



SAPIENZA
UNIVERSITÀ DI ROMA

Combinatorial Optimization

Exercises 6 — Ad hoc algorithms

Worked exercises on the chapters of Volume 6, ad hoc algorithms: each exercise is followed by its complete solution.

Fabio Furini

fabiofurini.github.io

Department of Computer, Control and Management Engineering — Sapienza University of Rome

Online version and downloadable material:
fabiofurini.github.io/combinatorial-optimization/
github.com/fabiofurini/combinatorial-optimization

Academic year 2026–2027

Contents

Knapsack problems	2
41 Knapsack: branch-and-bound on the critical item	3
Coloring problems	30
42 Vertex coloring: the DSATUR branch-and-bound	31



Knapsack problems



Knapsack: branch-and-bound on the critical item

Exercise 41.1 — Branch-and-Bound Algorithm for the 0/1 Knapsack Problem (P1)

We consider a knapsack capacity $c = 11$ and a set $\mathcal{J} = \{1, 2, 3, 4, 5\}$ of 5 items ($n = 5$) with the following profits and weights:

	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
p_j	8	6	14	6	2
w_j	3	3	9	5	2

Questions

1. Check if the items are sorted by non-increasing efficiency otherwise sort them in this order. Write the ILP_{KP} model and the LP_{KP} model for the KP instance.
2. Execute the branch-and-bound algorithm to solve the ILP_{KP} model for the 0/1 knapsack problem (KP) using:
 - (a) the split-item rule as branching-variable selection strategy;
 - (b) the LIFO rule as branching-node selection strategy;
 - (c) solve first the down-branching node and then the up-branching node;
 - (d) set the incumbent solution $\bar{\mathbf{x}}(\text{ILP}_{\text{KP}})$ to the following heuristic solution :

$$\bar{x}_1 = 1, \quad \bar{x}_2 = 1, \quad \bar{x}_3 = 0, \quad \bar{x}_4 = 1, \quad \bar{x}_5 = 0$$

and set the lower bound LB accordingly.

3. Determine and comment the correspondent branch-and-bound tree showing the reason why the branching nodes can be pruned.

Solution

1. The items are sorted by non-increasing efficiency since we have:

$$\frac{8}{3} \geq \frac{6}{3} \geq \frac{14}{9} \geq \frac{6}{5} \geq \frac{2}{2}$$

This is the ILP_{KP} model for the KP instance:

$$\begin{aligned} \max \quad & 8x_1 + 6x_2 + 14x_3 + 6x_4 + 2x_5 \\ & 3x_1 + 3x_2 + 9x_3 + 5x_4 + 2x_5 \leq 11 \\ & x_1, x_2, x_3, x_4, x_5 \in \{0, 1\} \end{aligned}$$

This is the LP_{KP} model for the KP instance:

$$\begin{aligned} \max \quad & 8x_1 + 6x_2 + 14x_3 + 6x_4 + 2x_5 \\ & 3x_1 + 3x_2 + 9x_3 + 5x_4 + 2x_5 \leq 11 \\ & x_1 \leq 1 \\ & x_2 \leq 1 \\ & x_3 \leq 1 \\ & x_4 \leq 1 \\ & x_5 \leq 1 \\ & x_1, x_2, x_3, x_4, x_5 \geq 0 \end{aligned}$$

2. Execution of the branch-and-bound algorithm to solve the ILP_{KP} model.

- **Initialization**

For the LP_{KP}¹ problem, the split item is $s = 3$ and the residual capacity is $rc = 5$. An optimal solution $\tilde{x}(\text{LP}_{\text{KP}}^1)$ is:

$$\tilde{x}_1 = 1, \tilde{x}_2 = 1, \tilde{x}_3 = \frac{5}{9}, \tilde{x}_4 = 0, \tilde{x}_5 = 0$$

the optimal value is:

$$z(\text{LP}_{\text{KP}}^1) = 8 \cdot 1 + 6 \cdot 1 + 14 \cdot \frac{5}{9} = \frac{196}{9}$$

The optimal solution $\tilde{x}(\text{LP}_{\text{KP}}^1)$ is **fractional** and the LP_{KP}¹ problem is **feasible**, so the LP_{KP}¹ problem is inserted in the problem list L and the lower bound LB is set to 20 ($\text{LB}(\tilde{x}) = 20$).

Due to the current lower bound LB and upper bound UB, we have:

$$20 \leq z(\text{ILP}_{\text{KP}}) \leq \underbrace{\left\lfloor \frac{196}{9} \right\rfloor}_{=21}$$

• **Iteration 1**

The LP_{KP}^1 problem is extracted and deleted from the problem list L .

The branching variable is x_3 . The LP_{KP}^2 and LP_{KP}^3 problems are created by unselecting ($x_3 \leq 0$) and selecting ($x_3 \geq 1$) item 3.

(a) Solution of the LP_{KP}^2 problem

We have

$$\mathcal{J}^2 = \{1, 2, 4, 5\}, \quad \mathcal{J}_{out}^2 = \{3\}, \quad \mathcal{J}_{in}^2 = \emptyset$$

The capacity is $c(2) = 11$ and the objective-function constant is $p(2) = 0$. The split item is $s = 5$ and the residual capacity is $rc = 0$. An optimal solution $\tilde{x}(LP_{KP}^2)$ is:

$$\tilde{x}_1 = 1, \quad \tilde{x}_2 = 1, \quad \tilde{x}_4 = 1, \quad \tilde{x}_5 = 0, \quad \text{and} \quad z(LP_{KP}^2) = 20$$

The LP_{KP}^2 problem is **feasible** and its optimal solution $\tilde{x}(LP_{KP}^2)$ is **integer**. It does not improve the incumbent solution, since $z(LP_{KP}^2) = 20 \leq 20 = LB$, accordingly, the branching node is pruned by integrality (when both criteria hold, the integrality comes first).

(b) Solution of the LP_{KP}^3 problem

We have

$$\mathcal{J}^3 = \{1, 2, 4, 5\}, \quad \mathcal{J}_{out}^3 = \emptyset, \quad \mathcal{J}_{in}^3 = \{3\}$$

The capacity is $c(3) = 2$ and the objective-function constant is $p(3) = 14$. The split item is $s = 1$ and the residual capacity is $rc = 2$. An optimal solution $\tilde{x}(LP_{KP}^3)$ is:

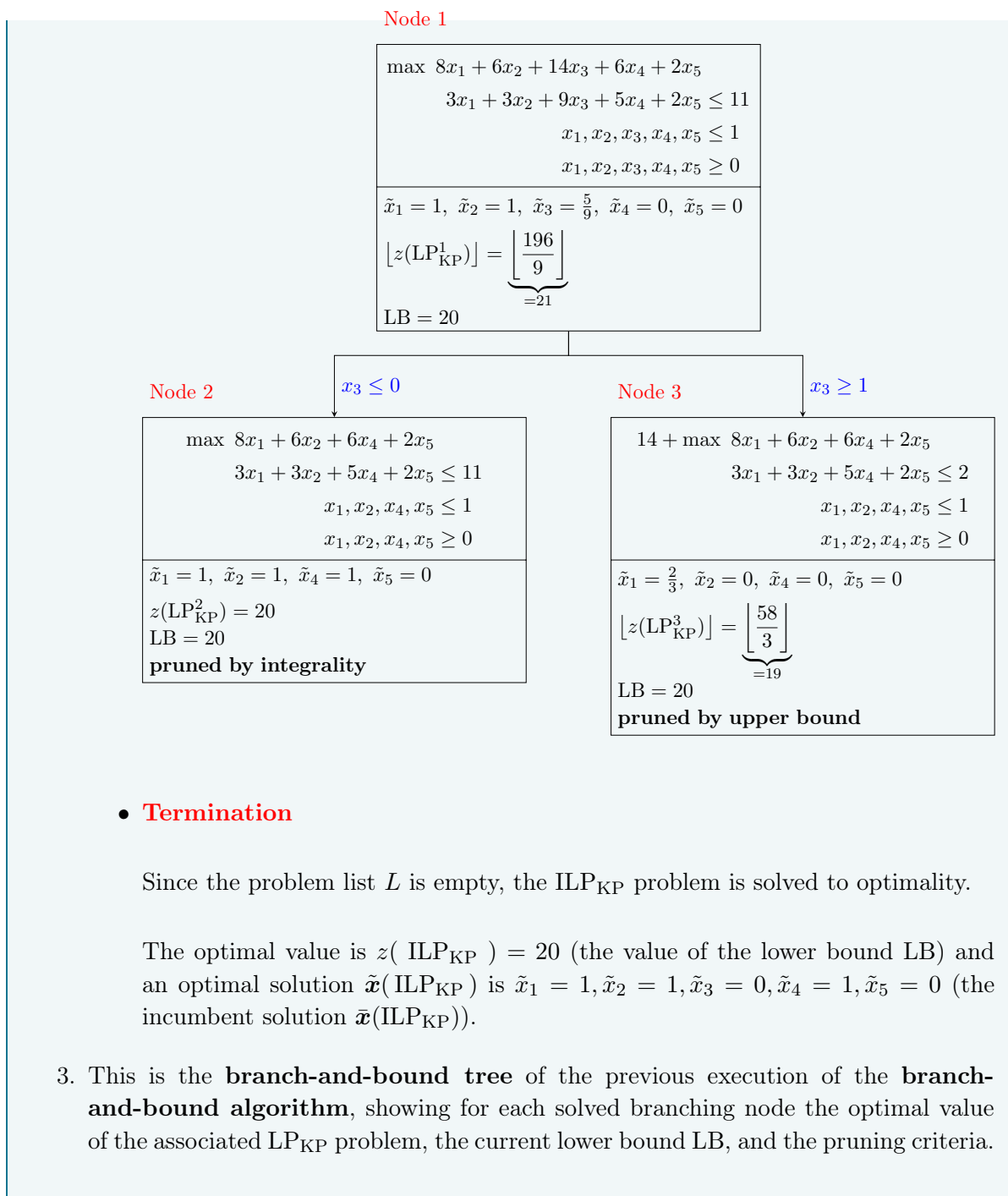
$$\tilde{x}_1 = \frac{2}{3}, \quad \tilde{x}_2 = 0, \quad \tilde{x}_4 = 0, \quad \tilde{x}_5 = 0, \quad \text{and} \quad z(LP_{KP}^3) = 14 + 8 \cdot \frac{2}{3} = \frac{58}{3}$$

The LP_{KP}^3 problem is **feasible** and we have:

$$\lfloor z(LP_{KP}^3) \rfloor \leq LB \quad \text{since} \quad \underbrace{\left\lfloor \frac{58}{3} \right\rfloor}_{=19} \leq 20$$

accordingly, the branching node is pruned by upper bound.

At the end of the first iteration, Node 2 is pruned by integrality and Node 3 is pruned by upper bound. The corresponding branch-and-bound tree is:

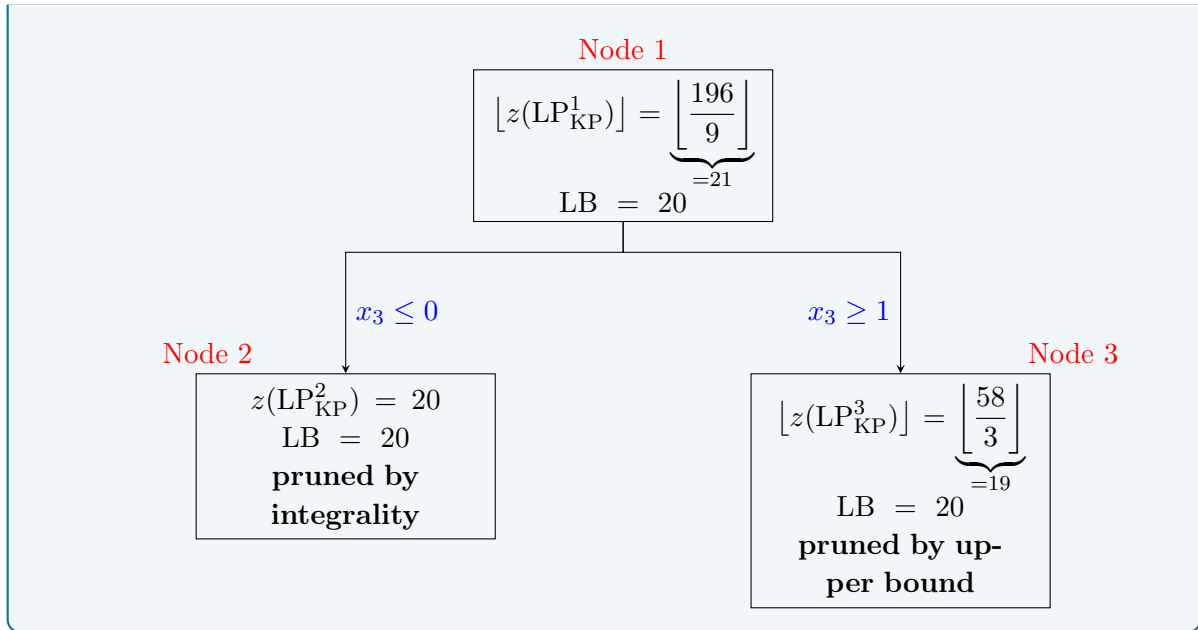


- **Termination**

Since the problem list L is empty, the ILP_{KP} problem is solved to optimality.

The optimal value is $z(\text{ILP}_{\text{KP}}) = 20$ (the value of the lower bound LB) and an optimal solution $\tilde{\mathbf{x}}(\text{ILP}_{\text{KP}})$ is $\tilde{x}_1 = 1, \tilde{x}_2 = 1, \tilde{x}_3 = 0, \tilde{x}_4 = 1, \tilde{x}_5 = 0$ (the incumbent solution $\bar{\mathbf{x}}(\text{ILP}_{\text{KP}})$).

3. This is the **branch-and-bound tree** of the previous execution of the **branch-and-bound algorithm**, showing for each solved branching node the optimal value of the associated LP_{KP} problem, the current lower bound LB, and the pruning criteria.



