Page No.:
Date: / /

Dhiraj Bodake
18141216

Tutorial No-5

Q.1) suppose you have 6 containers whose weight are 50, 10, 30, 20, 60, 5 & ship whose capacity is 100. Find an optimal solution to this instance of container loading problem.

→ In container loading problem, all container one of same size with constrain $\sum_{i=1}^{m} w_i x_i \leq C$

where $x_i = 0$ or 1 (0 if not included
1 if included)

w_i = weight of container
C = capacity

- To obtain optimal solution, we sort the given weight in increasing order.

weights = { 5, 10, 20, 30, 50, 60 }

$w_6 \quad w_2 \quad w_4 \quad w_3 \quad w_1 \quad w_5$

stage 1 :- we include container 6
$5 \times 1 \leq 100$
solution set = { 0, 0, 0, 0, 0, 1 }

stage 2 :- we include container 2
$5 + (10 \times 1) \leq 100$

solution set = { 0, 1, 0, 0, 0, 1 }

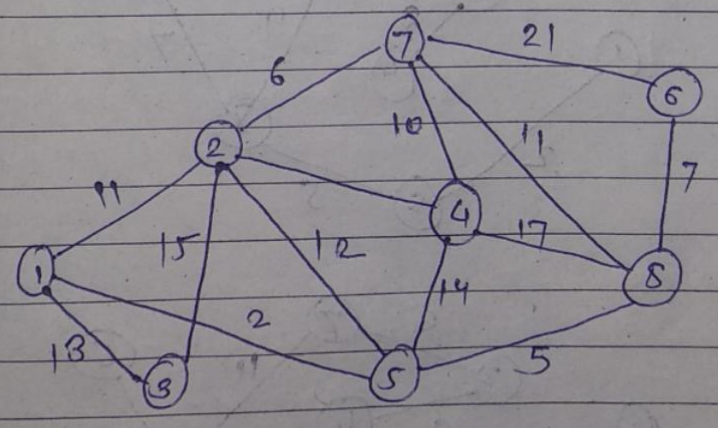
Stage 3 :- we include container 4,
$15 + (20 \times 1) \le 100$
solution set = {0, 1, 0, 1, 0, 1}

Stage 4 :- we include container 3
$35 + (30 \times 1) \le 100$
solution set = {0, 1, 1, 1, 0, 1}

Stage 5 :- as weight of next container that we can include is 50. since 65+50 is not less than 100, so it ~~can~~ not get included.

so maximum no. of container that are get loaded is 4. with weight 65.
containers → {2, 3, 4, 6}

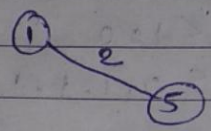
Q.2) compute a minimum cost spanning tree for the graph using (a) prim's algorithm
(b) kruskal's algorithm.



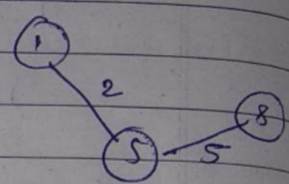
→ (a) prim's algorithm :-

- select a minimum cost edge from graph then select a minimum cost edge from graph which is connected to already selected vertex.

