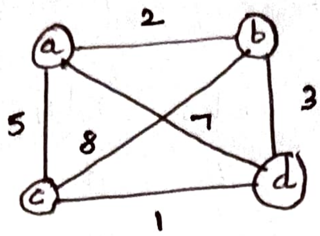
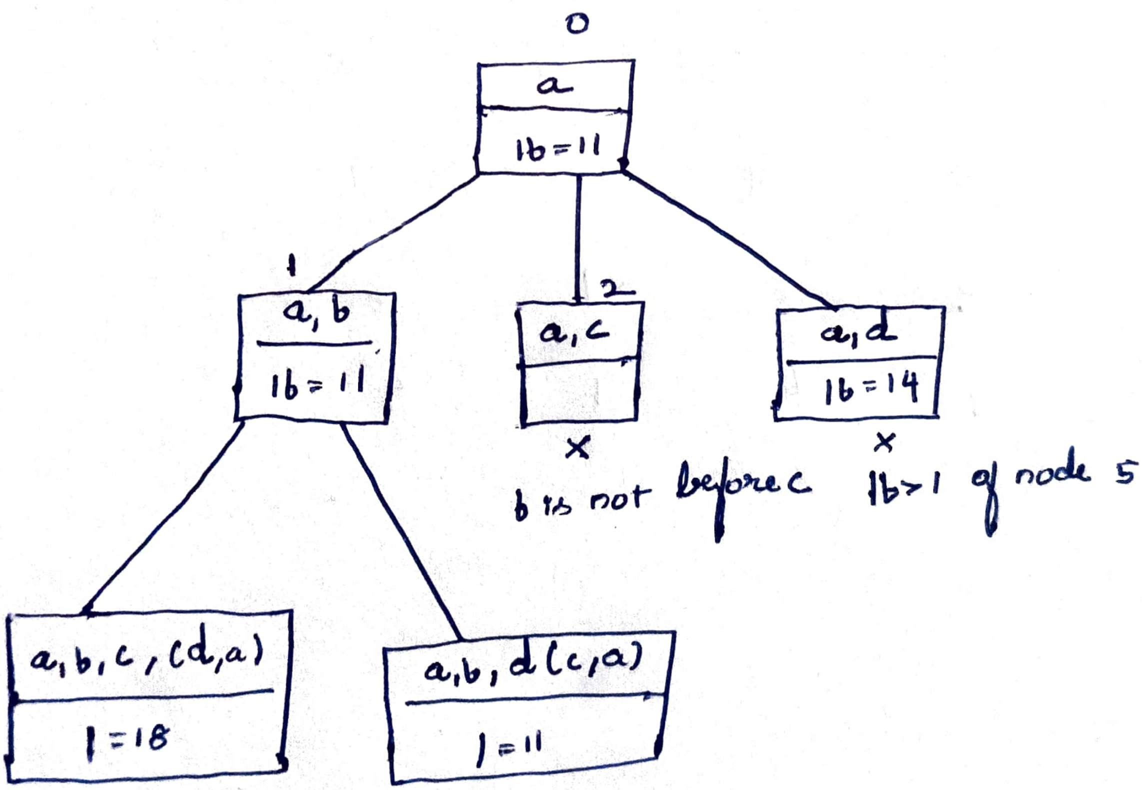
Apply the Branch - and - Bound Algorithm to solve the traveling salesman problem for the following graph.



Without loss of generality, we'll consider a as the starting vertex and ignore the tours in which c is visited before b.

Here is the state-space tree in question.



first tour

better tour
(optimal tour)

