by early finish time.

If we remove $a_1$ then

For $n-1$ activities

$S' = \{a_2, a_3, \dots, a_n\}$

Let there be an optimal solution for $S'$ (where activities overlapping with $a_1$ should not be part of solution)

After adding $a_1$ to that optimal solution

Now, we'll have optimal solution for $n$ activities.

Proof by contradiction:

Let there are $n$ activities

Let there is an optimal solution that $K$ activities have been scheduled.

If we remove '$a_1$'

Now activities are $n-1$

& their optimal solution consists '$K-1$' activities

Contradiction:

Let us assume that we've another optimal solution for subproblem (say K') better than 'K-1' so

$$K' > K-1$$

After adding $a_1$ to go towards complete problem.

$$K'+1 > K-\cancel{1}+\cancel{1}$$
$$K'+1 > K$$

It contradicts our assumption so not possible.

# Fractional knapSack:

Problem:-

→ There are n items.
→ Each item has some value $v_i$ & weight $w_i$
→ knapSack (like a bag) has capacity 'W'
→ You are allowed to add any item or skip
→ You can add fraction of item as well

∴ In binary knapSack, full item is added rather than an option to add some part of it

Goal:
To maximize value sum.
Here, relation is value sum ∝ value
& value sum $\propto \frac{1}{\text{weight}}$

| | A | B | C |
|---|---|---|---|
| Value | Rs. 100 | Rs. 500 | Rs. 1000 |
| weight | 3 pounds | 4 pounds | 10 pounds |

W=6

$\frac{\text{value}}{\text{pound}} = 100$

If value is choice then 600 added
If weight then A added & B partial i.e. 100 + 375 added i.e. 475

If we take both in choice as $\frac{\text{value}}{\text{weight}}$

| | A | B | C |
|---|---|---|---|
| $\frac{\text{value}}{\text{weight}}$ | 33.3 | 125 | 100 |

500 (4 of B) + 200 (2 of C added)
= 700

$\frac{W}{W_i}$

Proof by Contradiction [redacted]:

Optim[redacted] Sub[redacted]cture:

[redacted] there[redacted]are $n$ it[redacted]s:

[redacted]et th[redacted] is opti[redacted] solution that

(V) [redacted] value S[redacted]

[redacted] we [redacted]e $V_1$ (v[redacted] of first) [redacted] $W-w_1$

Now it[redacted]s $V-1$

& thei[redacted]optimal s[redacted]tion [redacted]onsists of [redacted] $V_1$

Let us ass[redacted]e th[redacted]t we [redacted]her optimal

Solution [redacted] S[redacted]roblem [redacted]ay [redacted]$V'$) ~~so~~

better [redacted]an [redacted]$V-V_1$

$V$[redacted]$V-V_1$

$V'+V_1 >$[redacted]$-V_1+$[redacted] (we add $V_1$)

$V'+V$[redacted]$>V$