when l

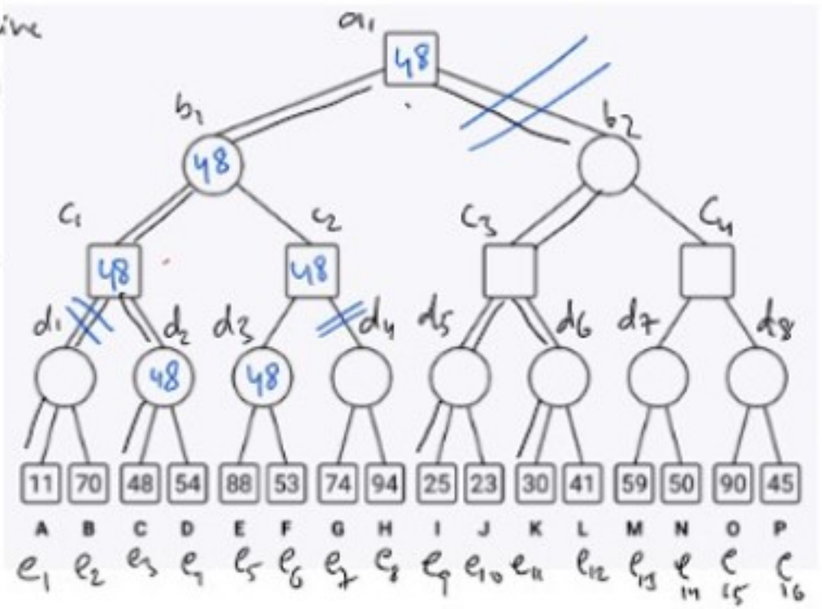
t_l allchildren t_l left child

if horizon is live

t_s min(e_1, eval)

when S

t_s P t_l younger sibling t_s P



Initial Clusters Horizon nodes : A, C, I, K

SSS* Solved status for Horizon nodes : A, C, I, K, D, E, F

0	e_3 S 48	e_{11} S 30	e_9 S 25	e_1 S 11
1	e_4 l 48	e_{11} S 30	e_9 S 25	e_1 S 11
2	e_4 S 48	e_{11} S 30	e_9 S 25	e_1 S 11
3	d_2 S 48	e_{11} S 30	e_9 S 25	e_1 S 11
4	c_1 S 48	e_{11} S 30	e_9 S 25	
5	c_2 l 48	e_{11} S 30	e_9 S 25	
6	d_3 l 48	d_4 l 48	e_{11} S 30	e_9 S 25
7	e_5 l 48	d_4 l 48	e_{11} S 30	e_9 S 25
8	e_5 S 48	d_4 l 48	e_{11} S 30	e_9 S 25
9	e_6 l 48	d_4 l 48	e_{11} S 30	e_9 S 25
10	e_6 S 48	d_4 l 48	e_{11} S 30	e_9 S 25
11	d_3 S 48	d_4 l 48	e_{11} S 30	e_9 S 25
12	c_2 S 48	e_{11} S 30	e_9 S 25	
13	b_1 S 48	e_{11} S 30	e_9 S 25	
14	a_1 S 48			
15	[ ]			

i) we will make a priority queue first

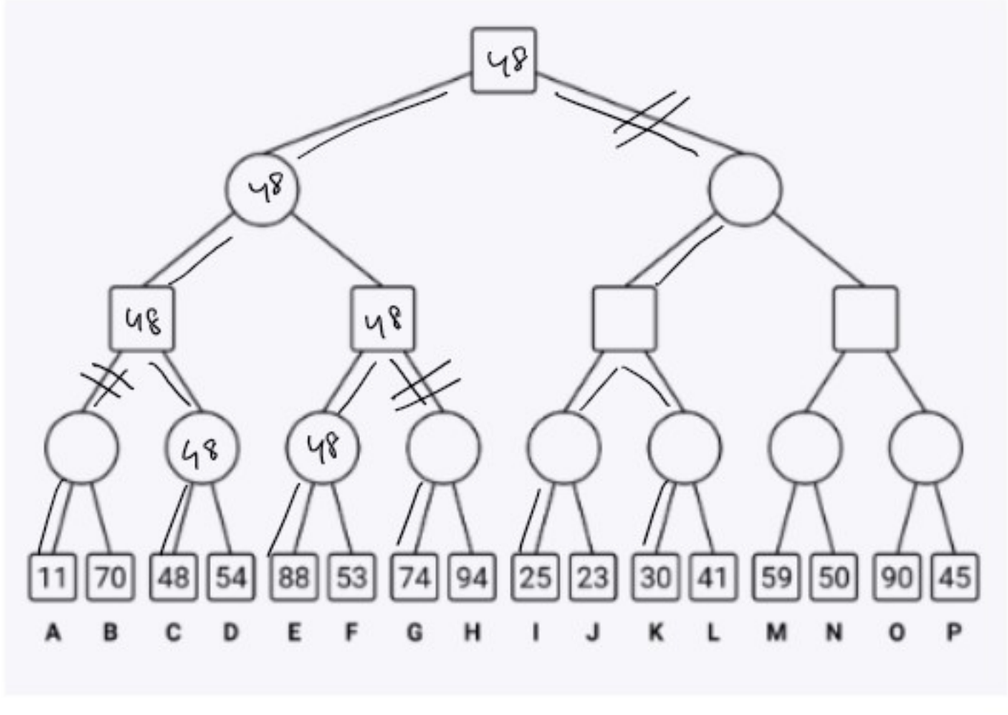
ii) we will add initial nodes with cost in descending order

iii) if we are at max node we will solve its adjacent sibling

iv) if there are no more max node siblings we will mark it's parent min node Solved

v) if we are at min node we will directly pass the value to parent node we will not solve min node adjacent sibling

ACIK



ASII CS~~48~~ 1525 KS30

DS~~54~~

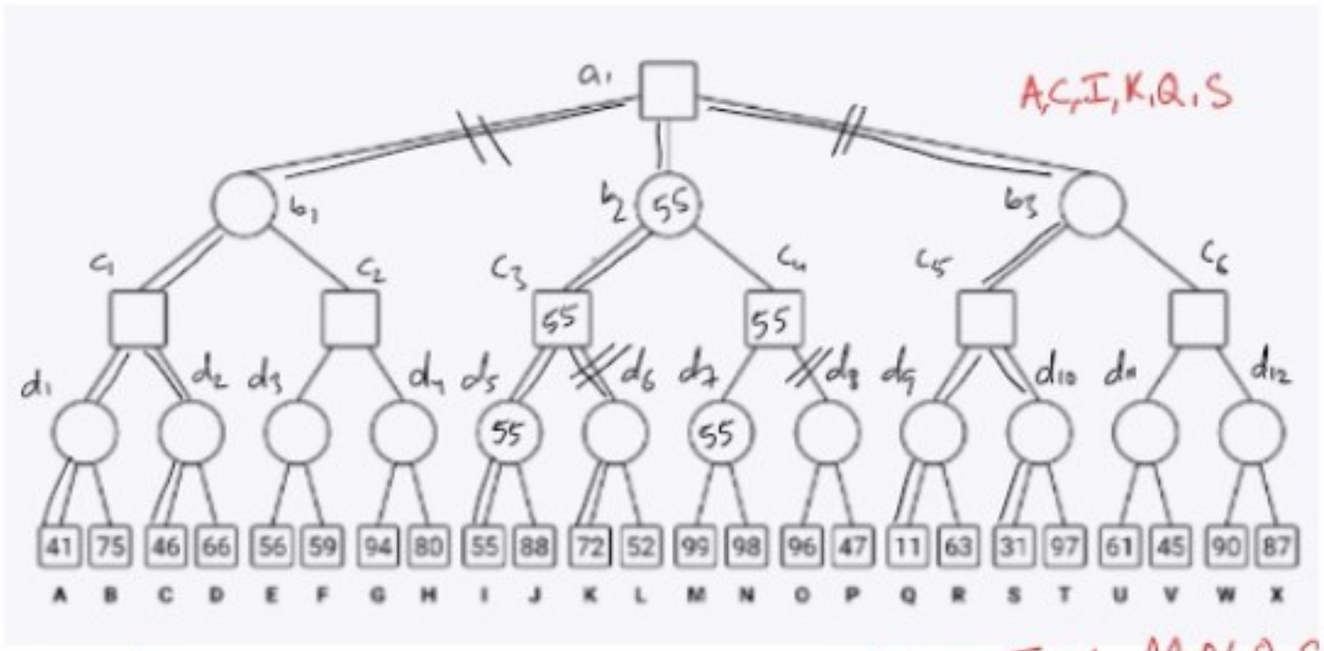
E~~188~~ ~~G174~~

ES48

FS48

Solved

D, E, F



A,C,I,J,K,L,M,N,Q,S

when L

t_L All child

t_L left child

if N is Horizon

t_S min(n, eval)

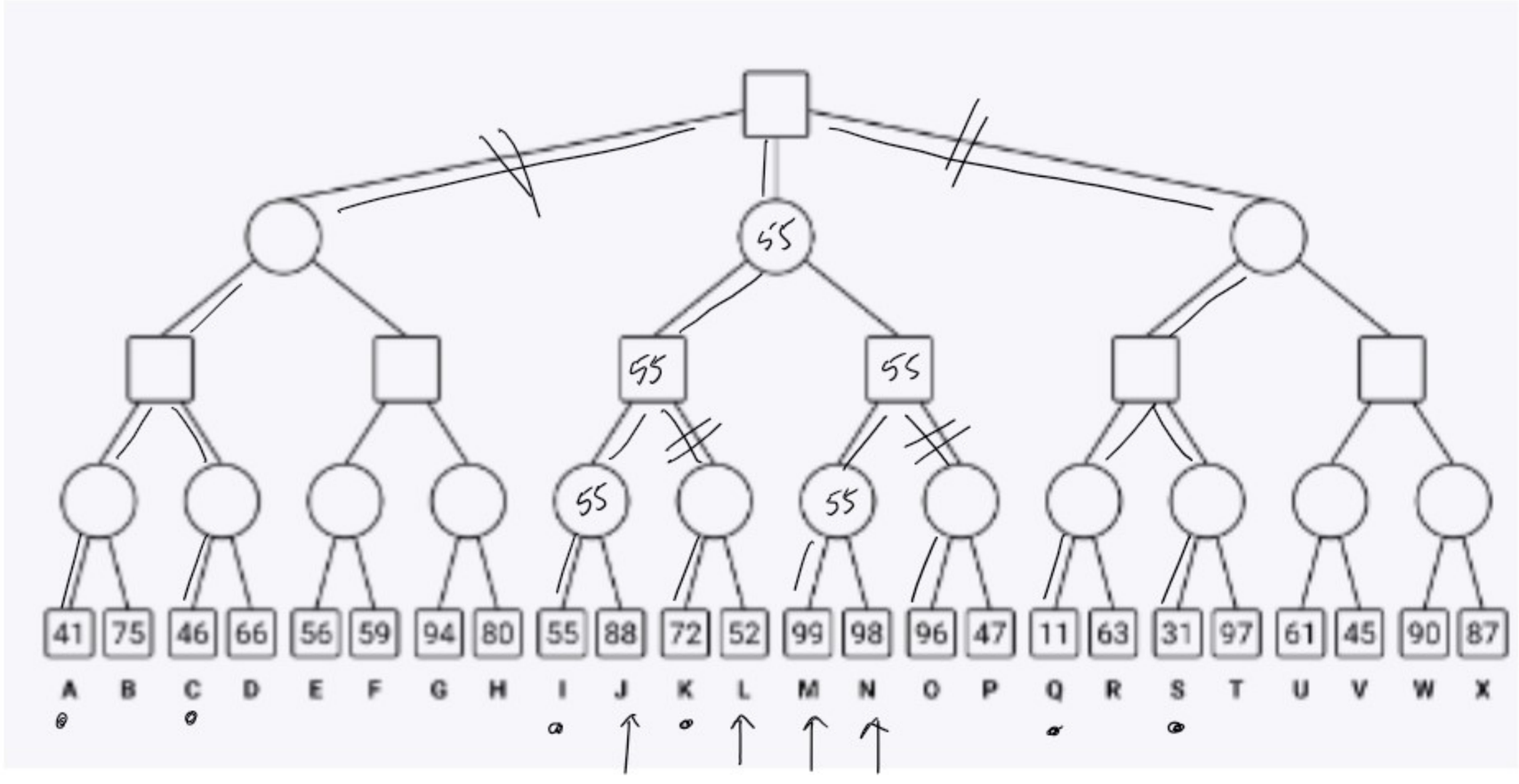
when S

t_S P

t_L next Sibling

t_S P & prune everything

0	KS72	IS55	CS46	AS41	SS31	QS11
1	LL72	"				
2	IS55	LS52	CS46	AS41	SS31	QS11
3	JL55					
4	JS55	LS52	CS46	AS41	SS31	QS11
5	d_5S55	LS52	CS46	AS41	SS31	QS11
6	C_3S55	CS46	AS41	SS31	QS11	
7	C_4L55					
8	d_7L55	d_8L55	CS46	AS41	SS31	QS11
9	ML55					
10	MS55	d_8L55	CS46	AS41	SS31	QS11
11	NL55					
12	NS55	d_8L55	CS46	AS41	SS31	QS11
13	d_7S55	d_8L55	CS46	AS41	SS31	QS11
14	C_4S55	CS46	AS41	SS31	QS11	
15	b_2S55	CS46	AS41	SS31	QS11	
16	a_1S55					
17	[ ]					



A541, C546, I555, K572, Q511, S531

R552

J555

M599 O596

M555 N598

N555