Exercise 41.2 — Branch-and-Bound Algorithm for the 0/1 Knapsack Problem (P2)

We consider a knapsack capacity $c = 8$ and a set $\mathcal{J} = \{1, 2, 3, 4\}$ of 4 items ($n = 4$) with the following profits and weights:

	$j = 1$	$j = 2$	$j = 3$	$j = 4$
p_j	40	42	25	12
w_j	4	7	5	3

Questions

1. Check if the items are sorted by non-increasing efficiency otherwise sort them in this order. Write the ILP_{KP} model and the LP_{KP} model for the KP instance.
2. Execute the branch-and-bound algorithm to solve the ILP_{KP} model for the 0/1 knapsack problem (KP) using:
 - (a) the split-item rule as branching-variable selection strategy;
 - (b) the LIFO rule as branching-node selection strategy;
 - (c) solve first the down-branching node and then the up-branching node;
 - (d) set the incumbent solution $\bar{x}(\text{ILP}_{\text{KP}})$ to the following heuristic solution :

$$\bar{x}_1 = 1, \quad \bar{x}_2 = 0, \quad \bar{x}_3 = 0, \quad \bar{x}_4 = 1$$

and set the lower bound LB accordingly.

3. Determine and comment the correspondent branch-and-bound tree showing the reason why the branching nodes can be pruned.

Solution

1. The items are sorted by non-increasing efficiency since we have:

$$\frac{40}{4} \geq \frac{42}{7} \geq \frac{25}{5} \geq \frac{12}{3}$$

This is the ILP_{KP} model for the KP instance:

$$\begin{aligned} \max \quad & 40 x_1 + 42 x_2 + 25 x_3 + 12 x_4 \\ & 4 x_1 + 7 x_2 + 5 x_3 + 3 x_4 \leq 8 \\ & x_1, x_2, x_3, x_4 \in \{0, 1\} \end{aligned}$$

This is the LP_{KP} model for the KP instance:

$$\begin{aligned} \max \quad & 40 x_1 + 42 x_2 + 25 x_3 + 12 x_4 \\ & 4 x_1 + 7 x_2 + 5 x_3 + 3 x_4 \leq 8 \\ & x_1 \leq 1 \\ & x_2 \leq 1 \\ & x_3 \leq 1 \\ & x_4 \leq 1 \\ & x_1, x_2, x_3, x_4 \geq 0 \end{aligned}$$

2. Execution of the branch-and-bound algorithm to solve the ILP_{KP} model.

- **Initialization**

For the LP_{KP}¹ problem, the split item is $s = 2$ and the residual capacity is $rc = 4$. An optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^1)$ is:

$$\tilde{x}_1 = 1, \quad \tilde{x}_2 = \frac{4}{7}, \quad \tilde{x}_3 = 0, \quad \tilde{x}_4 = 0$$

the optimal value is:

$$z(\text{LP}_{\text{KP}}^1) = 40 \cdot 1 + 42 \cdot \frac{4}{7} = 64$$

The optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^1)$ is **fractional** and the LP_{KP}¹ problem is **feasible**, so the LP_{KP}¹ problem is inserted in the problem list L and the lower bound LB is set to 52 ($\text{LB}(\tilde{\mathbf{x}}) = 52$).

Due to the current lower bound LB and upper bound UB, we have:

$$52 \leq z(\text{ILP}_{\text{KP}}) \leq 64$$

- **Iteration 1**

The LP_{KP}^1 problem is extracted and deleted from the problem list L .

The branching variable is x_2 . The LP_{KP}^2 and LP_{KP}^3 problems are created by unselecting ($x_2 \leq 0$) and selecting ($x_2 \geq 1$) item 2.

(a) Solution of the LP_{KP}^2 problem

We have

$$\mathcal{J}^2 = \{1, 3, 4\}, \quad \mathcal{J}_{out}^2 = \{2\}, \quad \mathcal{J}_{in}^2 = \emptyset$$

The capacity is $c(2) = 8$ and the objective-function constant is $p(2) = 0$. The split item is $s = 3$ and the residual capacity is $rc = 4$. An optimal solution $\tilde{x}(LP_{KP}^2)$ is:

$$\tilde{x}_1 = 1, \quad \tilde{x}_3 = \frac{4}{5}, \quad \tilde{x}_4 = 0 \quad \text{and} \quad z(LP_{KP}^2) = 40 \cdot 1 + 25 \cdot \frac{4}{5} = 60$$

The LP_{KP}^2 problem is **feasible** and we have:

$$\lfloor z(LP_{KP}^2) \rfloor > LB \quad \text{since} \quad 60 > 52$$

The optimal solution $\tilde{x}(LP_{KP}^2)$ is **fractional**, accordingly, the LP_{KP}^2 problem is **inserted in the problem list L** .

(b) Solution of the LP_{KP}^3 problem

We have

$$\mathcal{J}^3 = \{1, 3, 4\}, \quad \mathcal{J}_{out}^3 = \emptyset, \quad \mathcal{J}_{in}^3 = \{2\}$$

The capacity is $c(3) = 1$ and the objective-function constant is $p(3) = 42$. The split item is $s = 1$ and the residual capacity is $rc = 1$. An optimal solution $\tilde{x}(LP_{KP}^3)$ is:

$$\tilde{x}_1 = \frac{1}{4}, \quad \tilde{x}_3 = 0, \quad \tilde{x}_4 = 0 \quad \text{and} \quad z(LP_{KP}^3) = 42 + 40 \cdot \frac{1}{4} = 52$$

The LP_{KP}^3 problem is **feasible** and we have:

$$\lfloor z(LP_{KP}^3) \rfloor \leq LB \quad \text{since} \quad 52 \leq 52$$

accordingly, the branching node is pruned by upper bound.

At the end of the first iteration, Node 2 is inserted in the problem list L , while Node 3 is pruned by upper bound. The corresponding branch-and-bound tree is:

Node 1

$$\begin{array}{ll}
 \max & 40x_1 + 42x_2 + 25x_3 + 12x_4 \\
 & 4x_1 + 7x_2 + 5x_3 + 3x_4 \leq 8 \\
 & x_1, x_2, x_3, x_4 \leq 1 \\
 & x_1, x_2, x_3, x_4 \geq 0 \\
 \hline
 & \tilde{x}_1 = 1, \tilde{x}_2 = \frac{4}{7}, \tilde{x}_3 = 0, \tilde{x}_4 = 0 \\
 & z(\text{LP}_{\text{KP}}^1) = 64 \\
 & \text{LB} = 52
 \end{array}$$

Node 2

$$x_2 \leq 0$$

$$\begin{array}{ll}
 \max & 40x_1 + 25x_3 + 12x_4 \\
 & 4x_1 + 5x_3 + 3x_4 \leq 8 \\
 & x_1, x_3, x_4 \leq 1 \\
 & x_1, x_3, x_4 \geq 0 \\
 \hline
 & \tilde{x}_1 = 1, \tilde{x}_3 = \frac{4}{5}, \tilde{x}_4 = 0 \\
 & z(\text{LP}_{\text{KP}}^2) = 60 \\
 & \text{LB} = 52
 \end{array}$$

Node 3

$$x_2 \geq 1$$

$$\begin{array}{ll}
 42 + \max & 40x_1 + 25x_3 + 12x_4 \\
 & 4x_1 + 5x_3 + 3x_4 \leq 1 \\
 & x_1, x_3, x_4 \leq 1 \\
 & x_1, x_3, x_4 \geq 0 \\
 \hline
 & \tilde{x}_1 = \frac{1}{4}, \tilde{x}_3 = 0, \tilde{x}_4 = 0 \\
 & z(\text{LP}_{\text{KP}}^3) = 52 \\
 & \text{LB} = 52 \\
 & \text{pruned by upper bound}
 \end{array}$$

Due to the current lower bound LB and upper bound UB, we have:

$$52 \leq z(\text{ILP}_{\text{KP}}) \leq 60$$

• Iteration 2

The LP_{KP}^2 problem is extracted and deleted from the problem list L .

The branching variable is x_3 . The LP_{KP}^4 and LP_{KP}^5 problems are created by unselecting ($x_3 \leq 0$) and selecting ($x_3 \geq 1$) item 3.

(a) Solution of the LP_{KP}^4 problem

We have

$$\mathcal{J}^4 = \{1, 4\}, \quad \mathcal{J}_{\text{out}}^4 = \{2, 3\}, \quad \mathcal{J}_{\text{in}}^4 = \emptyset$$

The capacity is $c(4) = 8$ and the objective-function constant is $p(4) = 0$. All items fit into the knapsack. An optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^4)$ is:

$$\tilde{x}_1 = 1, \quad \tilde{x}_4 = 1 \quad \text{and} \quad z(\text{LP}_{\text{KP}}^4) = 40 \cdot 1 + 12 \cdot 1 = 52$$

The LP_{KP}^4 problem is **feasible** and its optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^4)$ is **integer**. It does not improve the incumbent solution, since $z(\text{LP}_{\text{KP}}^4) = 52 \leq 52 = \text{LB}$, accordingly, the branching node is pruned by integrality (when both criteria hold, the integrality comes first).

(b) Solution of the LP_{KP}^5 problem

We have

$$\mathcal{J}^5 = \{1, 4\}, \quad \mathcal{J}_{\text{out}}^5 = \{2\}, \quad \mathcal{J}_{\text{in}}^5 = \{3\}$$

The capacity is $c(5) = 3$ and the objective-function constant is $p(5) = 25$. The split item is $s = 1$ and the residual capacity is $rc = 3$. An optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^5)$ is:

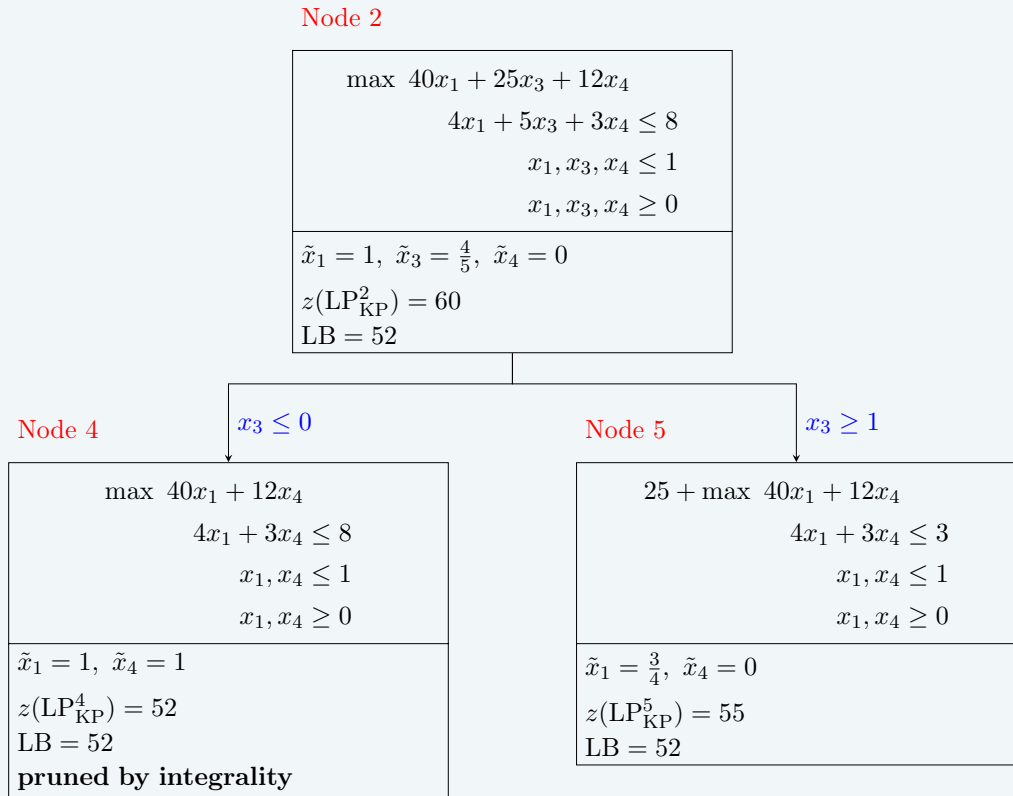
$$\tilde{x}_1 = \frac{3}{4}, \quad \tilde{x}_2 = 0, \quad \tilde{x}_3 = 1, \quad \tilde{x}_4 = 0 \quad \text{and} \quad z(\text{LP}_{\text{KP}}^5) = 25 + 40 \cdot \frac{3}{4} = 55$$

The LP_{KP}^5 problem is **feasible** and we have:

$$\lfloor z(\text{LP}_{\text{KP}}^5) \rfloor > \text{LB} \quad \text{since} \quad 55 > 52$$

The optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^5)$ is **fractional**, accordingly, the LP_{KP}^5 problem is **inserted in the problem list L** .

At the end of the second iteration, Node 5 is inserted in the problem list L , while Node 4 is pruned by integrality. The corresponding branch-and-bound tree is:



Due to the current lower bound LB and upper bound UB, we have:

$$52 \leq z(\text{ILP}_{\text{KP}}) \leq 55$$

- **Iteration 3**

The LP_{KP}^5 problem is extracted and deleted from the problem list L .

The branching variable is x_1 . The LP_{KP}^6 and LP_{KP}^7 problems are created by unselecting ($x_1 \leq 0$) and selecting ($x_1 \geq 1$) item 1.