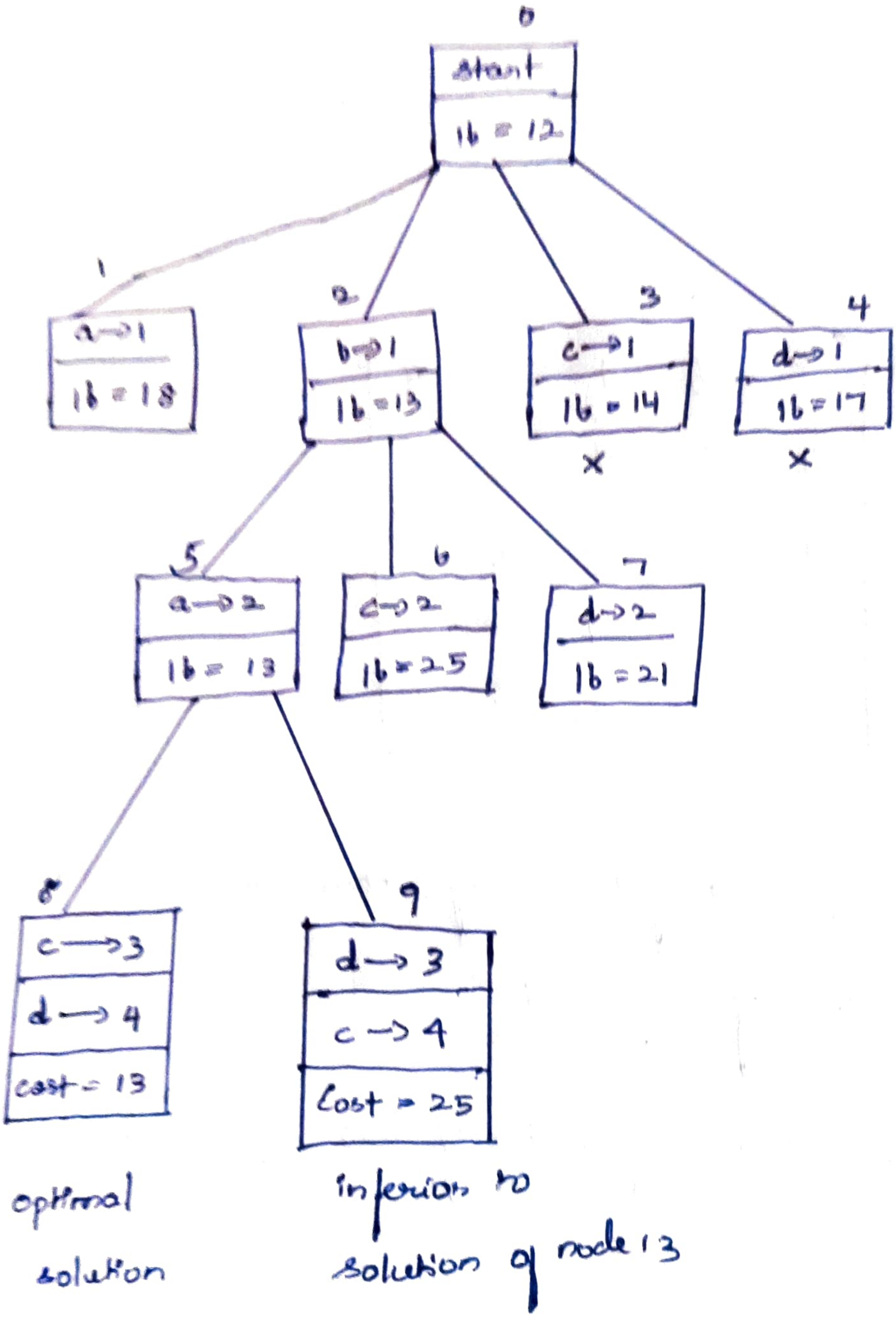
The found optimal tour is a, b, d, c, a of length 11.

2. Solve the same instance of the assignment problem as the on solved by the best-first branch-and-bound algorithm with th Bonding Function based on matrix columns rather than row

	Job1	Job2	Job3	Job4
A	9	2	7	8
B	6	4	3	7
C	5	8	1	8
D	7	6	9	4

$$C = \begin{array}{cccc} \text{Job 1} & \text{Job 2} & \text{Job 3} & \text{Job 4} \\ 9 & 2 & 7 & 8 \\ 6 & 4 & 3 & 7 \\ 5 & 8 & 1 & 8 \\ 7 & 6 & 9 & 4 \end{array}$$

State - space tree



The optimal assignment is b→1, a→2, c→3, d→4

3. a) Apply the nearest neighbour algorithm to the instance defined by the intercity distance matrix below. Start the algorithm at the first city, assuming that the cities are numbered from 1 to 5.

$$\begin{bmatrix} 0 & 14 & 4 & 10 & \infty \\ 14 & 0 & 5 & 8 & 7 \\ 4 & 5 & 0 & 9 & 16 \\ 10 & 8 & 9 & 0 & 32 \\ \infty & 7 & 16 & 32 & 0 \end{bmatrix}$$

The nearest-neighbor algorithm yields the tour 1-3-2-5-4-1 of length 58.

b) Compute the accuracy ratio of this approximate solution.

