Algorithm 7.6: Branch and Bound Algorithm to solve the TSP

(Input) $G = (N, A)$ *complete (di-)graph; cost coefficients* c_{ij} $\forall(i,j) \in A$.

(1) **Initial solution:**
Determine a Hamiltonian tour by some heuristic algorithm;
let z^* *be its objective value (upper bound).*

(2) **Initial relaxation:**
Solve the relaxed problem without the subtour elimination constraints;
let $\bar{z}_1$ *be its objective value (lower bound).*
If $\bar{z}_1 \geq z^*$ *(STOP), the heuristic solution is optimal.*
Otherwise, node P_1 *of the Branch and Bound tree represents the present problem and is the only live node.*

(3) **Branch and Bound procedure:**
Does any live node exist in the solution tree?
If yes: Choose a live node P_k *with the best lower bound, and goto Step (4).*
If no: The best known feasible tour is optimal. If no such tour is known, then there exists no feasible tour.

(4) *Does the solution represented by node* P_k *include subtours?*
If yes: Goto Step (5).
If no: (STOP), the solution in node P_k *is optimal.*

(5) **Branching:**
Let the subtour with the smallest number of arcs not already fixed at 1 be the tour $(1, 2, \ldots, r, 1)$. *Branch from node* P_k *to nodes* $P_{s+1}, P_{s+2}, \ldots, P_{s+r}$, *so that, in addition to the variables fixed earlier, at the node* P_{s+1} *we set* $x_{12} := 0$, *and at each node* P_{s+j}, $j = 2, \ldots, r$, *we set* $x_{j,j+1} := 0$ *(where* $x_{r,r+1} = x_{r,1}$*) and* $x_{i,i+1} := 1$ $\forall i = 1, \ldots, j$ *(all arcs before the excluded arc will be included in the tour).*

(6) **Bounding:**
For each node P_{s+j} *do:*
Solve an assignment problem including the variables preset in Step (5).
Let its objective value be denoted by $\bar{z}_{s+j}$.
If $\bar{z}_{s+j} \geq z^*$, *disregard the node* P_{s+j} *(fathoming).*
If $\bar{z}_{s+j} < z^*$ *and the solution includes no subtours, set* $z^* := \bar{z}_{s+j}$.
If $\bar{z}_{s+j} < z^*$ *and the solution includes subtours, the node* P_{s+j} *is live.*
Goto Step (3).

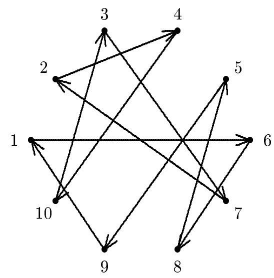
(Output) *Optimal TSP tour.*

Example 7.7:

Consider the following TSP with $n = 10$ and distance matrix $C = (c_{ij})$:

$$C = \begin{pmatrix} - & 9 & 8 & 3 & 5 & 1 & 3 & 3 & 7 & 4 \\ 2 & - & 6 & 1 & 3 & 6 & 1 & 9 & 9 & 7 \\ 9 & 2 & - & 8 & 8 & 5 & 1 & 3 & 7 & 9 \\ 3 & 6 & 9 & - & 5 & 4 & 8 & 9 & 6 & 2 \\ 8 & 5 & 6 & 5 & - & 3 & 7 & 6 & 4 & 5 \\ 9 & 2 & 8 & 2 & 8 & - & 9 & 4 & 3 & 3 \\ 4 & 1 & 3 & 6 & 6 & 5 & - & 4 & 2 & 8 \\ 7 & 7 & 6 & 1 & 1 & 9 & 4 & - & 6 & 9 \\ 1 & 6 & 9 & 6 & 2 & 8 & 6 & 3 & - & 3 \\ 6 & 8 & 2 & 5 & 6 & 7 & 7 & 5 & 7 & - \end{pmatrix}$$

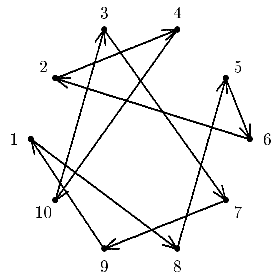
One of the solutions of the assignment problem contains the two subtours $(6, 8, 5, 9, 1, 6)$ and $(2, 4, 10, 3, 7, 2)$ and has a total cost of 18:



Branching is done on the first of these subtours and $r = 5$ subproblems are created from node P_1, with the following sets of included and excluded arcs:

Node P_2: $x_{68} = 0$
Node P_3: $x_{85} = 0$; $x_{68} = 1$
Node P_4: $x_{59} = 0$; $x_{68} = x_{85} = 1$
Node P_5: $x_{91} = 0$; $x_{68} = x_{85} = x_{59} = 1$
Node P_6: $x_{16} = 0$; $x_{68} = x_{85} = x_{59} = x_{91} = 1$.

The solutions of the corresponding assignment problems have objective values of 18, 19, 18, 19 and 19, respectively.
The process continues selecting a live node with the lowest objective value, e.g., node P_2. The solution generated at node P_2 consists of the two subtours $(7, 2, 7)$ and $(1, 6, 4, 10, 3, 8, 5, 9, 1)$. Branching is therefore done on the first of these subtours. The first of the two resulting subproblems with $x_{72} = 0$ (in addition to $x_{68} = 0$) yields the Hamiltonian circuit $(1, 8, 5, 6, 2, 4, 10, 3, 7, 9, 1)$ with cost 18. As node P_7 is one of the live nodes with lowest objective value, it is therefore optimal.



The branching tree is shown below: