

Name: Jonas de Deus Guterres

SID: 1301183615

Class: IF-42-INT

Date 27th April 2020
Name: Jonas de Deus Guterres
Class: IF-42-INT

RESUME

1 - Branch and Bound

1.1 Job sequencing

Jobs = {J1, J2, J3, J4}
P = {10, 5, 8, 3}
d = {1, 2, 1, 2}
S1 = {J1, J4}
S2 = {1, 0, 0, 1}

Branch and Bound is almost same with the backtracking algorithm but the difference is, backtracking using first depth search and branch and bound using Breadth first search. Here is the following illustrate about the tree of Branch and bound:

1st Method.
Using Breadth first Search.

∴ Therefore it finish in 16 nodes in J4 and J4.

No. _____
Date: _____

2nd Method.
Using stack approach.

5 ∴ we follow the
4 J1 in the bottom
3 and J4 in the
2 top of stack

→ So the next job I am going to do next is 5 because it is the top of the stack, but because node 5 already in the J4, so we pop also node 4 in J3.

6 ∴ we push 6 to
the stack because
3 node six can be
2 Job 4.

→ So the next job of node six already cannot edit anymore, so we pop six and three from the stack.

2 ∴ it remains node 2 in the stack

8
7
2

∴ we push node 8 and 7 to the stack.

Name: Jonas de Deus Guterres

SID: 1301183615

Class: IF-42-INT

After that we pop the top stack 8 and 7

so, we continuously do it until the stack becomes empty. So the difference between 1st method and 2nd methods is, 1st method using queue and 2nd Method using stack.

QUEUE \rightarrow FIFO \rightarrow 1st Method
 STACK \rightarrow LIFO \rightarrow 2nd Method
 Least Cost \rightarrow LC-BB \rightarrow 3rd Method

Now we are going to see the 3rd Method and this method is depend on the cost. so we are going the minimum/Least cost until the end of the process.

LC-BB

so we continuously go the leaf until we got all possible job.

(1.2) Job sequency with Deadline

So remember this problem can also using greedy algorithm and Dynamic algorithm. Because Branch and Bound also looking for minimization, so we can use it.

Job	1	2	3	4
Penalty	5	10	6	3
deadline	1	3	2	1
time	1	2	1	1

Upper = Sum of all penalties except that included in solution

Cost = Sum of all penalties till the last job considere

$U = \sum_{i \in S} P_i$

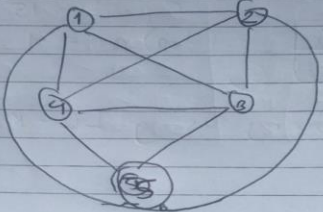
$C = \sum_{i \in f} P_i$

Name: Jonas de Deus Guterres

SID: 1301183615

Class: IF-42-INT

1.4: Traveling Salesperson Branch-n-Bound



	1	2	3	4	5
1		∞	20	30	10
2	15		∞	16	4
3	3	5		∞	2
4	19	6	18		∞
5	16	4	7	16	

→ the first things we have to do is take out the minimum value

	1	2	3	4	5
1		∞	20	30	10
2	15		∞	16	4
3	3	5		∞	2
4	19	6	18		∞
5	16	4	7	16	

minimum value = 1

21 + 4 = 25

Subtract 4

	1	2	3	4	5
1		∞	10	20	0
2	13		∞	14	2
3	1	3		∞	0
4	16	3	15		∞
5	12	0	3	12	

Subtract

	1	2	3	4	5
1		∞	10	17	0
2	12		∞	11	2
3	0	3		∞	0
4	15	3	12		∞
5	11	0	0	12	

10 + 25 + 0
C(1,2) + r + t
C = 35

upper = 28
∴ so will updated the upper when you arrive to the leaf.

we cut the node because the cost higher than 28 so we will not use that leaves anymore

so the minimal cost is 28 with Path = {1, 4, 2, 5, 3}

	1	2	3	4	5
1		∞	∞	∞	∞
2	1		∞	∞	2
3	∞	3		∞	0
4	4	3	∞		∞
5	0	0	∞	12	

	1	2	3	4	5
1		∞	∞	∞	∞
2	∞		∞	11	2
3	0	∞		∞	0
4	15	∞	12		∞
5	11	∞	0	12	

It will same with another nodes (process)

Name: Jonas de Deus Guterres

SID: 1301183615

Class: IF-42-INT

Date: _____

④

	1	2	3	4	5
1	∞	∞	∞	∞	∞
2	12	∞	11	∞	6
3	0	3	∞	∞	2
4	∞	3	12	∞	0
5	11	0	0	∞	∞

⑤

	1	2	3	4	5
1	∞	∞	∞	∞	∞
2	10	∞	9	0	∞
3	0	3	∞	0	∞
4	12	0	9	∞	∞
5	∞	0	0	12	∞

⑥

	1	2	3	4	5
1	∞	∞	∞	∞	∞
2	∞	∞	11	∞	0
3	0	∞	∞	∞	2
4	∞	∞	∞	∞	∞
5	11	∞	∞	∞	∞

⑦

	1	2	3	4	5
1	∞	∞	∞	∞	∞
2	1	∞	∞	∞	0
3	∞	1	∞	∞	0
4	∞	∞	∞	∞	∞
5	0	0	∞	∞	∞

⑧

	1	2	3	4	5
1	∞	∞	∞	∞	∞
2	1	∞	0	∞	∞
3	0	3	∞	∞	∞
4	∞	∞	∞	∞	∞
5	∞	0	0	∞	∞

⑨

	1	2	3	4	5
1	∞	∞	∞	∞	∞
2	∞	∞	∞	∞	∞
3	∞	∞	∞	∞	0
4	∞	∞	∞	∞	∞
5	0	∞	∞	∞	∞

10

	1	2	3	4	5
1	∞	∞	∞	∞	∞
2	∞	∞	∞	∞	∞
3	0	∞	∞	∞	∞
4	∞	∞	∞	∞	∞
5	∞	∞	0	∞	∞