(a) Solution of the LP_{KP}^6 problem

We have

$$\mathcal{I}^6 = \{4\}, \quad \mathcal{I}_{out}^6 = \{1, 2\}, \quad \mathcal{I}_{in}^6 = \{3\}$$

The capacity is $c(6) = 3$ and the objective-function constant is $p(6) = 25$. All items fit into the knapsack. An optimal solution $\tilde{x}(LP_{KP}^6)$ is:

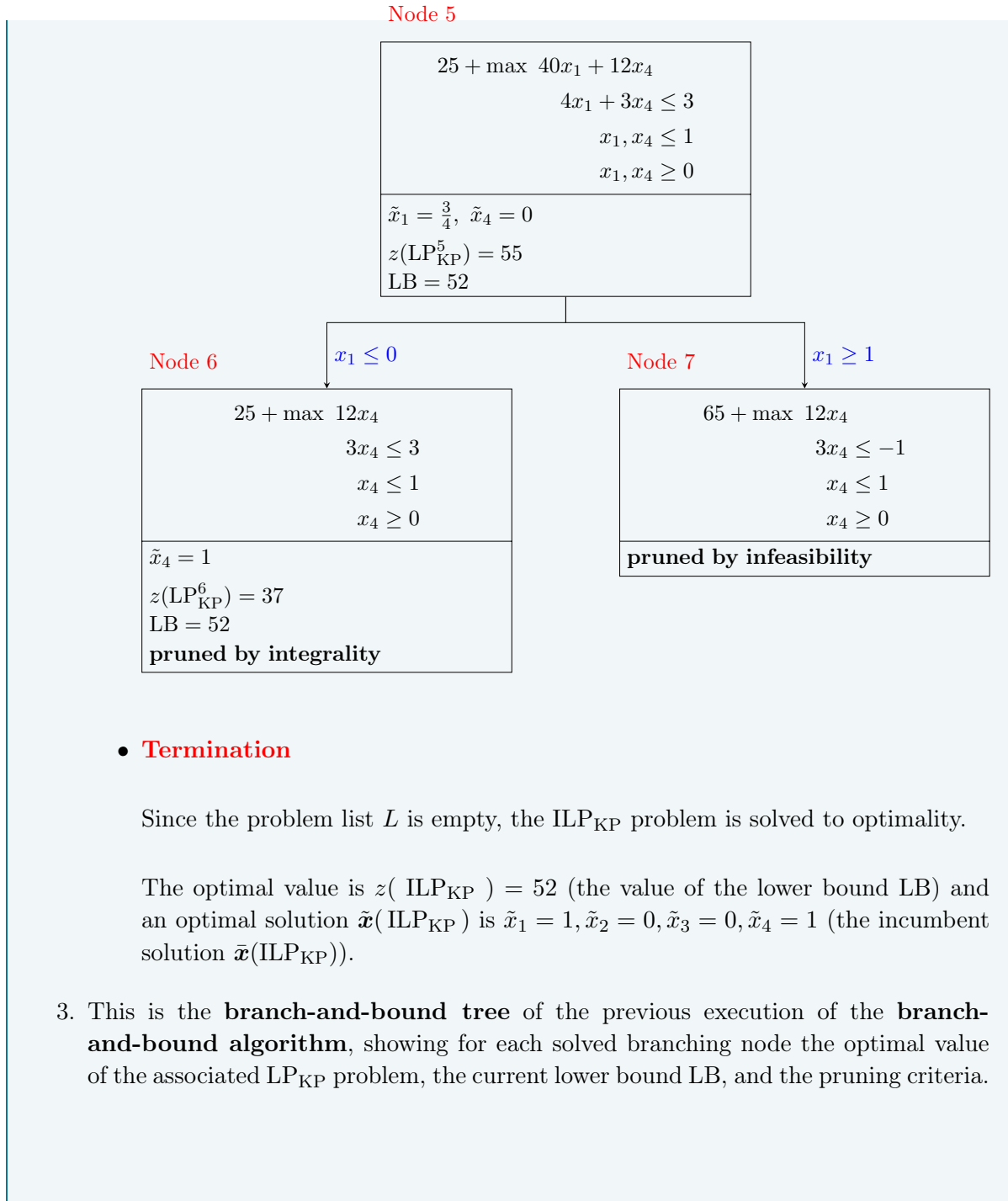
$$\tilde{x}_4 = 1 \quad \text{and} \quad z(LP_{KP}^6) = 25 + 12 \cdot 1 = 37$$

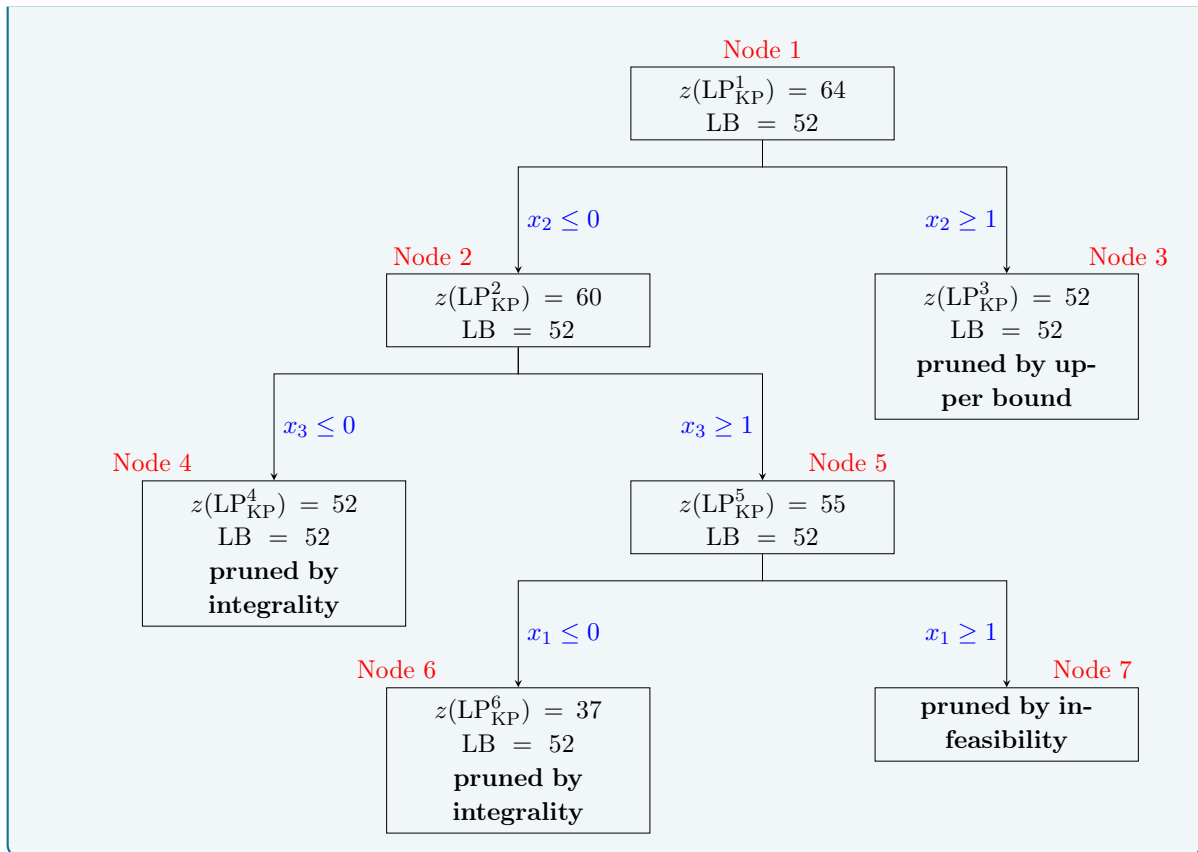
The LP_{KP}^6 problem is **feasible** and its optimal solution $\tilde{x}(LP_{KP}^6)$ is **integer**. It does not improve the incumbent solution, since $z(LP_{KP}^6) = 37 < 52 = LB$, accordingly, the branching node is pruned by integrality (when both criteria hold, the integrality comes first).

(b) Solution of the LP_{KP}^7 problem

The LP_{KP}^7 problem is **infeasible**, accordingly the branching node is pruned by infeasibility.

At the end of the third iteration, Node 6 is pruned by integrality and Node 7 is pruned by infeasibility. The corresponding branch-and-bound tree is:





Exercise 41.3 — Branch-and-Bound Algorithm for the 0/1 Knapsack Problem (P3)

We consider a knapsack capacity $c = 15$ and a set $\mathcal{J} = \{1, 2, 3, 4\}$ of 4 items ($n = 4$) with the following profits and weights:

	$j = 1$	$j = 2$	$j = 3$	$j = 4$
p_j	24	17	12	6
w_j	10	8	6	5

Questions

1. Check if the items are sorted by non-increasing efficiency otherwise sort them in this order. Write the ILP_{KP} model and the LP_{KP} model for the KP instance.
2. Execute the branch-and-bound algorithm to solve the ILP_{KP} model for the 0/1 knapsack problem (KP) using:
 - (a) the split-item rule as branching-variable selection strategy;
 - (b) the LIFO rule as branching-node selection strategy;
 - (c) solve first the down-branching node and then the up-branching node;

(d) set the incumbent solution $\bar{\mathbf{x}}(\text{ILP}_{\text{KP}})$ to the following heuristic solution :

$$\bar{x}_1 = 1, \quad \bar{x}_2 = 0, \quad \bar{x}_3 = 0, \quad \bar{x}_4 = 1$$

and set the lower bound LB accordingly.

3. Determine and comment the correspondent branch-and-bound tree showing the reason why the branching nodes can be pruned.

Solution

1. The items are sorted by non-increasing efficiency since we have:

$$\frac{24}{10} \geq \frac{17}{8} \geq \frac{12}{6} \geq \frac{6}{5}$$

This is the ILP_{KP} model for the KP instance:

$$\begin{aligned} \max \quad & 24 x_1 + 17 x_2 + 12 x_3 + 6 x_4 \\ & 10 x_1 + 8 x_2 + 6 x_3 + 5 x_4 \leq 15 \\ & x_1, x_2, x_3, x_4 \in \{0, 1\} \end{aligned}$$

This is the LP_{KP} model for the KP instance:

$$\begin{aligned} \max \quad & 24 x_1 + 17 x_2 + 12 x_3 + 6 x_4 \\ & 10 x_1 + 8 x_2 + 6 x_3 + 5 x_4 \leq 15 \\ & x_1 \leq 1 \\ & x_2 \leq 1 \\ & x_3 \leq 1 \\ & x_4 \leq 1 \\ & x_1, x_2, x_3, x_4 \geq 0 \end{aligned}$$

2. Execution of the branch-and-bound algorithm to solve the ILP_{KP} model.

- **Initialization**

For the LP_{KP}^1 problem, the split item is $s = 2$ and the residual capacity is $rc = 5$. An optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^1)$ is:

$$\tilde{x}_1 = 1, \quad \tilde{x}_2 = \frac{5}{8}, \quad \tilde{x}_3 = 0, \quad \tilde{x}_4 = 0$$

the optimal value is:

$$z(\text{LP}_{\text{KP}}^1) = 24 \cdot 1 + 17 \cdot \frac{5}{8} = \frac{277}{8}$$

The optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^1)$ is **fractional** and the LP_{KP}^1 problem is **feasible**, so the LP_{KP}^1 problem is inserted in the problem list L and the lower bound LB is set to 30 ($\text{LB}(\tilde{\mathbf{x}}) = 30$).

Due to the current lower bound LB and upper bound UB , we have:

$$30 \leq z(\text{ILP}_{\text{KP}}) \leq \underbrace{\left\lfloor \frac{277}{8} \right\rfloor}_{=34}$$

• **Iteration 1**

The LP_{KP}^1 problem is extracted and deleted from the problem list L .

The branching variable is x_2 . The LP_{KP}^2 and LP_{KP}^3 problems are created by unselecting ($x_2 \leq 0$) and selecting ($x_2 \geq 1$) item 2.

(a) Solution of the LP_{KP}^2 problem

We have

$$\mathcal{J}^2 = \{1, 3, 4\}, \quad \mathcal{J}_{\text{out}}^2 = \{2\}, \quad \mathcal{J}_{\text{in}}^2 = \emptyset$$

The capacity is $c(2) = 15$ and the objective-function constant is $p(2) = 0$. The split item is $s = 3$ and the residual capacity is $rc = 5$. An optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^2)$ is:

$$\tilde{x}_1 = 1, \quad \tilde{x}_3 = \frac{5}{6}, \quad \tilde{x}_4 = 0 \quad \text{and} \quad z(\text{LP}_{\text{KP}}^2) = 24 \cdot 1 + 12 \cdot \frac{5}{6} = 34$$

The LP_{KP}^2 problem is **feasible** and we have:

$$\lfloor z(\text{LP}_{\text{KP}}^2) \rfloor > \text{LB} \quad \text{since} \quad 34 > 30$$

The optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^2)$ is **fractional**, accordingly, the LP_{KP}^2 problem is **inserted in the problem list** L .

(b) Solution of the LP_{KP}^3 problem

We have

$$\mathcal{J}^3 = \{1, 3, 4\}, \quad \mathcal{J}_{\text{out}}^3 = \emptyset, \quad \mathcal{J}_{\text{in}}^3 = \{2\}$$

The capacity is $c(3) = 7$ and the objective-function constant is $p(3) = 17$. The split item is $s = 1$ and the residual capacity is $rc = 7$. An optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^3)$ is:

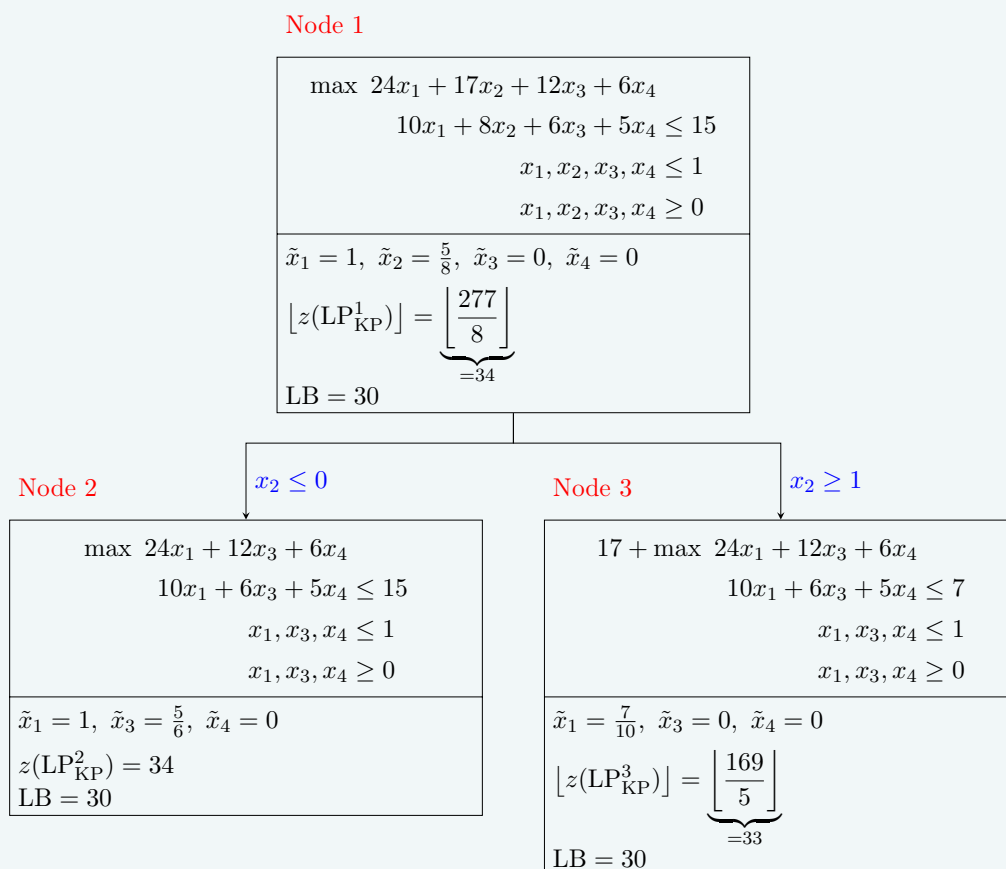
$$\tilde{x}_1 = \frac{7}{10}, \quad \tilde{x}_3 = 0, \quad \tilde{x}_4 = 0 \quad \text{and} \quad z(\text{LP}_{\text{KP}}^3) = 17 + 24 \cdot \frac{7}{10} = \frac{169}{5}$$

The LP_{KP}^3 problem is **feasible** and we have:

$$\lfloor z(\text{LP}_{\text{KP}}^3) \rfloor > \text{LB} \quad \text{since} \quad \underbrace{\left\lfloor \frac{169}{5} \right\rfloor}_{=33} > 30$$

The optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^3)$ is **fractional**, accordingly, the LP_{KP}^3 problem is **inserted in the problem list L** .

