



**MAT3007 · Homework 9**

Due: 23:59, Dec 12, 2024

**Instructions:**

- Homework problems must be carefully and clearly answered to receive full credit. Complete sentences that establish a clear logical progression are highly recommended.
- You must submit your assignment in Blackboard.
- The homework must be written in English.
- Late submission will not be graded.
- Each student **must not copy** homework solutions from another student or from any other source.

---

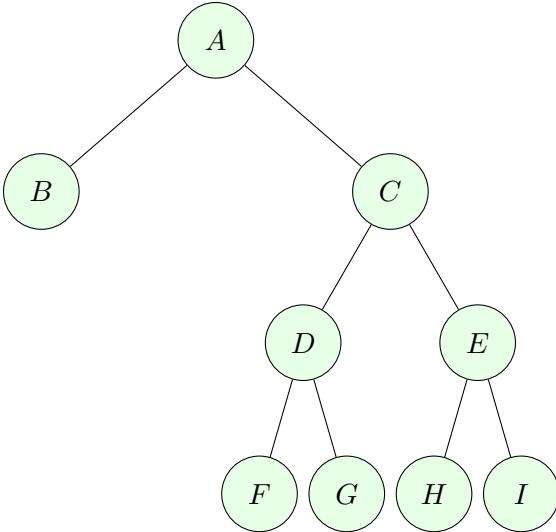
**Problem 1: True or False (20 pts).**

For each of the following statements, state whether it is true or false. Please explain your answers (if not true, please show a counterexample).

- If the linear programming relaxation of an integer programming problem is infeasible, then the integer program itself is also infeasible.
- If the linear programming relaxation of an integer programming problem is unbounded, then the integer program itself is also unbounded.
- If  $x^*$  is an optimal solution of the linear programming relaxation of an integer programming problem, and  $x^*$  satisfies the integer restrictions, then  $x^*$  is an optimal solution to the integer programming problem.
- If the optimal solution to the linear programming relaxation of integer programming problem has exactly one variable taking a non-integer value, then the branch and bound algorithm will terminate after branching once and exploring the resulting two nodes.

**Problem 2: Branch-and-Bound Tree (20 pts).**

Suppose, the branch-and-bound algorithm is being executed for an integer programming problem with **maximization** objective, and the current state of the search corresponds to the following tree:



Node	LP opt. obj.	$x^{LP}$ integer?
A	70.5	no
B	64.5	no
C	67	no
D	65.5	no
E	66	no
F	57	yes
G	60.5	no
H	Infeasible	n/a
I	61.5	yes

The letters in the nodes correspond to the order in which the LP relaxations have been solved (in alphabetical order). The table above right lists for each node the optimal objective function value for the corresponding node LP relaxation, and indicates whether the resulting solution satisfies all integer restrictions of the original integer program. Assume the original integer program has five decision variables,  $x_1, \dots, x_5$ , and requires all the decision variables to take integer values. Let  $z^*$  denote the optimal value of the integer program being solved.

- (a) Let  $z^*$  be the optimal objective function value for the integer program. What is the **largest lower bound**  $\underline{z}$  that you can determine given the current state of the search (i.e., the largest number  $\underline{z}$  so that  $\underline{z} \leq z^*$ )?
- (b) Let  $z^*$  be the optimal objective function value for the integer program. What is the **smallest upper bound**  $\bar{z}$  that you can determine given the current state of the search (i.e., the smallest number  $\bar{z}$  so that  $\bar{z} \geq z^*$ )?
- (c) Identify the leaf nodes (among  $B, F, G, H, I$ ) that can be fathomed (i.e., leaf nodes that do not require further exploration), and for each of these, give the reason why it can be fathomed (use the node labels to identify them).
- (d) Suppose the optimal solution to the LP relaxation at node C is  $(x_1, x_2, x_3, x_4, x_5) = (1, 0, 0, 0.75, 3)$ . What additional constraints would have been added to create nodes D and E?



### Problem 3: Branch-and-Bound Method (30 pts).

Use the branch-and-bound method to solve the following integer program:

$$\begin{aligned}
 & \text{maximize} && 17x + 12y \\
 & \text{subject to} && 10x + 7y \leq 40 \\
 & && x + y \leq 5 \\
 & && x, y \geq 0 \\
 & && x, y \in \mathbb{Z}.
 \end{aligned}$$

Form the branch-and-bound tree and indicate the solution associated with each node (similar to the procedures introduced in the lecture). Please include your calculations in your answer.

