Kunal Bandooni
CE
2014413

TUTORIAL-7

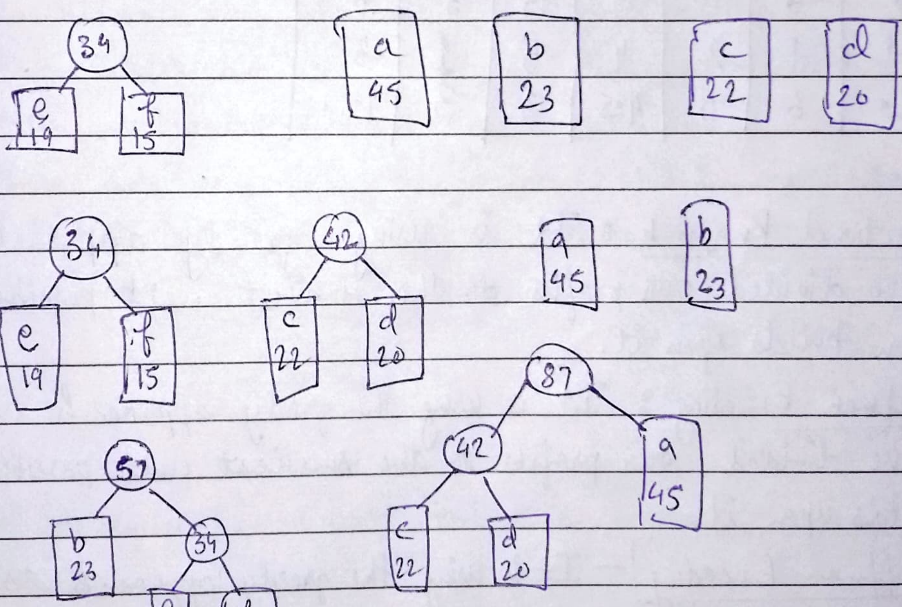
Sol1. It is an algorithmic paradigm that builds up a solution by adjoining smaller pieces together, always choosing the next piece that offers the most obvious & immediate benefit.

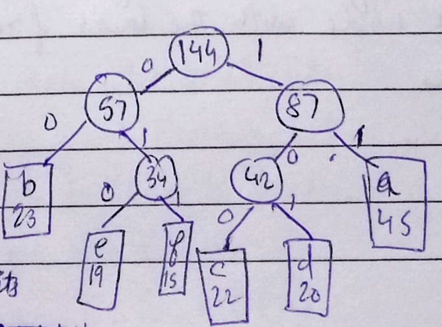
We should use greedy approach whenever a locally optimal solution is also globally optimal.

Sol2.

Name	TC	SC
Activity Selection	O(nlogn) – O(n)	O(n)
Job Sequencing	O(n²) – O(nlogn)	O(n)
Fractional Knapsack	O(nlogn) – O(n)	O(n)
Huffman Encoding	O(nlogn) – O(logn)	O(n)

Sol3. a=45 , d=20 , b=23 , e=19 , c=22, f=15





Total bits = (45×2)+(23×2)
+(22×3)+(20×3)+
(19×3)+(15×3)=364 bits

a = 11
b = 00
c = 100
~~d = 101~~
e = 010
f = 011

~~Total bits~~=

Sol 4. A 2-tree is used to implement Huffman encoding algorithm. It is a binary tree where every node has either 2-child or no child.

- Applications of Huffman Encoding –
- – Data compression in long files without any lass.
- – To implement traffic routes with traffic magnitude.

Sol 5.

v	10	5	15	7	6	18	3
w	2	3	5	7	1	4	1
v/w	5	1.67	3	1	6	4.5	3

$k = 15-1-2-4-5-1-2=0$

Profit $= 30+10+18+15+3+3.34$
$=79.34$

v	6	10	18	15	3	5
w	1	2	4	5	1	3
v/w	6	5	4.5	3	3	1.67

Sol 6. Fractional Knapsack : It is using a greedy approach as we have divided our profits to the smallest unit possible & then builds upon it.

Huffman Encoding : It is using the greedy approach as we have divided our profits to the smallest unit possible & then builds upon it.

Huffman Encoding – It is using the greedy approach as it always places the node with the lower frequency further from the parent node.

Sol7.

	↓			↓	↓	↓
Start	1	2	0	6	9	10
End	3	5	7	8	11	12
Index	0	1	2	3	4	5

Jobs To Do ⇒ [0], [3], [4] or [5]

i.e. ⇒ Max = 4

Sol8.

	Profit	Deadline
a	20	2
b	15	2
c	10	1
d	5	3
e	1	3

0	1	2
b	a	d

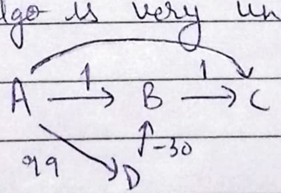
0 1 2 3

Profit = 20 + 15 + 5
= 40

Sol9. Times when not to use greedy algorithm :-

- When approach involves a lot of assumptions, such as "pick always the..."
- When we need complex implementation.
- When we are making performance - critical application.

eg - Dijkstra's algo is very unoptimised for graphs with negative edges



Here, wes cannat find distance of pair [A, C] → It gives 0, though it is -200.

Sol10. TC of job sequencing = $O(n^2)$, but we can improve it using priority queue by using (Max Heap).

Algorithm :-

1. Sort based on deadlines
2. Iterate the end & calculate the available slots b/w two consecutive deadline
   Put everything in Max Heap.
3. If slots available & there are jobs in MaxHeap, include Job ID with max profit & deadlines in the result.
4. Sort the array based on deadlines.

TC = $O(n \log n)$; SC = $O(n)$