At the end of the first iteration, Node 2 and Node 3 are both inserted in the problem list L . The corresponding branch-and-bound tree is:



Due to the current lower bound LB and upper bound UB, we have:

$$30 \leq z(\text{ILP}_{\text{KP}}) \leq \underbrace{\max \left\{ 34, \underbrace{\left\lfloor \frac{169}{5} \right\rfloor}_{=33} \right\}}_{=34}$$

• **Iteration 2**

The LP_{KP}^3 problem is extracted and deleted from the problem list L .

The branching variable is x_1 . The LP_{KP}^4 and LP_{KP}^5 problems are created by unselecting ($x_1 \leq 0$) and selecting ($x_1 \geq 1$) item 1.

(a) Solution of the LP_{KP}^4 problem

We have

$$\mathcal{J}^4 = \{3, 4\}, \quad \mathcal{J}_{\text{out}}^4 = \{1\}, \quad \mathcal{J}_{\text{in}}^4 = \{2\}$$

The capacity is $c(4) = 7$ and the objective-function constant is $p(4) = 17$. The split item is $s = 4$ and the residual capacity is $rc = 1$. An optimal solution $\tilde{x}(\text{LP}_{\text{KP}}^4)$ is:

$$\tilde{x}_3 = 1, \quad \tilde{x}_4 = \frac{1}{5} \quad \text{and} \quad z(\text{LP}_{\text{KP}}^4) = 17 + 12 \cdot 1 + 6 \cdot \frac{1}{5} = \frac{151}{5}$$

The LP_{KP}^4 problem is **feasible** and we have:

$$\lfloor z(\text{LP}_{\text{KP}}^4) \rfloor \leq \text{LB} \quad \text{since} \quad \underbrace{\left\lfloor \frac{151}{5} \right\rfloor}_{=30} \leq 30$$

accordingly, the branching node is pruned by upper bound.

(b) Solution of the LP_{KP}^5 problem

The LP_{KP}^5 problem is **infeasible**, accordingly the branching node is pruned by infeasibility.

At the end of the second iteration, Node 4 is pruned by upper bound and Node 5 is pruned by infeasibility. The corresponding branch-and-bound tree is:

Node 3

$$\begin{aligned}
 &17 + \max \quad 24x_1 + 12x_3 + 6x_4 \\
 &\quad 10x_1 + 6x_3 + 5x_4 \leq 7 \\
 &\quad x_1, x_3, x_4 \leq 1 \\
 &\quad x_1, x_3, x_4 \geq 0 \\
 &\tilde{x}_1 = \frac{7}{10}, \tilde{x}_3 = 0, \tilde{x}_4 = 0 \\
 &\lfloor z(\text{LP}_{\text{KP}}^3) \rfloor = \underbrace{\left\lfloor \frac{169}{5} \right\rfloor}_{=33} \\
 &\text{LB} = 30
 \end{aligned}$$

Node 4

$$x_1 \leq 0$$

$$\begin{aligned}
 &41 + \max \quad 12x_3 + 6x_4 \\
 &\quad 6x_3 + 5x_4 \leq 7 \\
 &\quad x_3, x_4 \leq 1 \\
 &\quad x_3, x_4 \geq 0 \\
 &\tilde{x}_3 = 1, \tilde{x}_4 = \frac{1}{5} \\
 &\lfloor z(\text{LP}_{\text{KP}}^4) \rfloor = \underbrace{\left\lfloor \frac{151}{5} \right\rfloor}_{=30} \\
 &\text{LB} = 30 \\
 &\text{pruned by upper bound}
 \end{aligned}$$

Node 5

$$x_1 \geq 1$$

$$\begin{aligned}
 &41 + \max \quad 12x_3 + 6x_4 \\
 &\quad 6x_3 + 5x_4 \leq -3 \\
 &\quad x_3, x_4 \leq 1 \\
 &\quad x_3, x_4 \geq 0 \\
 &\text{pruned by infeasibility}
 \end{aligned}$$

Due to the current lower bound LB and upper bound UB, we have:

$$30 \leq z(\text{ILP}_{\text{KP}}) \leq 34$$

• Iteration 3

The LP_{KP}^2 problem is extracted and deleted from the problem list L .

The branching variable is x_3 . The LP_{KP}^6 and LP_{KP}^7 problems are created by unselecting $(x_3 \leq 0)$ and selecting $(x_3 \geq 1)$ item 3.

(a) Solution of the LP_{KP}^6 problem

We have

$$\mathcal{J}^6 = \{1, 4\}, \quad \mathcal{J}_{\text{out}}^6 = \{2, 3\}, \quad \mathcal{J}_{\text{in}}^6 = \emptyset$$

The capacity is $c(6) = 15$ and the objective-function constant is $p(6) = 0$. All items fit into the knapsack. An optimal solution $\tilde{x}(\text{LP}_{\text{KP}}^6)$ is:

$$\tilde{x}_1 = 1, \tilde{x}_4 = 1 \quad \text{and} \quad z(\text{LP}_{\text{KP}}^6) = 24 \cdot 1 + 6 \cdot 1 = 30$$

The LP_{KP}^6 problem is **feasible** and its optimal solution $\tilde{x}(\text{LP}_{\text{KP}}^6)$ is **integer**. It does not improve the incumbent solution, since $z(\text{LP}_{\text{KP}}^6) = 30 \leq 30 = \text{LB}$,

accordingly, the branching node is pruned by integrality (when both criteria hold, the integrality comes first).

(b) Solution of the LP_{KP}^7 problem

We have

$$\mathcal{J}^7 = \{1, 4\}, \quad \mathcal{J}_{\text{out}}^7 = \{2\}, \quad \mathcal{J}_{\text{in}}^7 = \{3\}$$

The capacity is $c(7) = 9$ and the objective-function constant is $p(7) = 12$. The split item is $s = 1$ and the residual capacity is $rc = 9$. An optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^7)$ is:

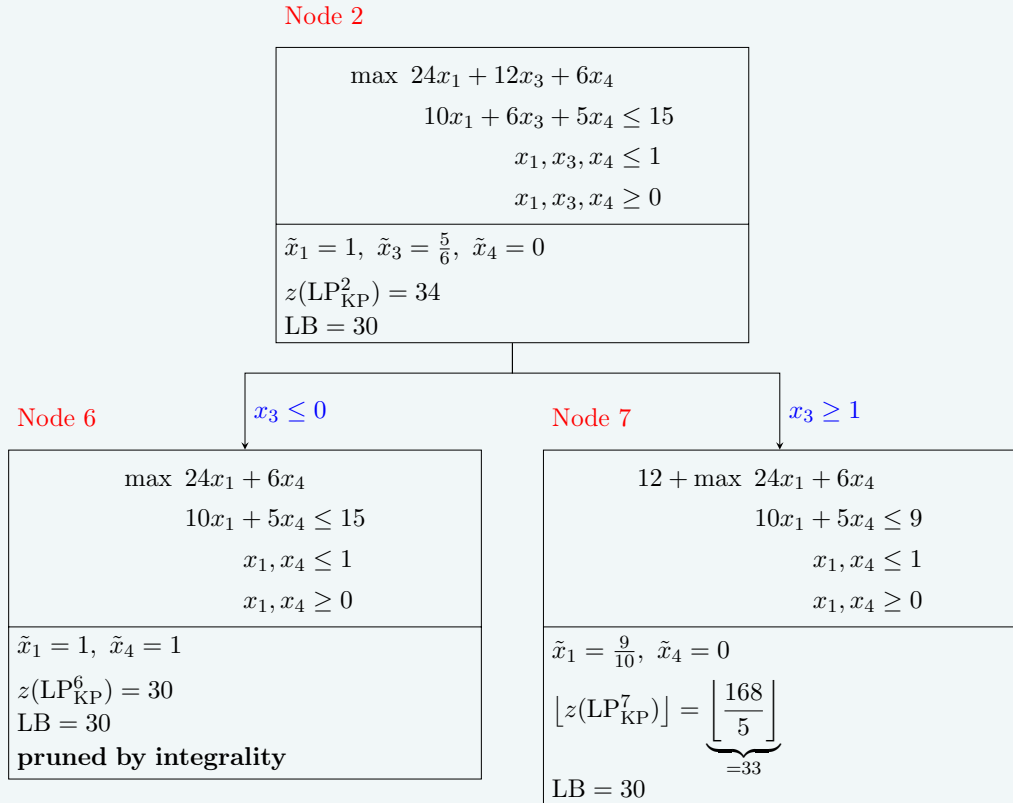
$$\tilde{x}_1 = \frac{9}{10}, \quad \tilde{x}_4 = 0 \quad \text{and} \quad z(\text{LP}_{\text{KP}}^7) = 12 + 24 \cdot \frac{9}{10} = \frac{168}{5}$$

The LP_{KP}^7 problem is **feasible** and we have:

$$\lfloor z(\text{LP}_{\text{KP}}^7) \rfloor > \text{LB} \quad \text{since} \quad \underbrace{\left\lfloor \frac{168}{5} \right\rfloor}_{=33} > 30$$

The optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^7)$ is **fractional**, accordingly, the LP_{KP}^7 problem is **inserted in the problem list L** .

At the end of the third iteration, Node 6 is pruned by integrality and Node 7 is inserted in the problem list L . The corresponding branch-and-bound tree is:



- **Iteration 4**

The LP_{KP}^7 problem is extracted and deleted from the problem list L .

The branching variable is x_1 . The LP_{KP}^8 and LP_{KP}^9 problems are created by unselecting ($x_1 \leq 0$) and selecting ($x_1 \geq 1$) item 1.

(a) Solution of the LP_{KP}^8 problem

We have

$$\mathcal{I}^8 = \{4\}, \quad \mathcal{I}_{out}^8 = \{1, 2\}, \quad \mathcal{I}_{in}^8 = \{3\}$$

The capacity is $c(8) = 9$ and the objective-function constant is $p(8) = 12$. All items fit into the knapsack. An optimal solution $\tilde{x}(LP_{KP}^8)$ is:

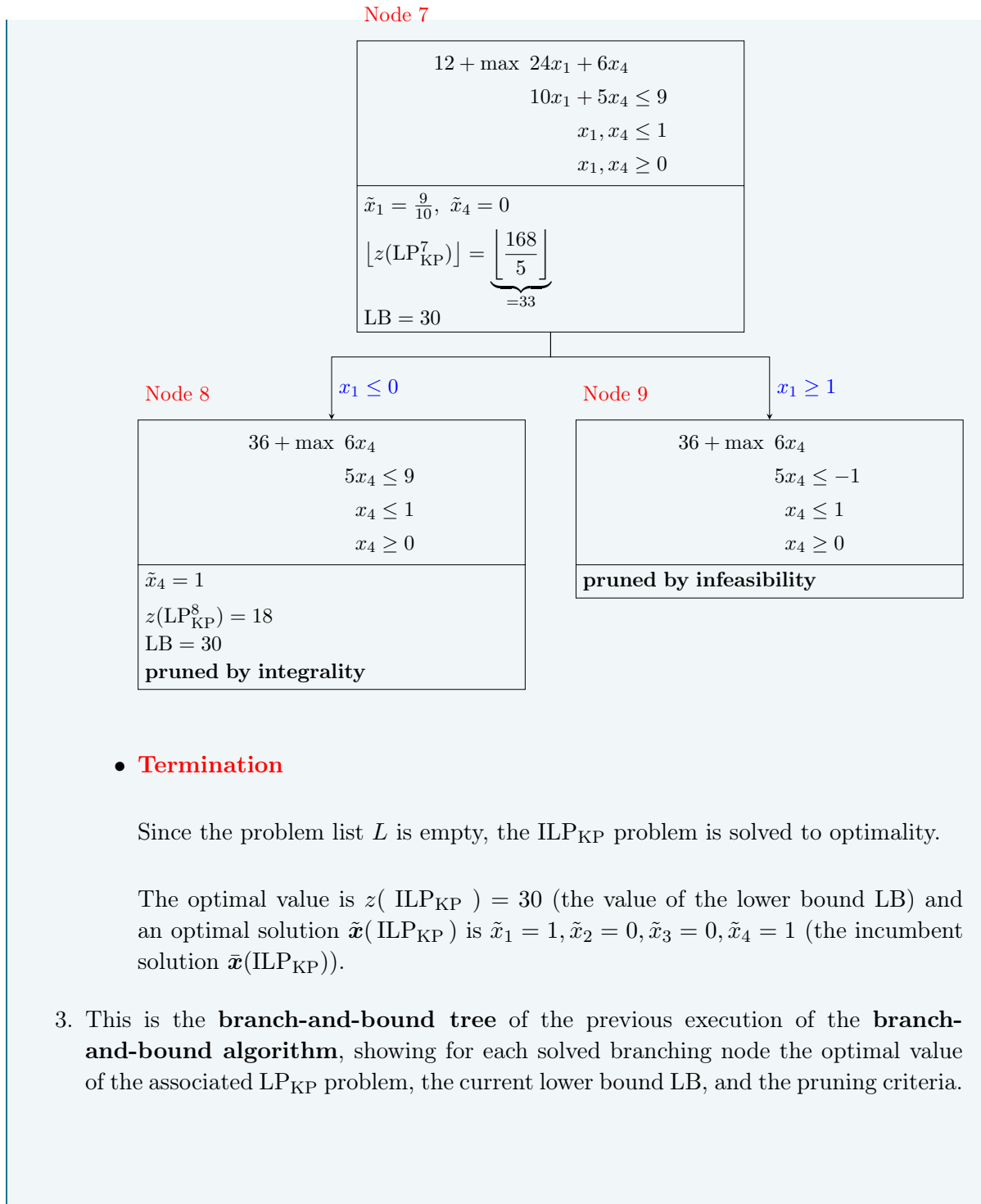
$$\tilde{x}_4 = 1 \quad \text{and} \quad z(LP_{KP}^8) = 12 + 6 \cdot 1 = 18$$

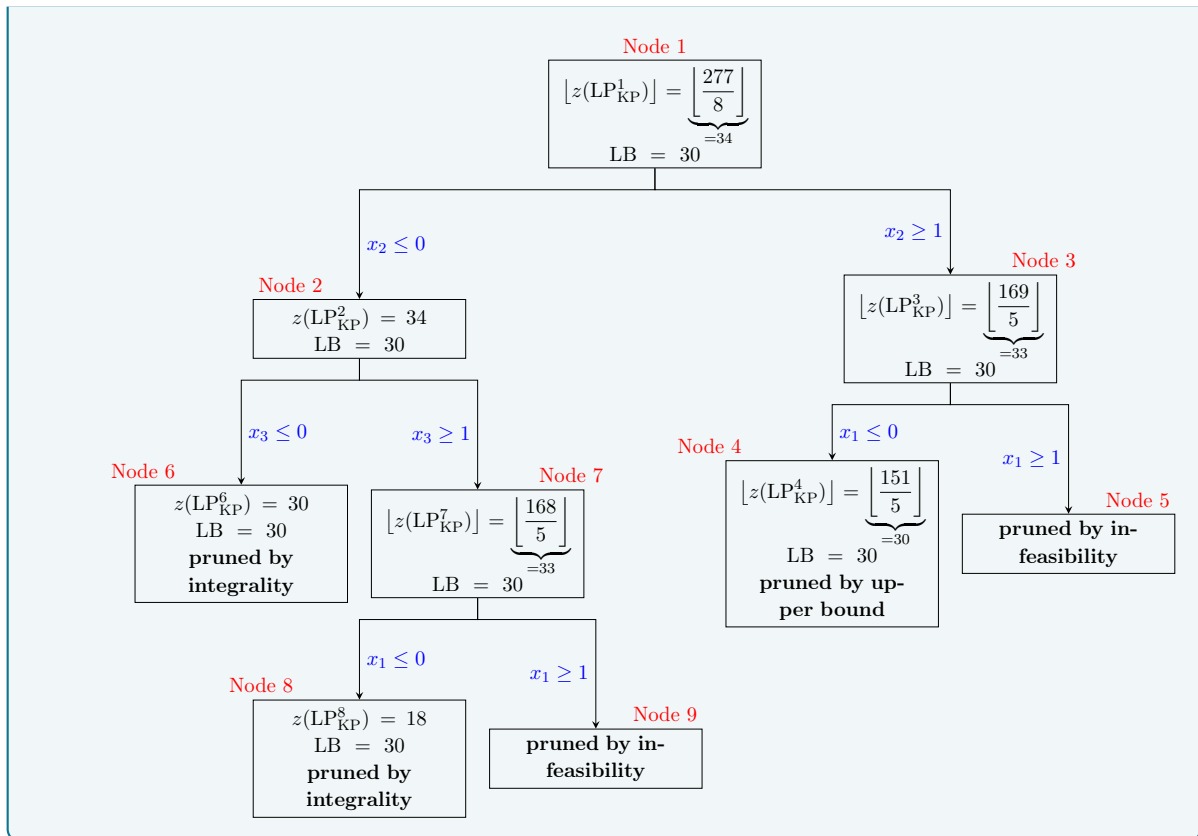
The LP_{KP}^8 problem is **feasible** and its optimal solution $\tilde{x}(LP_{KP}^8)$ is **integer**. It does not improve the incumbent solution, since $z(LP_{KP}^8) = 18 < 30 = LB$, accordingly, the branching node is pruned by integrality (when both criteria hold, the integrality comes first).

(b) Solution of the LP_{KP}^9 problem

The LP_{KP}^9 problem is **infeasible**, accordingly the branching node is pruned by infeasibility.

At the end of the fourth iteration, Node 8 is pruned by integrality and Node 9 is pruned by infeasibility. The corresponding branch-and-bound tree is:





Exercise 41.4 — Branch-and-Bound Algorithm for the 0/1 Knapsack Problem (P4)

We consider a knapsack capacity $c = 9$ and a set $\mathcal{J} = \{1, 2, 3, 4, 5, 6\}$ of 6 items ($n = 6$) with the following profits and weights:

	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$
p_j	6	5	8	9	6	7
w_j	2	3	6	7	5	9

Questions

1. Check if the items are sorted by non-increasing efficiency otherwise sort them in this order. Write the ILP_{KP} model and the LP_{KP} model for the KP instance.
2. Execute the branch-and-bound algorithm to solve the ILP_{KP} model for the 0/1 knapsack problem (KP) using:
 - (a) the split-item rule as branching-variable selection strategy;
 - (b) the FIFO rule as branching-node selection strategy;
 - (c) solve first the down-branching node and then the up-branching node;

(d) set the incumbent solution $\bar{x}(\text{ILP}_{\text{KP}})$ to the following heuristic solution :

$$\bar{x}_1 = 1, \quad \bar{x}_2 = 1, \quad \bar{x}_3 = 0, \quad \bar{x}_4 = 0, \quad \bar{x}_5 = 0, \quad \bar{x}_6 = 0$$

and set the lower bound LB accordingly.

3. Determine and comment the correspondent branch-and-bound tree showing the reason why the branching nodes can be pruned.

Solution

1. The items are sorted by non-increasing efficiency since we have:

$$\frac{6}{2} \geq \frac{5}{3} \geq \frac{8}{6} \geq \frac{9}{7} \geq \frac{6}{5} \geq \frac{7}{9}$$

This is the ILP_{KP} model for the KP instance:

$$\begin{aligned} \max \quad & 6x_1 + 5x_2 + 8x_3 + 9x_4 + 6x_5 + 7x_6 \\ & 2x_1 + 3x_2 + 6x_3 + 7x_4 + 5x_5 + 9x_6 \leq 9 \\ & x_1, x_2, x_3, x_4, x_5, x_6 \in \{0, 1\} \end{aligned}$$

This is the LP_{KP} model for the KP instance:

$$\begin{aligned} \max \quad & 6x_1 + 5x_2 + 8x_3 + 9x_4 + 6x_5 + 7x_6 \\ & 2x_1 + 3x_2 + 6x_3 + 7x_4 + 5x_5 + 9x_6 \leq 9 \\ & x_1 \leq 1 \\ & \quad x_2 \leq 1 \\ & \quad \quad x_3 \leq 1 \\ & \quad \quad \quad x_4 \leq 1 \\ & \quad \quad \quad \quad x_5 \leq 1 \\ & \quad \quad \quad \quad \quad x_6 \leq 1 \\ & x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \end{aligned}$$

2. Execution of the branch-and-bound algorithm to solve the ILP_{KP} model.

- **Initialization**

For the LP_{KP}^1 problem, the split item is $s = 3$ and the residual capacity is $rc = 4$. An optimal solution $\tilde{\mathbf{x}}(LP_{KP}^1)$ is:

$$\tilde{x}_1 = 1, \quad \tilde{x}_2 = 1, \quad \tilde{x}_3 = \frac{2}{3}, \quad \tilde{x}_4 = 0, \quad \tilde{x}_5 = 0, \quad \tilde{x}_6 = 0$$

the optimal value is:

$$z(LP_{KP}^1) = 6 \cdot 1 + 5 \cdot 1 + 8 \cdot \frac{2}{3} = \frac{49}{3}$$

The optimal solution $\tilde{\mathbf{x}}(LP_{KP}^1)$ is **fractional** and the LP_{KP}^1 problem is **feasible**, so the LP_{KP}^1 problem is inserted in the problem list L and the lower bound LB is set to 11 ($LB(\tilde{\mathbf{x}}) = 11$).

Due to the current lower bound LB and upper bound UB , we have:

$$11 \leq z(ILP_{KP}) \leq \underbrace{\left\lfloor \frac{49}{3} \right\rfloor}_{=16}$$

• **Iteration 1**

The LP_{KP}^1 problem is extracted and deleted from the problem list L .

The branching variable is x_3 . The LP_{KP}^2 and LP_{KP}^3 problems are created by unselecting ($x_3 \leq 0$) and selecting ($x_3 \geq 1$) item 3.

(a) Solution of the LP_{KP}^2 problem

We have

$$\mathcal{J}^2 = \{1, 2, 4, 5, 6\}, \quad \mathcal{J}_{out}^2 = \{3\}, \quad \mathcal{J}_{in}^2 = \emptyset$$

The capacity is $c(2) = 9$ and the objective-function constant is $p(2) = 0$. The split item is $s = 4$ and the residual capacity is $rc = 4$. An optimal solution $\tilde{\mathbf{x}}(LP_{KP}^2)$ is:

$$\tilde{x}_1 = 1, \quad \tilde{x}_2 = 1, \quad \tilde{x}_4 = \frac{4}{7}, \quad \tilde{x}_5 = 0, \quad \tilde{x}_6 = 0 \quad \text{and}$$

$$z(LP_{KP}^2) = 6 \cdot 1 + 5 \cdot 1 + 9 \cdot \frac{4}{7} = \frac{113}{7}$$

The LP_{KP}^2 problem is **feasible** and we have:

$$\lfloor z(LP_{KP}^2) \rfloor > LB \quad \text{since} \quad \underbrace{\left\lfloor \frac{113}{7} \right\rfloor}_{=16} > 11$$

The optimal solution $\tilde{\mathbf{x}}(LP_{KP}^2)$ is **fractional**, accordingly, the LP_{KP}^2 problem is **inserted in the problem list** L .

(b) Solution of the LP_{KP}^3 problem

We have

$$\mathcal{J}^3 = \{1, 2, 4, 5, 6\}, \quad \mathcal{J}_{out}^3 = \emptyset, \quad \mathcal{J}_{in}^3 = \{3\}$$

The capacity is $c(3) = 3$ and the objective-function constant is $p(3) = 8$. The split item is $s = 2$ and the residual capacity is $rc = 1$. An optimal solution $\tilde{x}(LP_{KP}^3)$ is:

$$\tilde{x}_1 = 1, \quad \tilde{x}_2 = \frac{1}{3}, \quad \tilde{x}_4 = 0, \quad \tilde{x}_5 = 0, \quad \tilde{x}_6 = 0 \quad \text{and}$$

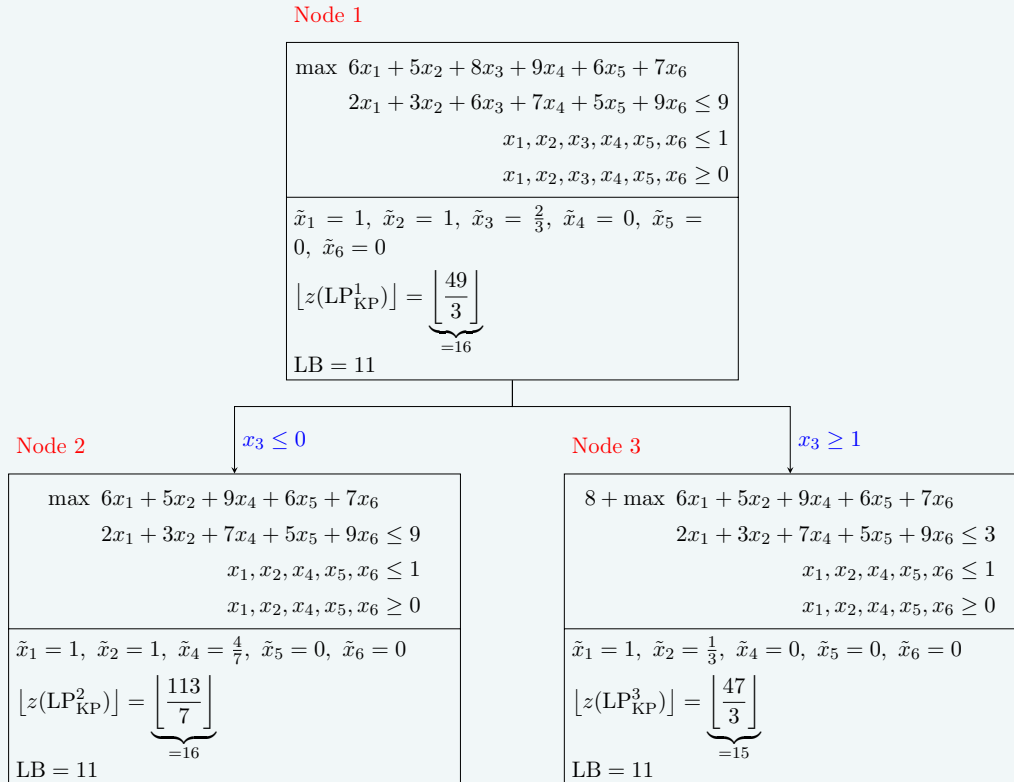
$$z(LP_{KP}^3) = 8 + 6 \cdot 1 + 5 \cdot \frac{1}{3} = \frac{47}{3}$$

The LP_{KP}^3 problem is **feasible** and we have:

$$\lfloor z(LP_{KP}^3) \rfloor > LB \quad \text{since} \quad \underbrace{\left\lfloor \frac{47}{3} \right\rfloor}_{=15} > 11$$

The optimal solution $\tilde{x}(LP_{KP}^3)$ is **fractional**, accordingly, the LP_{KP}^3 problem is **inserted in the problem list L** .