To compute the accuracy ratio, we'll have to find the length of the optimal tour for the instances given by the distance matrix.

$$\begin{bmatrix} 0 & 14 & 4 & 10 & \infty \\ 14 & 0 & 5 & 8 & 7 \\ 4 & 5 & 0 & 9 & 16 \\ 10 & 8 & 9 & 0 & 32 \\ \infty & 7 & 16 & 32 & 0 \end{bmatrix}$$

Generating all the finite - length tours that start and end at city 1 and visit city 2 before city 3.

1-2-3-5-4-1 of length 77

1-2-4-5-3-1 of length 74

1-2-5-3-4-1 of length 56

1-2-5-4-3-1 of length 66

1-4-2-5-3-1 of length 45

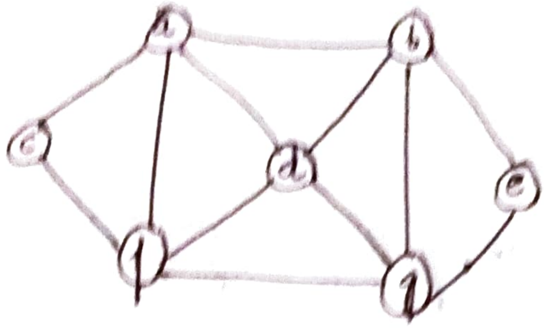
1-4-5-2-3-1 of length 58,

With the tour 1-4-2-5-3-1 of length 45 being optimal. Hence, the accuracy ratio of the approximate solution obtained by the nearest-neighbor algorithm is