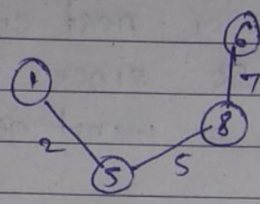
step ①



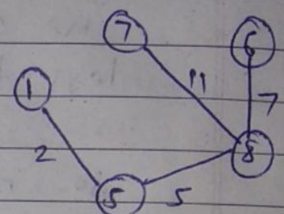
step ②



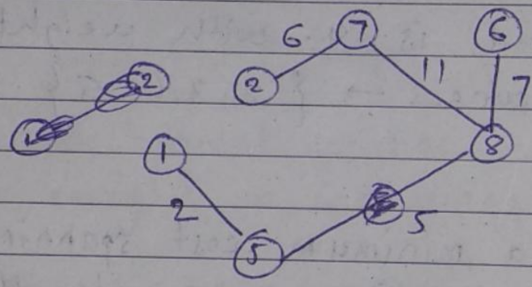
step ③



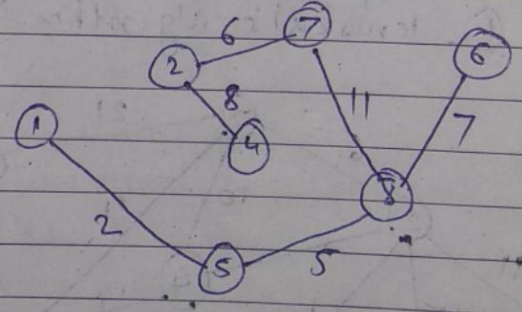
step ④



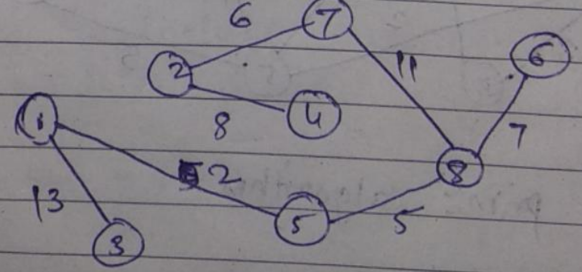
step ⑤



step ⑥



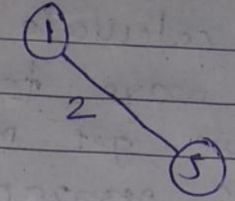
step ⑦

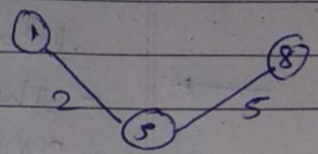


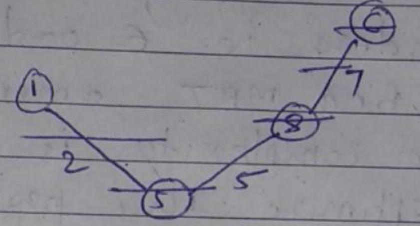
minimum cost = ~~55~~ 52

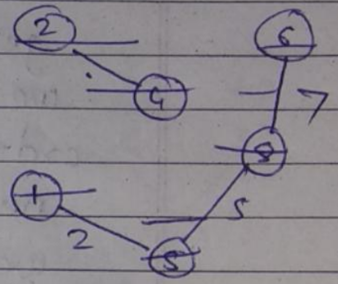
② Kruskal's Algorithm:-

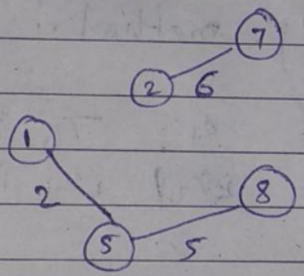
- alway select minimum cost edge, but if it form cycle don't considered it. discard it.

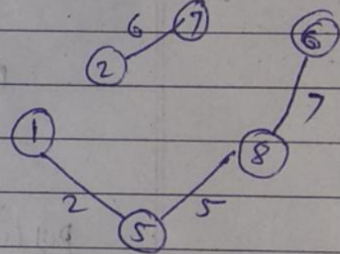
step ① 

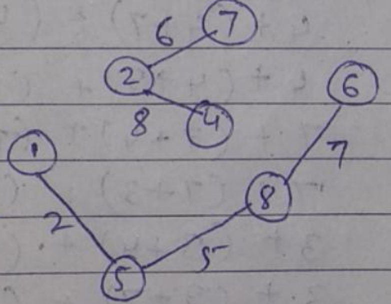
step ② 

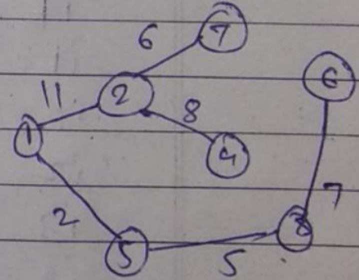
~~step ③~~ 

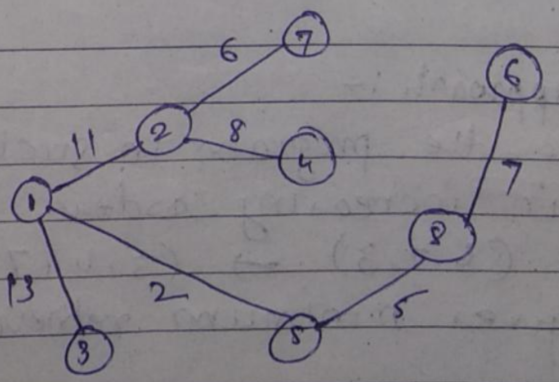
~~step ④~~ 

step ③ 

step ④ 

step ⑤ 

step ⑥ 

step ⑦ 

minimum cost = 52

Q.3) consider the following data and find a permutation data that result in optimal silution. (minimum MRT)

program	i1	i2	i3
length	4	7	3

→ select program & calculate the MRT.

- There are no. of ways to arrange programs and from this we get MRT.
- Here we have 3 programs then we get 3! orders i.e. 6 orders and we need to find MRT among these orders. Time complexity is high so we need to optimize this problem. so we use greedy method.

$n = 3 \quad l = 4, 7, 3$

$p = i1 \; i2 \; i3$

possible orders	MRT
i1 i2 i3	$4 + (4+7) + (4+7+3) = 29$
i1 i3 i2	$4 + (4+3) + (4+3+7) = 25$
i2 i1 i3	$7 + (7+4) + (7+4+3) = 32$
i2 i3 i1	$7 + (7+3) + (7+3+4) = 31$
i3 i1 i2	$3 + (3+4) + (3+4+7) = \boxed{24}$
i3 i2 i1	$3 + (3+7) + (3+7+4) = 27$

Greedy approach:-

Arrange the program in such order that length it is in increasing order

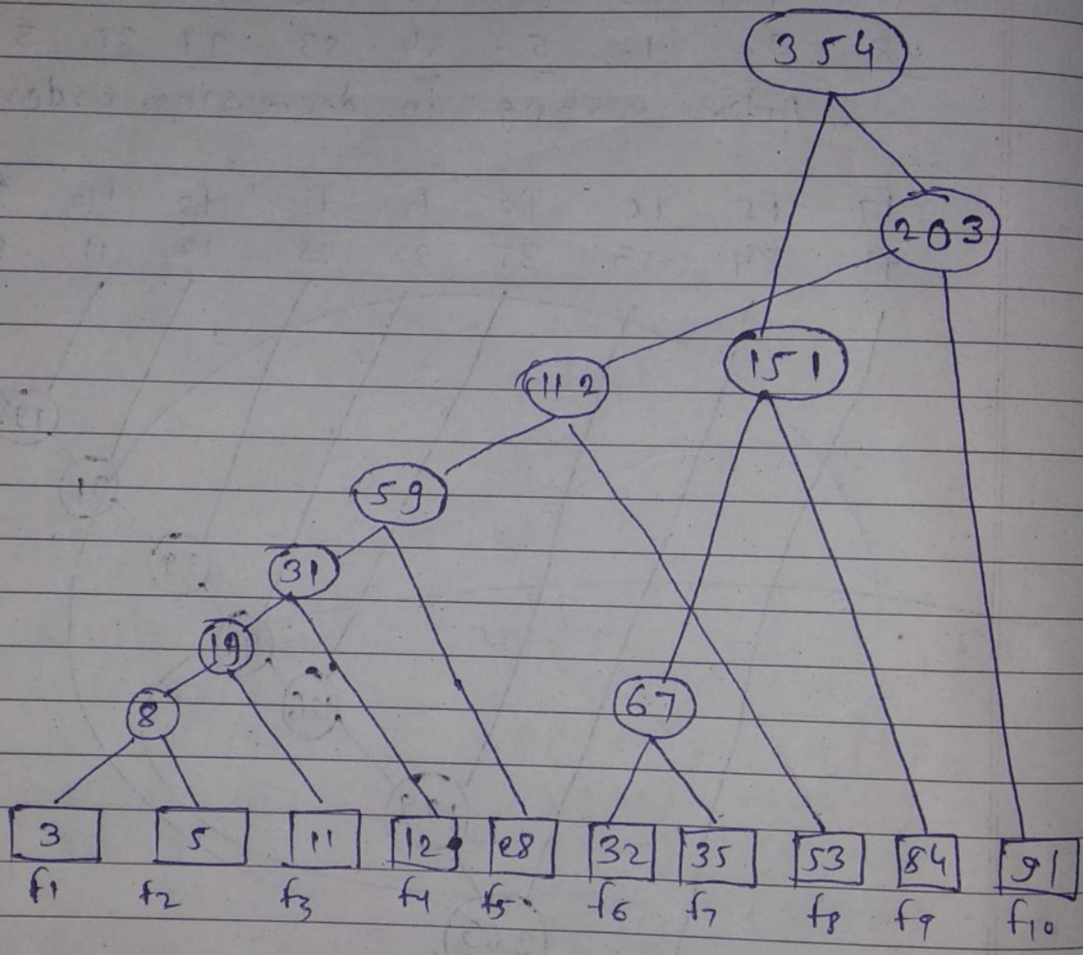
eg - $(4,7,3) \rightarrow (3,4,7)$

this gives minimum retrieving time = 24

GoodLuck Page No:
Date

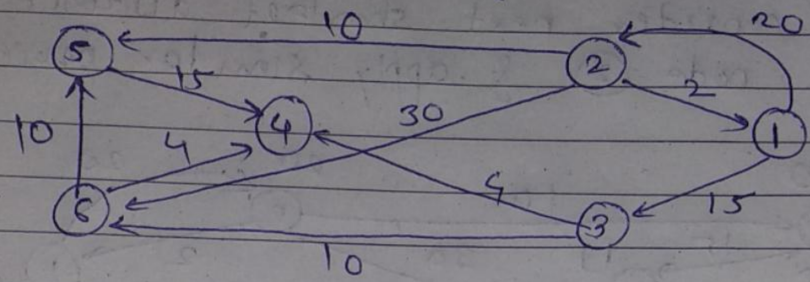
Q.4) find an optimal binary merge pattern for ten files whose length are
28, 32, 12, 5, 84, 53, 91, 35, 3 & 11

— Here we use greedy method
- Always choose 2 files with lowest/least length.



Total optimal cost
= 8 + 19 + 31 + 59 + 67 + 112 + 151 + 203 + 354
= 1004

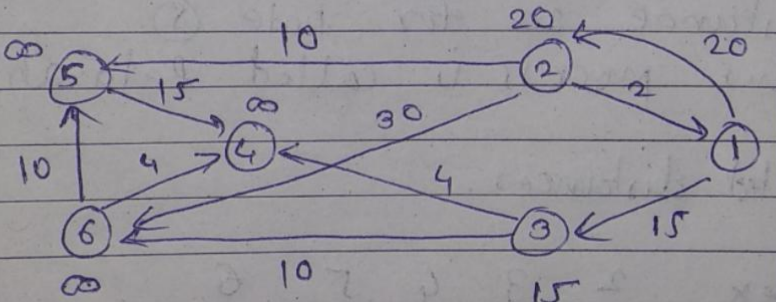
Q.5) use algo. shortest path to obtain in non decreasing order the lengths of the shortest paths from vertex 1 to all remaining vertices in the diagram.



→ Here we use Dijkstra's algorithm
- Here we can apply Relaxation ie.

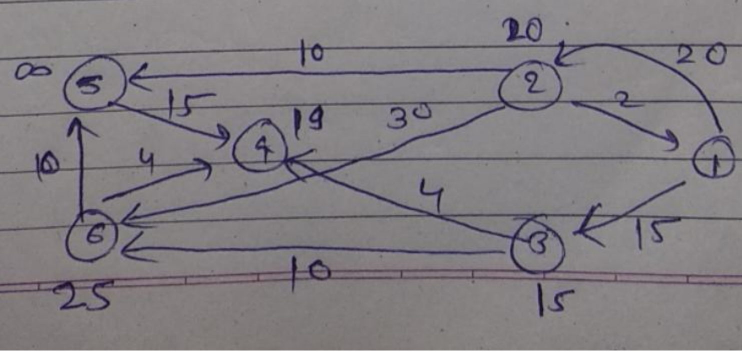
$$\text{if } (d[u] + c(u,v) < d[v])$$
$$d[v] = d[u] + c[u,v]$$

source node 1.
from this only node 2 & 3 is connected directly so vertices other than these are set to at ∞ distance.



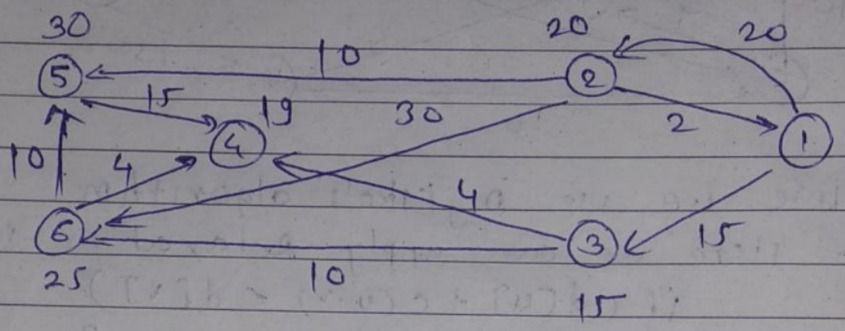
v	2	3
d[v]	20	15

now we consider node 3 among node 2 & 3 as it's distance is lowest & apply similar process.



v	2	3	4	6
d[v]	20	15	19	25

now next node with lowest distance from source node ① is node ④ but from node ④ there is no any outgoing edge, so we consider next shortest distance node i.e. node ② & apply similar process.



As to reach node ⑤ we have to path one from node ③ to ⑥ and then ⑥ to ⑤ it ~~is~~ cost 35.
whereas from node ③ to nod ⑤ it cost 30. so we choose shortest path distance 30 for node ⑤.
This process is called Relaxation.

shortest distances:

Vertex	2	3	4	5	6
distance	20	15	19	30	25