At the end of the first iteration, Node 2 and Node 3 are both inserted in the problem list L . The corresponding branch-and-bound tree is:



Due to the current lower bound LB and upper bound UB, we have:

$$11 \leq z(\text{ILP}_{\text{KP}}) \leq \underbrace{\max \left\{ \underbrace{\left\lfloor \frac{113}{7} \right\rfloor}_{=16}, \underbrace{\left\lfloor \frac{47}{3} \right\rfloor}_{=15} \right\}}_{=16}$$

• **Iteration 2**

The LP_{KP}^2 problem is extracted and deleted from the problem list L .

The branching variable is x_4 . The LP_{KP}^4 and LP_{KP}^5 problems are created by unselecting ($x_4 \leq 0$) and selecting ($x_4 \geq 1$) item 4.

(a) Solution of the LP_{KP}^4 problem

We have

$$\mathcal{J}^4 = \{1, 2, 5, 6\}, \quad \mathcal{J}_{\text{out}}^4 = \{3, 4\}, \quad \mathcal{J}_{\text{in}}^4 = \emptyset$$

The capacity is $c(4) = 9$ and the objective-function constant is $p(4) = 0$. The split item is $s = 5$ and the residual capacity is $rc = 4$. An optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^4)$ is:

$$\tilde{x}_1 = 1, \quad \tilde{x}_2 = 1, \quad \tilde{x}_5 = \frac{4}{5}, \quad \tilde{x}_6 = 0 \quad \text{and} \quad z(\text{LP}_{\text{KP}}^4) = 6 \cdot 1 + 5 \cdot 1 + 6 \cdot \frac{4}{5} = \frac{79}{5}$$

The LP_{KP}^4 problem is **feasible** and we have:

$$\lfloor z(\text{LP}_{\text{KP}}^4) \rfloor > \text{LB} \quad \text{since} \quad \underbrace{\left\lfloor \frac{79}{5} \right\rfloor}_{=15} > 11$$

The optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^4)$ is **fractional**, accordingly, the LP_{KP}^4 problem is **inserted in the problem list** L .

(b) Solution of the LP_{KP}^5 problem

We have

$$\mathcal{J}^5 = \{1, 2, 5, 6\}, \quad \mathcal{J}_{\text{out}}^5 = \{3\}, \quad \mathcal{J}_{\text{in}}^5 = \{4\}$$

The capacity is $c(5) = 2$ and the objective-function constant is $p(5) = 9$. The split item is $s = 2$ and the residual capacity is $rc = 0$. An optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^5)$ is:

$$\tilde{x}_1 = 1, \quad \tilde{x}_2 = 0, \quad \tilde{x}_5 = 0, \quad \tilde{x}_6 = 0 \quad \text{and} \quad z(\text{LP}_{\text{KP}}^5) = 9 + 6 \cdot 1 = 15$$

The LP_{KP}^5 problem is **feasible** and we have:

$$\lfloor z(\text{LP}_{\text{KP}}^5) \rfloor > \text{LB} \quad \text{since} \quad 15 > 11$$

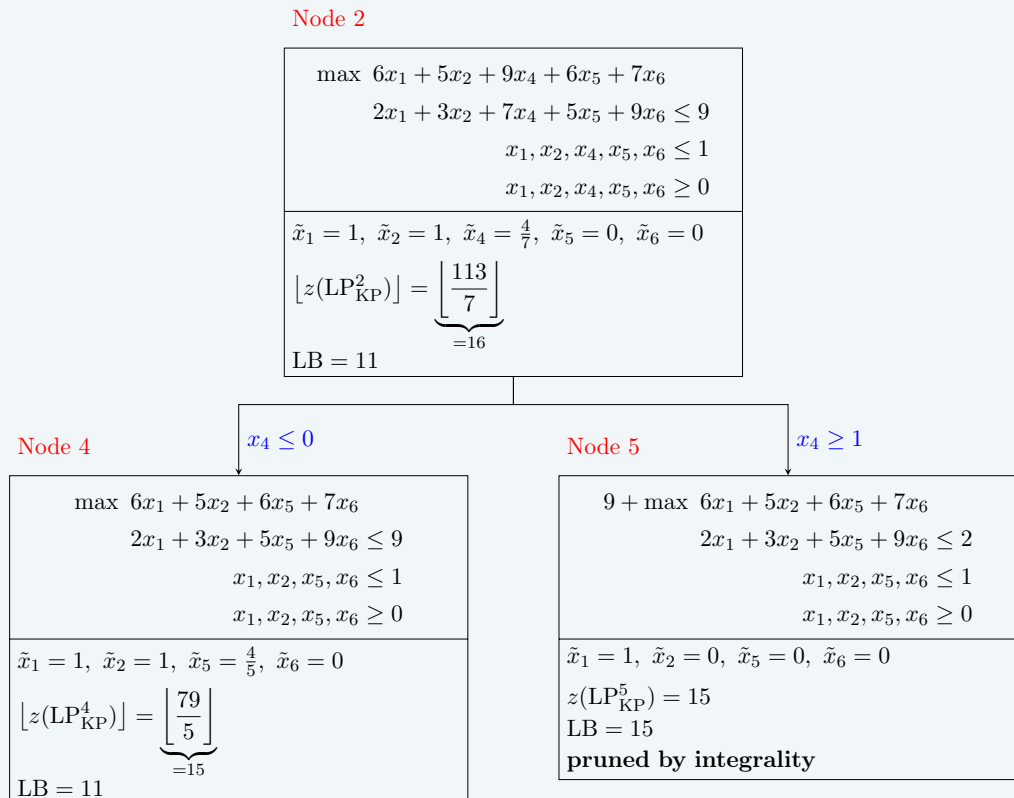
The optimal solution $\tilde{\mathbf{x}}(\text{LP}_{\text{KP}}^5)$ is **integer**, accordingly, the lower bound LB becomes 15 and the incumbent solution $\bar{\mathbf{x}}(\text{ILP}_{\text{KP}})$ becomes $\bar{x}_1 = 1, \bar{x}_2 = 0, \bar{x}_3 = 0, \bar{x}_4 = 1, \bar{x}_5 = 0, \bar{x}_6 = 0$.

Accordingly, the branching node is pruned by integrality.

Since the lower bound is improved, the LP_{KP}^3 problem and the LP_{KP}^4 problem are pruned by upper bound and removed from the problem list L . We have:

$$\begin{aligned} \lfloor z(\text{LP}_{\text{KP}}^3) \rfloor &\leq \text{LB} & \text{since} & \underbrace{\left\lfloor \frac{47}{3} \right\rfloor}_{=15} \leq 15 \\ \lfloor z(\text{LP}_{\text{KP}}^4) \rfloor &\leq \text{LB} & \text{since} & \underbrace{\left\lfloor \frac{79}{5} \right\rfloor}_{=15} \leq 15 \end{aligned}$$

At the end of the second iteration, Node 5 improves the lower bound to $\text{LB} = 15$ and is pruned by integrality; Node 3 and Node 4 are then pruned by upper bound. The corresponding branch-and-bound tree is:

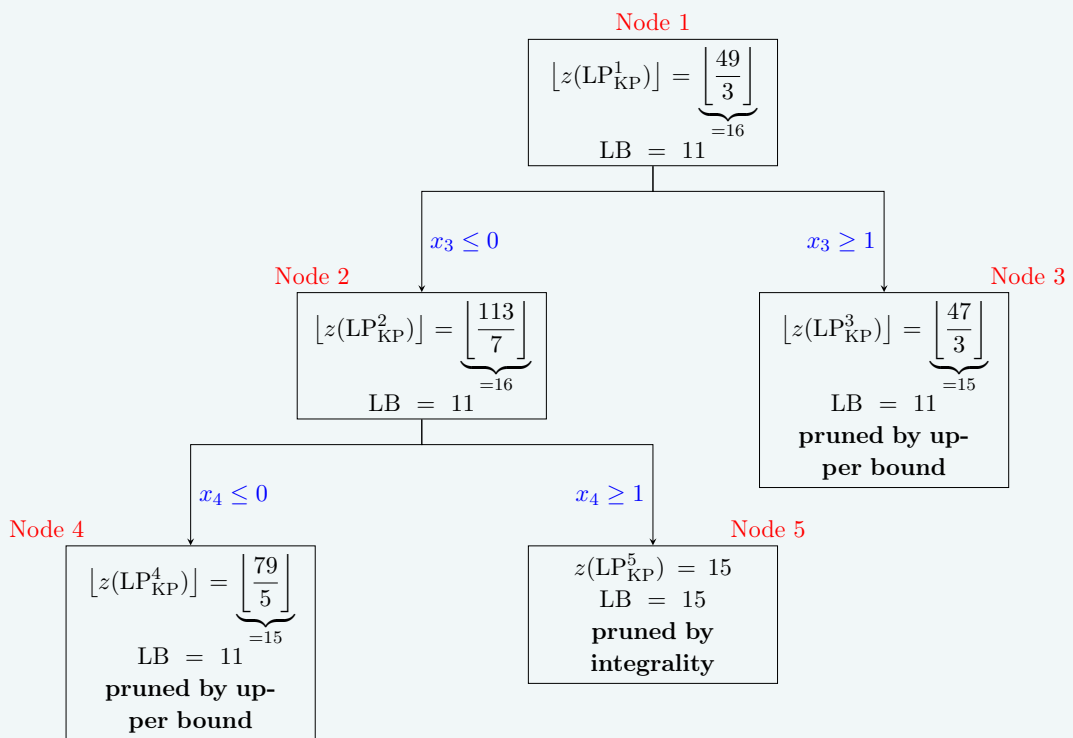


- **Termination**

Since the problem list L is empty, the ILP_{KP} problem is solved to optimality.

The optimal value is $z(\text{ILP}_{\text{KP}}) = 15$ (the value of the lower bound LB) and an optimal solution $\tilde{\mathbf{x}}(\text{ILP}_{\text{KP}})$ is $\tilde{x}_1 = 1, \tilde{x}_2 = 0, \tilde{x}_3 = 0, \tilde{x}_4 = 1, \tilde{x}_5 = 0, \tilde{x}_6 = 0$ (the incumbent solution $\bar{\mathbf{x}}(\text{ILP}_{\text{KP}})$).

3. This is the **branch-and-bound tree** of the previous execution of the **branch-and-bound algorithm**, showing for each solved branching node the optimal value of the associated LP_{KP} problem, the current lower bound LB, and the pruning criteria.



Coloring problems

Vertex coloring: the DSATUR branch-and-bound

Rules used in these exercises

The DSATUR branch-and-bound algorithm of the chapter [Vertex coloring: the DSATUR branch-and-bound](#). At a node k , with $q(k)$ colors used, colored vertices V^k and uncolored vertices U^k :

$$\begin{aligned} sat_k(v) &= |\{col^k(u) : u \in N(v) \cap V^k\}|, & du_k(v) &= |N(v) \cap U^k|, \\ F_k(v) &= \{c \leq q(k) : \text{no neighbour of } v \text{ has color } c\}. \end{aligned} \quad (D1)$$

- (D2) **Bounds.** UB = colors of the *DSATUR* heuristic; K = greedy clique (vertices by non-increasing degree, ties by smallest index; a vertex enters if adjacent to all of K), LB = $|K|$, and K 's i -th vertex is pre-colored with color i .
- (D3) **Branching.** The vertex $v \in U^k$ of maximum $sat_k(v)$, then maximum $du_k(v)$, then smallest index; one child for each $c \in F_k(v)$ and one for the new color $q(k) + 1$, in increasing order of color.
- (D4) **Pruning.** A node with $q \geq \text{UB}$ is pruned by bound. A complete coloring with $q < \text{UB}$ is a new incumbent: $\text{UB} \leftarrow q$, the nodes of the list with $q \geq \text{UB}$ are removed, and the algorithm stops if $\text{UB} = \text{LB}$.
- (D5) **Search.** Depth first: the list is a stack and the child with the smallest color is extracted first; the nodes are numbered in the order of evaluation.

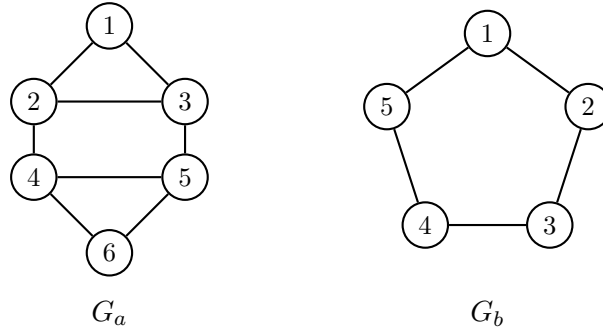
Exercise 42.1 — The root node of the DSATUR branch-and-bound (P1)

We consider two graphs. The graph G_a of an exam calendar, with $V = \{1, \dots, 6\}$ and the edges

$$E_a = \{\{1, 2\}, \{1, 3\}, \{2, 3\}, \{2, 4\}, \{3, 5\}, \{4, 5\}, \{4, 6\}, \{5, 6\}\},$$

and the cycle $G_b = C_5$, with $V = \{1, \dots, 5\}$ and the edges

$$E_b = \{\{1, 2\}, \{1, 5\}, \{2, 3\}, \{3, 4\}, \{4, 5\}\}.$$



Questions

1. Run the *DSATUR* heuristic on G_a and give the upper bound UB.
2. Find the greedy clique of G_a and the lower bound LB. Does the *DSATUR* branch-and-bound algorithm branch?
3. Answer the same two questions for G_b .
4. For G_b , write the root node and execute the first iteration of the algorithm: the branching vertex, its feasible colors and the children.

Solution

We use the rules (D1)–(D5) of the box at the beginning of the chapter.

1. The steps of *DSATUR* on G_a are:

step	1	2	3	4	5	6
v	2	3	1	4	5	6
$sat(v)$	0	1	2	1	1	2
$du(v)$	3	2	0	2	1	0
$col(v)$	1	2	3	2	1	3

At step 3 vertex 1 sees the colors 1 and 2, so it is chosen before the vertices with $sat = 1$. *DSATUR* gives $\{\{2, 5\}, \{3, 4\}, \{1, 6\}\}$, hence $UB = 3$.