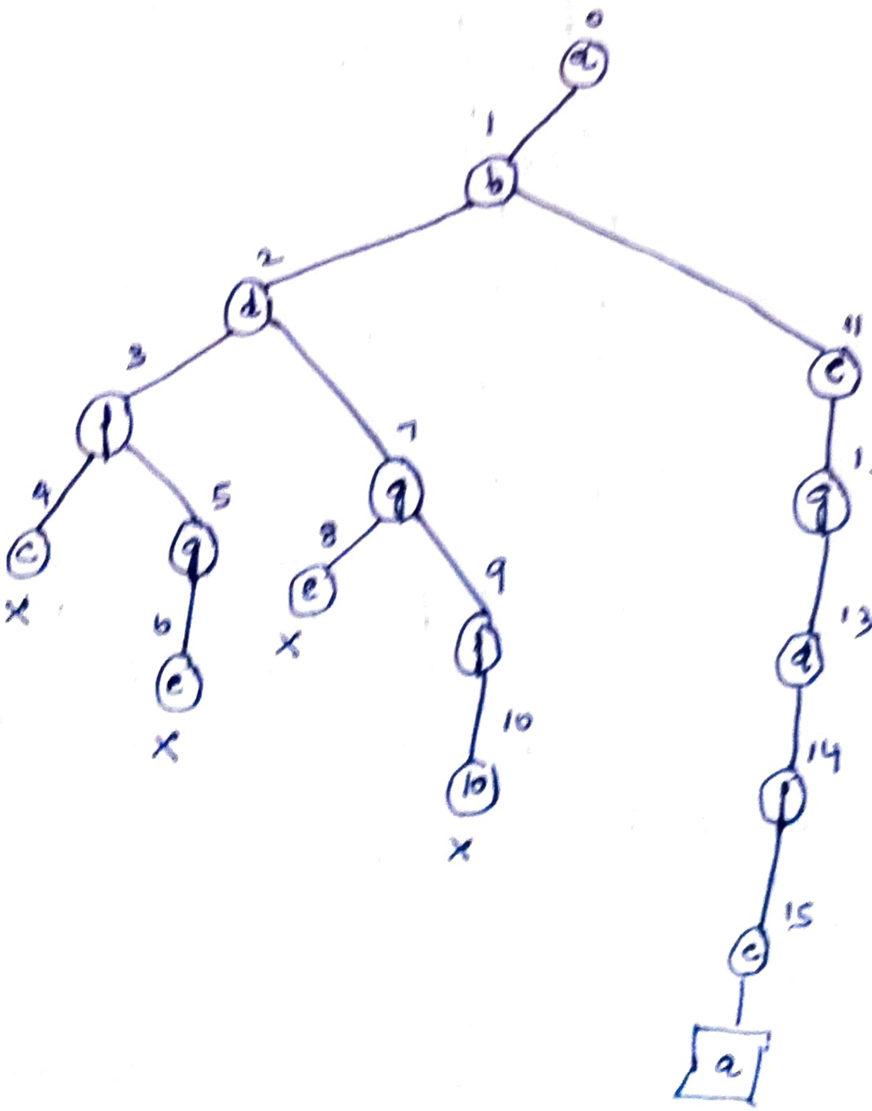
$$r(Sa) = \frac{f(Sa)}{f(S^*)}$$

$$= \frac{58}{45} \approx 1.29.$$

1 Apply backtracking to the problem of finding a Hamiltonian circuit in the following graph.



State - space tree :



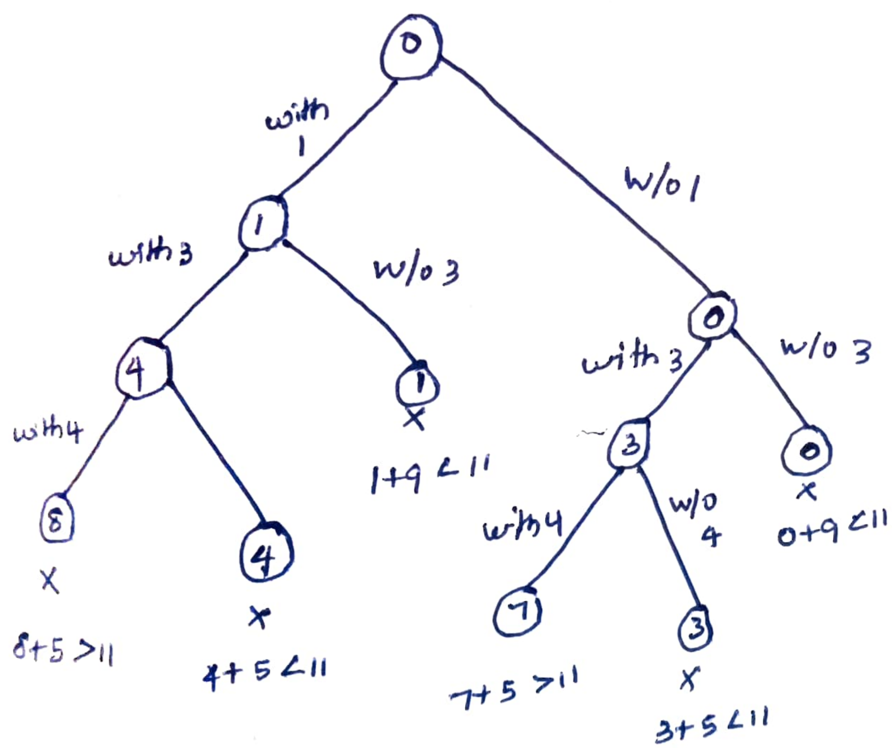
Solution

Apply backtracking to solve the following instance of the subset sum problem : A = { 1, 3, 4 1 5 } and d = 11

State - space tree for the given instance of the subset-sum problem:

S = {1, 3, 4, 5}

d = 11.



There is no solution to this instance of the problem.

6 Will the Backtracking algorithm work correctly if we use just one of the two inequalities to terminate a node as nonpromising?

The algorithm will still work correctly but the state-space tree will contain more nodes than necessary.