2. The degrees are $\deg(1) = 2$, $\deg(2) = 3$, $\deg(3) = 3$, $\deg(4) = 3$, $\deg(5) = 3$, $\deg(6) = 2$, so the vertices are scanned in the order 2, 3, 4, 5, 1, 6. The vertices 2, 3, 1 enter, in this order; every other vertex misses an edge to a vertex already in K . Hence $K = \{1, 2, 3\}$ (in the order of entry: $k_1 = 2$, $k_2 = 3$, $k_3 = 1$) and $LB = 3$. Since $LB = UB = 3$, the coloring of *DSATUR* is optimal and the algorithm stops in Step 0, without branching: $\chi(G_a) = 3$.
3. The steps of *DSATUR* on G_b are:

step	1	2	3	4	5
v	1	2	3	4	5
$sat(v)$	0	1	1	1	2
$du(v)$	2	1	1	1	0
$col(v)$	1	2	1	2	3

$DSATUR$ gives $\{\{1, 3\}, \{2, 4\}, \{5\}\}$, hence $UB = 3$. All the degrees are equal, so the vertices are scanned in the order 1, 2, 3, 4, 5. The vertices 1, 2 enter, in this order; every other vertex misses an edge to a vertex already in K . Hence $K = \{1, 2\}$ (in the order of entry: $k_1 = 1, k_2 = 2$) and $LB = 2$. Now $LB = 2 < 3 = UB$: the algorithm branches.

4. Node 1 colors $col^1(1) = 1, col^1(2) = 2$, with $q(1) = 2$. The vertices 3, 5 have $sat = 1$ and the same $du = 1$: the smallest index wins, and the branching vertex is 3. Its neighbour 2 has color 2, so $F_1(3) = \{1\}$.

- Node 2: $col(3) = 1$, with $q(2) = 2 < UB = 3$: the node survives.
- Node 3: $col(3) = 3$, the new color, with $q(3) = 3 \geq UB = 3$: pruned by bound.

The search continues from node 2: it is the same search as on the outer cycle of the Petersen graph, and it ends with 6 nodes and $\chi(G_b) = 3$.

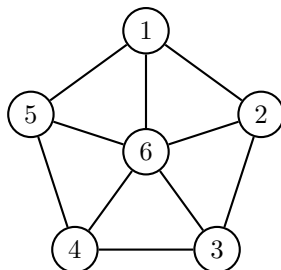
On G_a the clique $\{2, 3, 1\}$ meets the $DSATUR$ bound: no branching. On G_b the clique bound is 2 and the tree is needed to prove $\chi(C_5) = 3$.

Exercise 42.2 — Proving a lower bound: the wheel W_5 (P2)

We consider the wheel W_5 (a cycle on $1, \dots, 5$ and the hub 6), the graph $G = (V, E)$ with $V = \{1, \dots, 6\}$ and the edges

$$E = \{\{1, 2\}, \{1, 5\}, \{1, 6\}, \{2, 3\}, \{2, 6\}, \{3, 4\}, \{3, 6\}, \{4, 5\}, \{4, 6\}, \{5, 6\}\},$$

drawn below:



Questions

1. Run the $DSATUR$ heuristic: give the order of the vertices, the coloring and the upper bound UB .

2. Find the greedy clique K , the lower bound LB and the root node of the DSATUR branch-and-bound algorithm.
3. Execute the DSATUR branch-and-bound algorithm, iteration by iteration.
4. Draw the branch-and-bound tree and give the chromatic number of G with an optimal coloring.

Solution

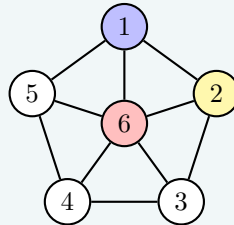
We use the rules (D1)–(D5) of the box at the beginning of the chapter.

1. The vertex chosen at each step, its saturation degree, its degree among the uncolored vertices and its color are:

step	1	2	3	4	5	6
v	6	1	2	3	4	5
$sat(v)$	0	1	2	2	2	3
$du(v)$	5	2	1	1	1	0
$col(v)$	1	2	3	2	3	4

DSATUR gives $\{\{6\}, \{1, 3\}, \{2, 4\}, \{5\}\}$, hence $UB = 4$.

2. The degrees are $\deg(1) = 3$, $\deg(2) = 3$, $\deg(3) = 3$, $\deg(4) = 3$, $\deg(5) = 3$, $\deg(6) = 5$, so the vertices are scanned in the order 6, 1, 2, 3, 4, 5. The vertices 6, 1, 2 enter, in this order; every other vertex misses an edge to a vertex already in K . Hence $K = \{1, 2, 6\}$ (in the order of entry: $k_1 = 6$, $k_2 = 1$, $k_3 = 2$) and $LB = 3$. Since $LB = 3 < 4 = UB$, the algorithm branches. Node 1 colors $col^1(6) = 1$, $col^1(1) = 2$, $col^1(2) = 3$, with $q(1) = 3$:



3. Iteration 1

Node 1 is extracted from the list, with $Q_1 = \{6\}$, $Q_2 = \{1\}$, $Q_3 = \{2\}$ and $q(1) = 3$. The vertices 3, 5 have $sat = 2$ and the same $du = 1$: the smallest index wins, and the branching vertex is 3. Its feasible colors are $F_1(3) = \{2\}$.

- Node 2: $col(3) = 2$, with $q(2) = 3 < UB = 4$: the node survives.
- Node 3: $col(3) = 4$, the new color, with $q(3) = 4 \geq UB = 4$: pruned by bound.

Iteration 2

Node 2 is extracted from the list, with $Q_1 = \{6\}$, $Q_2 = \{1, 3\}$, $Q_3 = \{2\}$ and $q(2) = 3$. The vertices 4, 5 have $sat = 2$ and the same $du = 1$: the smallest index wins, and the branching vertex is 4. Its feasible colors are $F_2(4) = \{3\}$.

- Node 4: $col(4) = 3$, with $q(4) = 3 < UB = 4$: the node survives.
- Node 5: $col(4) = 4$, the new color, with $q(5) = 4 \geq UB = 4$: pruned by bound.

Iteration 3

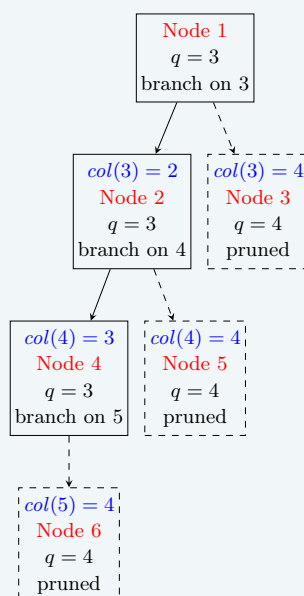
Node 4 is extracted from the list, with $Q_1 = \{6\}$, $Q_2 = \{1, 3\}$, $Q_3 = \{2, 4\}$ and $q(4) = 3$. Vertex 5 is the only one with $sat = 3$: it is the branching vertex. Its feasible colors are $F_4(5) = \emptyset$.

- Node 6: $col(5) = 4$, the new color, with $q(6) = 4 \geq UB = 4$: pruned by bound.

Termination

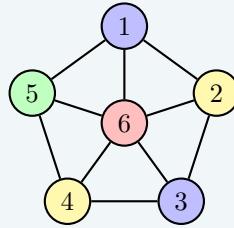
The list is empty: the incumbent solution of *DSATUR* is optimal.

4. The tree has 6 nodes; the pruned nodes are dashed.



Node	Parent	Child	q	UB	Branching vertex	Outcome
1	–	root	3	4	3	branched
2	1	$col(3) = 2$	3	4	4	branched
3	1	$col(3) = 4$	4	4	–	pruned by bound
4	2	$col(4) = 3$	3	4	5	branched
5	2	$col(4) = 4$	4	4	–	pruned by bound
6	4	$col(5) = 4$	4	4	–	pruned by bound

Hence $\chi(G) = 4$, with the optimal coloring $\{\{6\}, \{1, 3\}, \{2, 4\}, \{5\}\}$:



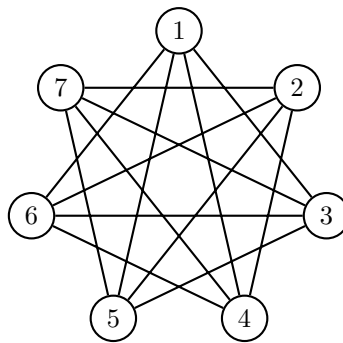
The clique bound gives only 3: the tree proves that the odd cycle and the hub need a fourth color.

Exercise 42.3 — The complement of the cycle C_7 (P3)

We consider the complement of the cycle C_7 (two vertices are adjacent if they are not consecutive on the cycle $1, 2, \dots, 7$), the graph $G = (V, E)$ with $V = \{1, \dots, 7\}$ and the edges

$$E = \{\{1, 3\}, \{1, 4\}, \{1, 5\}, \{1, 6\}, \{2, 4\}, \{2, 5\}, \{2, 6\}, \\ \{2, 7\}, \{3, 5\}, \{3, 6\}, \{3, 7\}, \{4, 6\}, \{4, 7\}, \{5, 7\}\},$$

drawn below:



Questions

1. Run the *DSATUR* heuristic: give the order of the vertices, the coloring and the upper bound UB.
2. Find the greedy clique K , the lower bound LB and the root node of the DSATUR branch-and-bound algorithm.
3. Execute the DSATUR branch-and-bound algorithm, iteration by iteration.
4. Draw the branch-and-bound tree and give the chromatic number of G with an optimal coloring.

Solution

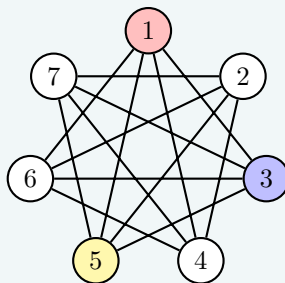
We use the rules (D1)–(D5) of the box at the beginning of the chapter.

1. The vertex chosen at each step, its saturation degree, its degree among the uncolored vertices and its color are:

step	1	2	3	4	5	6	7
v	1	3	5	6	4	2	7
$sat(v)$	0	1	2	2	2	2	3
$du(v)$	4	3	2	2	2	1	0
$col(v)$	1	2	3	3	2	1	4

$DSATUR$ gives $\{\{1, 2\}, \{3, 4\}, \{5, 6\}, \{7\}\}$, hence $UB = 4$.

- All the degrees are equal, so the vertices are scanned in the order 1, 2, 3, 4, 5, 6, 7. The vertices 1, 3, 5 enter, in this order; every other vertex misses an edge to a vertex already in K . Hence $K = \{1, 3, 5\}$ (in the order of entry: $k_1 = 1, k_2 = 3, k_3 = 5$) and $LB = 3$. Since $LB = 3 < 4 = UB$, the algorithm branches. Node 1 colors $col^1(1) = 1, col^1(3) = 2, col^1(5) = 3$, with $q(1) = 3$:



3. Iteration 1

Node 1 is extracted from the list, with $Q_1 = \{1\}, Q_2 = \{3\}, Q_3 = \{5\}$ and $q(1) = 3$. The vertices 6, 7 have $sat = 2$ and the same $du = 2$: the smallest index wins, and the branching vertex is 6. Its feasible colors are $F_1(6) = \{3\}$.

- Node 2: $col(6) = 3$, with $q(2) = 3 < UB = 4$: the node survives.
- Node 3: $col(6) = 4$, the new color, with $q(3) = 4 \geq UB = 4$: pruned by bound.

Iteration 2

Node 2 is extracted from the list, with $Q_1 = \{1\}, Q_2 = \{3\}, Q_3 = \{5, 6\}$ and $q(2) = 3$. The vertices 4, 7 have $sat = 2$ and the same $du = 2$: the smallest index wins, and the branching vertex is 4. Its feasible colors are $F_2(4) = \{2\}$.

- Node 4: $col(4) = 2$, with $q(4) = 3 < UB = 4$: the node survives.
- Node 5: $col(4) = 4$, the new color, with $q(5) = 4 \geq UB = 4$: pruned by bound.

Iteration 3

Node 4 is extracted from the list, with $Q_1 = \{1\}, Q_2 = \{3, 4\}, Q_3 = \{5, 6\}$ and $q(4) = 3$. The vertices 2, 7 have $sat = 2$ and the same $du = 1$: the smallest index wins, and the branching vertex is 2. Its feasible colors are $F_4(2) = \{1\}$.

- Node 6: $col(2) = 1$, with $q(6) = 3 < UB = 4$: the node survives.

- Node 7: $col(2) = 4$, the new color, with $q(7) = 4 \geq UB = 4$: pruned by bound.

Iteration 4

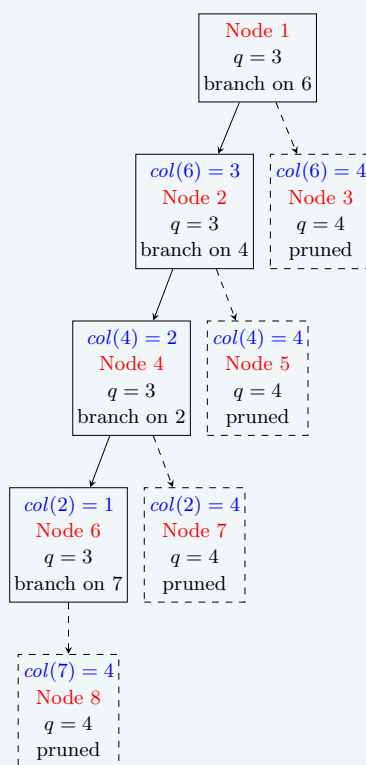
Node 6 is extracted from the list, with $Q_1 = \{1, 2\}$, $Q_2 = \{3, 4\}$, $Q_3 = \{5, 6\}$ and $q(6) = 3$. Vertex 7 is the only one with $sat = 3$: it is the branching vertex. Its feasible colors are $F_6(7) = \emptyset$.

- Node 8: $col(7) = 4$, the new color, with $q(8) = 4 \geq UB = 4$: pruned by bound.

Termination

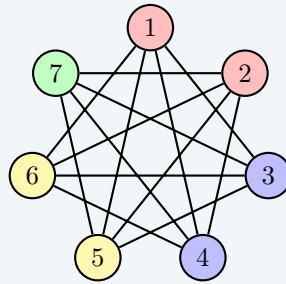
The list is empty: the incumbent solution of *DSATUR* is optimal.

- The tree has 8 nodes; the pruned nodes are dashed.



Node	Parent	Child	q	UB	Branching vertex	Outcome
1	—	root	3	4	6	branched
2	1	$col(6) = 3$	3	4	4	branched
3	1	$col(6) = 4$	4	4	—	pruned by bound
4	2	$col(4) = 2$	3	4	2	branched
5	2	$col(4) = 4$	4	4	—	pruned by bound
6	4	$col(2) = 1$	3	4	7	branched
7	4	$col(2) = 4$	4	4	—	pruned by bound
8	6	$col(7) = 4$	4	4	—	pruned by bound

Hence $\chi(G) = 4$, with the optimal coloring $\{\{1, 2\}, \{3, 4\}, \{5, 6\}, \{7\}\}$:



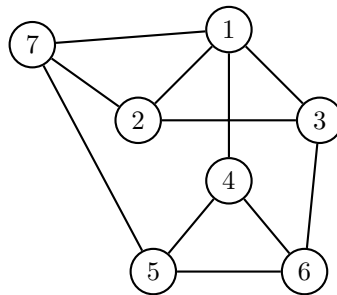
Every stable set of G has at most 2 vertices, so the stable set bound $\lceil 7/2 \rceil = 4$ of Chapter 4 confirms the result.

Exercise 42.4 — Improving the incumbent solution (P4)

We consider a graph with 7 vertices and 11 edges, the graph $G = (V, E)$ with $V = \{1, \dots, 7\}$ and the edges

$$E = \{\{1, 2\}, \{1, 3\}, \{1, 4\}, \{1, 7\}, \{2, 3\}, \{2, 7\}, \\ \{3, 6\}, \{4, 5\}, \{4, 6\}, \{5, 6\}, \{5, 7\}\},$$

drawn below:



Questions

1. Run the *DSATUR* heuristic: give the order of the vertices, the coloring and the upper bound UB.
2. Find the greedy clique K , the lower bound LB and the root node of the DSATUR branch-and-bound algorithm.
3. Execute the DSATUR branch-and-bound algorithm, iteration by iteration.
4. Draw the branch-and-bound tree and give the chromatic number of G with an optimal coloring.

Solution

We use the rules (D1)–(D5) of the box at the beginning of the chapter.

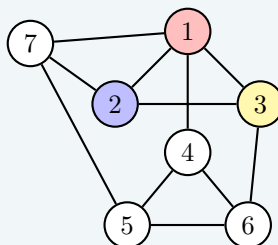
1. The vertex chosen at each step, its saturation degree, its degree among the uncolored

vertices and its color are:

step	1	2	3	4	5	6	7
v	1	2	3	7	4	5	6
$sat(v)$	0	1	2	2	1	2	3
$du(v)$	4	2	1	1	2	1	0
$col(v)$	1	2	3	3	2	1	4

$DSATUR$ gives $\{\{1, 5\}, \{2, 4\}, \{3, 7\}, \{6\}\}$, hence $UB = 4$.

- The degrees are $\deg(1) = 4$, $\deg(2) = 3$, $\deg(3) = 3$, $\deg(4) = 3$, $\deg(5) = 3$, $\deg(6) = 3$, $\deg(7) = 3$, so the vertices are scanned in the order 1, 2, 3, 4, 5, 6, 7. The vertices 1, 2, 3 enter, in this order; every other vertex misses an edge to a vertex already in K . Hence $K = \{1, 2, 3\}$ (in the order of entry: $k_1 = 1$, $k_2 = 2$, $k_3 = 3$) and $LB = 3$. Since $LB = 3 < 4 = UB$, the algorithm branches. Node 1 colors $col^1(1) = 1$, $col^1(2) = 2$, $col^1(3) = 3$, with $q(1) = 3$:



3. Iteration 1

Node 1 is extracted from the list, with $Q_1 = \{1\}$, $Q_2 = \{2\}$, $Q_3 = \{3\}$ and $q(1) = 3$. Vertex 7 is the only one with $sat = 2$: it is the branching vertex. Its feasible colors are $F_1(7) = \{3\}$.

- Node 2: $col(7) = 3$, with $q(2) = 3 < UB = 4$: the node survives.
- Node 3: $col(7) = 4$, the new color, with $q(3) = 4 \geq UB = 4$: pruned by bound.

Iteration 2

Node 2 is extracted from the list, with $Q_1 = \{1\}$, $Q_2 = \{2\}$, $Q_3 = \{3, 7\}$ and $q(2) = 3$. The vertices 4, 5, 6 have $sat = 1$ and the same $du = 2$: the smallest index wins, and the branching vertex is 4. Its feasible colors are $F_2(4) = \{2, 3\}$.

- Node 4: $col(4) = 2$, with $q(4) = 3 < UB = 4$: the node survives.
- Node 5: $col(4) = 3$, with $q(5) = 3 < UB = 4$: the node survives.
- Node 6: $col(4) = 4$, the new color, with $q(6) = 4 \geq UB = 4$: pruned by bound.

Iteration 3

Node 4 is extracted from the list, with $Q_1 = \{1\}$, $Q_2 = \{2, 4\}$, $Q_3 = \{3, 7\}$ and $q(4) = 3$. The vertices 5, 6 have $sat = 2$ and the same $du = 1$: the smallest index wins, and the branching vertex is 5. Its feasible colors are $F_4(5) = \{1\}$.

- Node 7: $col(5) = 1$, with $q(7) = 3 < UB = 4$: the node survives.
- Node 8: $col(5) = 4$, the new color, with $q(8) = 4 \geq UB = 4$: pruned by bound.

Iteration 4

Node 7 is extracted from the list, with $Q_1 = \{1, 5\}$, $Q_2 = \{2, 4\}$, $Q_3 = \{3, 7\}$ and $q(7) = 3$. Vertex 6 is the only one with $sat = 3$: it is the branching vertex. Its feasible colors are $F_7(6) = \emptyset$.

- Node 9: $col(6) = 4$, the new color, with $q(9) = 4 \geq UB = 4$: pruned by bound.

Iteration 5

Node 5 is extracted from the list, with $Q_1 = \{1\}$, $Q_2 = \{2\}$, $Q_3 = \{3, 4, 7\}$ and $q(5) = 3$. The vertices 5, 6 have $sat = 1$ and the same $du = 1$: the smallest index wins, and the branching vertex is 5. Its feasible colors are $F_5(5) = \{1, 2\}$.

- Node 10: $col(5) = 1$, with $q(10) = 3 < UB = 4$: the node survives.
- Node 11: $col(5) = 2$, with $q(11) = 3 < UB = 4$: the node survives.
- Node 12: $col(5) = 4$, the new color, with $q(12) = 4 \geq UB = 4$: pruned by bound.

Iteration 6

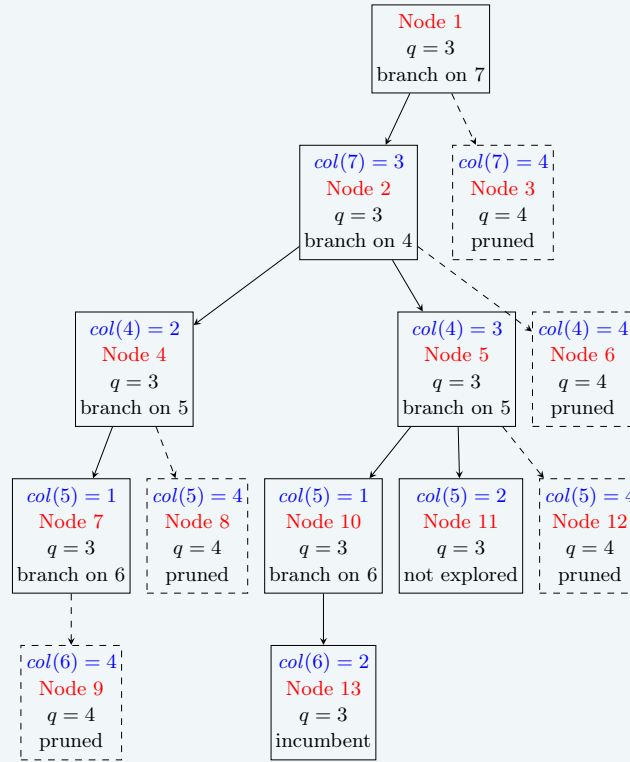
Node 10 is extracted from the list, with $Q_1 = \{1, 5\}$, $Q_2 = \{2\}$, $Q_3 = \{3, 4, 7\}$ and $q(10) = 3$. Vertex 6 is the only one with $sat = 2$: it is the branching vertex. Its feasible colors are $F_{10}(6) = \{2\}$.

- Node 13: $col(6) = 2$, with $q(13) = 3 < UB = 4$: a complete coloring with fewer colors: new incumbent solution, with $\{\{1, 5\}, \{2, 6\}, \{3, 4, 7\}\}$ and $UB = 3$.

Termination

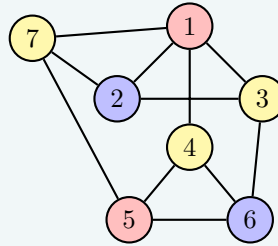
The new incumbent solution gives $UB = 3$. The nodes of the list with $q \geq UB$ are removed: node 11. Since $UB = LB = 3$, the incumbent solution is optimal: STOP.

4. The tree has 13 nodes; the pruned nodes are dashed.



Node	Parent	Child	q	UB	Branching vertex	Outcome
1	—	root	3	4	7	branched
2	1	$col(7) = 3$	3	4	4	branched
3	1	$col(7) = 4$	4	4	—	pruned by bound
4	2	$col(4) = 2$	3	4	5	branched
5	2	$col(4) = 3$	3	4	5	branched
6	2	$col(4) = 4$	4	4	—	pruned by bound
7	4	$col(5) = 1$	3	4	6	branched
8	4	$col(5) = 4$	4	4	—	pruned by bound
9	7	$col(6) = 4$	4	4	—	pruned by bound
10	5	$col(5) = 1$	3	4	6	branched
11	5	$col(5) = 2$	3	4	—	not explored
12	5	$col(5) = 4$	4	4	—	pruned by bound
13	10	$col(6) = 2$	3	3	—	new incumbent

Hence $\chi(G) = 3$, with the optimal coloring $\{\{1, 5\}, \{2, 6\}, \{3, 4, 7\}\}$:



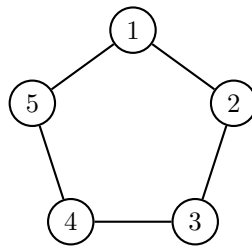
Here *DSATUR* is not optimal: the branch-and-bound algorithm improves the incumbent solution from 4 to 3 colors and stops as soon as UB meets the clique bound.

Exercise 42.5 — The no-symmetry branching (P5)

We consider the cycle C_5 , with $V = \{1, \dots, 5\}$ and the edges

$$E = \{\{1, 2\}, \{1, 5\}, \{2, 3\}, \{3, 4\}, \{4, 5\}\},$$

drawn below. Its greedy clique is $K = \{1, 2\}$, so the root node colors $col^1(1) = 1$ and $col^1(2) = 2$.



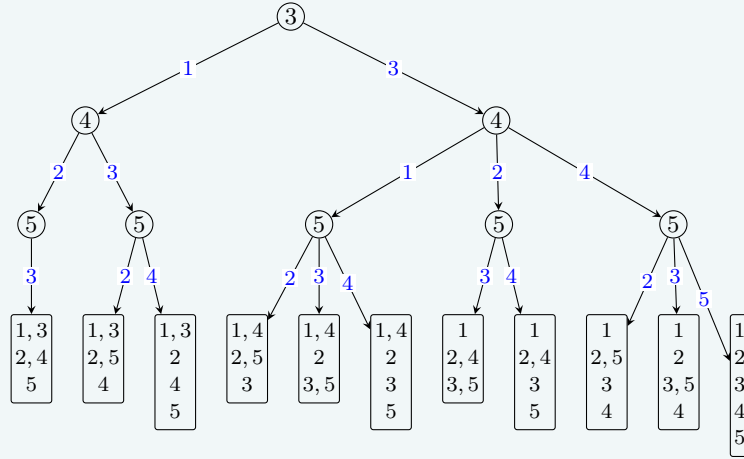
Questions

1. Draw the tree of the *DSATUR* branch-and-bound algorithm *without pruning* (no upper bound): at every node, the branching vertex and one child for each feasible color and for the new color. How many nodes and how many leaves does it have?
2. List the partitions of V given by the leaves, by number of colors. Are they all the partitions of V into stable sets?
3. How many assignments of the five colors $1, \dots, 5$ to the vertices have no edge with equal colors? Explain the difference with the number of leaves.
4. With the upper bound $UB = 3$ of *DSATUR* and the pruning rule, how many nodes does the tree have?

Solution

We use the rules (D1)–(D5) of the box at the beginning of the chapter.

1. Each internal node shows its branching vertex, each edge the color given to it (blue), each leaf its color classes, one per line:



The tree has 19 nodes and 11 leaves.

2. The leaves, by number of colors:

colors	leaves	partitions
3	5	$\{\{1\}, \{2, 4\}, \{3, 5\}\}$ $\{\{1, 3\}, \{2, 4\}, \{5\}\}$ $\{\{1, 3\}, \{2, 5\}, \{4\}\}$ $\{\{1, 4\}, \{2\}, \{3, 5\}\}$ $\{\{1, 4\}, \{2, 5\}, \{3\}\}$
4	5	$\{\{1\}, \{2\}, \{3, 5\}, \{4\}\}$ $\{\{1\}, \{2, 4\}, \{3\}, \{5\}\}$ $\{\{1\}, \{2, 5\}, \{3\}, \{4\}\}$ $\{\{1, 3\}, \{2\}, \{4\}, \{5\}\}$ $\{\{1, 4\}, \{2\}, \{3\}, \{5\}\}$
5	1	$\{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}\}$

The 11 leaves give 11 different partitions, and the enumeration of all the partitions of V into stable sets gives the same 11: each feasible solution appears exactly once, as the proposition on the no-symmetry branching states.

3. By enumeration, 1020 assignments. A partition with q classes can be named with $5 \cdot 4 \cdots (5 - q + 1)$ ordered choices of colors, so

$$5 \cdot 60 + 5 \cdot 120 + 1 \cdot 120 = 1020.$$

The labelled colorings contain each partition many times, with the colors renamed; the tree contains each one once.

4. The pruning leaves 6 nodes instead of 19: every node with $q = 3$ is pruned, and the search proves $\chi(C_5) = 3$ along a single path, as in the first exercise of this chapter.

Without pruning, the tree of C_5 has 11 leaves, one per partition into stable sets, against 1020 labelled colorings; with $UB = 3$ only 6 nodes remain.