**Problem 4: Manufacturing Company (30 pts).**

A manufacturing company plans to build new factories (variables  $x_1$  and  $x_2$ ) and warehouses (variables  $x_3$  and  $x_4$ ) in Shenzhen and/or Beijing. The company wants to solve the following binary integer program to determine the location and number of the potential factories and warehouses:

$$\begin{aligned} \text{maximize} \quad & 9x_1 + 5x_2 + 6x_3 + 4x_4 \\ \text{subject to} \quad & 6x_1 + 3x_2 + 5x_3 + 2x_4 \leq 10 \\ & x_3 + x_4 \leq 1 \\ & x_3 - x_1 \leq 0 \\ & x_4 - x_2 \leq 0 \\ & x_1, x_2, x_3, x_4 \in \{0, 1\}. \end{aligned}$$

- (a) Discuss and interpret the meaning of the constraints " $x_3 + x_4 \leq 1$ ", " $x_3 - x_1 \leq 0$ ", and " $x_4 - x_2 \leq 0$ ".
- (b) Use the branch-and-bound method to solve the integer problem. You are allowed to use an LP solver to solve each of the relaxed linear programs. Please specify the branch-and-bound tree and what you did at each node.

**Problem 1: True or False (20 pts).**

For each of the following statements, state whether it is true or false. Please explain your answers (if not true, please show a counterexample).

- If the linear programming relaxation of an integer programming problem is infeasible, then the integer program itself is also infeasible.
- If the linear programming relaxation of an integer programming problem is unbounded, then the integer program itself is also unbounded.
- If  $x^*$  is an optimal solution of the linear programming relaxation of an integer programming problem, and  $x^*$  satisfies the integer restrictions, then  $x^*$  is an optimal solution to the integer programming problem.
- If the optimal solution to the linear programming relaxation of integer programming problem has exactly one variable taking a non-integer value, then the branch and bound algorithm will terminate after branching once and exploring the resulting two nodes.

T

T

T

F

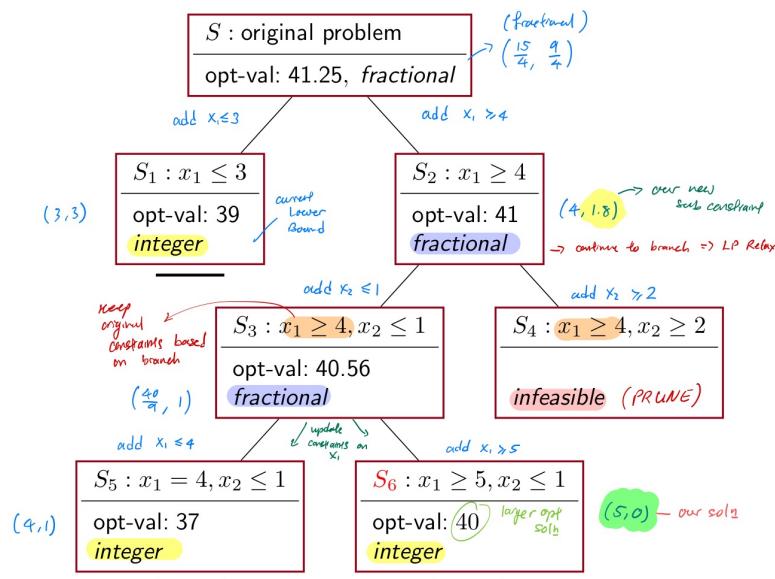
a) IP is a more strict version of LP. If LP relaxation infeasible  $\Rightarrow$  original IP is also infeasible (T)

b) Similar to (a). But since IP keeps all existing constraints of LP, (T)  
IP LP unbounded  $\Rightarrow$  IP unbounded.

c) If  $x^*$  is optimal for LP relaxation and satisfies IP restrictions (integer value) then it is also optimal for IP since it achieves the best solution w/o violating any constraints (T)

d) Branch and bound might take more than 1 branch since the resulting 2 nodes (F)  
might also produce optimal solutions each taking a non-integer value.

Example from lecture:



From the 2nd branch ( $S_2$ )

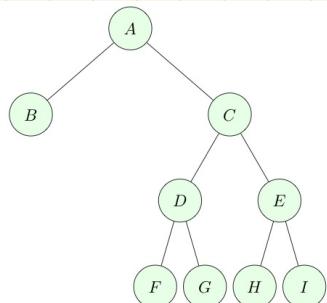
we have in cases where the optimal solution has exactly 1 non-integer value

i.e.  $(4, 1.8)$

$\Rightarrow$  The 1st branch after ( $S_3, S_4$ ) still results in cases with nodes with non-integer values ( $S_5$ ) which requires another branch.

**Problem 2: Branch-and-Bound Tree (20 pts).**

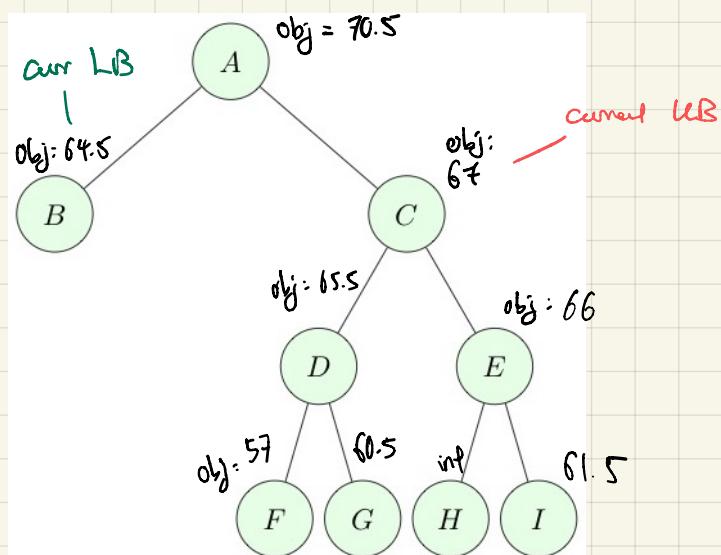
Suppose, the branch-and-bound algorithm is being executed for an integer programming problem with **maximization** objective, and the current state of the search corresponds to the following tree:



Node	LP opt. obj.	$x^{LP}$ integer?
A	70.5	no
B	64.5	no
C	67	no
D	65.5	no
E	66	no
F	57	yes
G	60.5	no
H	Infeasible	n/a
I	61.5	yes

The letters in the nodes correspond to the order in which the LP relaxations have been solved (in alphabetical order). The table above right lists for each node the optimal objective function value for the corresponding node LP relaxation, and indicates whether the resulting solution satisfies all integer restrictions of the original integer program. Assume the original integer program has five decision variables,  $x_1, \dots, x_5$ , and requires all the decision variables to take integer values. Let  $z^*$  denote the optimal value of the integer program being solved.

- Let  $z^*$  be the optimal objective function value for the integer program. What is the **largest lower bound**  $\underline{z}$  that you can determine given the current state of the search (i.e., the largest number  $\underline{z}$  so that  $\underline{z} \leq z^*$ )?
- Let  $z^*$  be the optimal objective function value for the integer program. What is the **smallest upper bound**  $\bar{z}$  that you can determine given the current state of the search (i.e., the smallest number  $\bar{z}$  so that  $\bar{z} \geq z^*$ )?
- Identify the leaf nodes (among B, F, G, H, I) that can be fathomed (i.e., leaf nodes that do not require further exploration), and for each of these, give the reason why it can be fathomed (use the node labels to identify them).
- Suppose the optimal solution to the LP relaxation at node C is  $(x_1, x_2, x_3, x_4, x_5) = (1, 0, 0, 0.75, 3)$ . What additional constraints would have been added to create nodes D and E?



a) Largest Lower Bound = 57

b) Smallest Upper Bound = 66

c) Leaf node: Can be fathomed:

B ✓  
YES

F ✓  
YES

G ✓  
YES

H ✓  
YES

I ✓  
YES

- $x_4 \leq 0$  for node D
- $x_4 \geq 1$  for node E

Reason:

In branch and bound, better to go deep than wide due to memory efficiency and since most integer-solutions lie deep in tree.

$\Rightarrow$  Since this is a maximization, and obj value of C > B ( $67 > 64.5$ ), B can be fathomed

The objective value of B = 64.5, this is the current lower bound. Since Obj value(F) = 57 which is less than 64.5, no better solution can come from this branch  $\Rightarrow$  F is fathomed.

Since no solution from G can produce an obj value  $> 60.5$ , no better solution will come from node G that produces an obj value  $> 61.5$  (node I)

Since H is infeasible  $\Rightarrow$  no need to consider this subproblem further.

This is the optimal solution

**Problem 3: Branch-and-Bound Method (30 pts).**  
Use the branch-and-bound method to solve the following integer program:

$$\begin{aligned} \text{maximize } & 17x + 12y \\ \text{subject to } & 10x + 7y \leq 40 \\ & x + y \leq 5 \\ & x, y \geq 0 \\ & x, y \in \mathbb{Z}. \end{aligned}$$

Form the branch-and-bound tree and indicate the solution associated with each node (similar to the procedures introduced in the lecture). Please include your calculations in your answer.

$S_{\text{original}}$ , Relaxed LP:

$$y = 5 - x \Rightarrow 10x + 7y = 40$$

$$\Rightarrow 10x + 7(5 - x) = 40$$

$$10x + 35 - 7x = 40 \quad \therefore \text{opt value}$$

$$\begin{aligned} 3x &= 5 \\ x &= \frac{5}{3} \\ &= 1.67 \end{aligned}$$

$$\therefore y = \frac{10}{3}, \text{ branch for } x$$

$$S_1: x \leq 1, \text{ opt soln} = (1, 4), \text{ opt val: } 65$$

Current L-bound = 65, stop here, expand  $S_2$

$$S_2: x \geq 2, x = 2 \Rightarrow 10x + 7y = 40$$

$$\Rightarrow 7y = 20 \Leftrightarrow y = \frac{20}{7}$$

$$\text{opt soln: } (2, \frac{20}{7}), \text{ opt val: } \frac{478}{7}$$

branch for  $y$

$$S_3: y \leq 2, \text{ graphically, opt soln: } (2.6, 2)$$

and opt val: 68.2  $\Rightarrow$  branch for  $x$

$S_4: y \geq 3$ , infeasible  $\Rightarrow$  prune

$$S_5: x \leq 2 \text{ with } (S_2) \Rightarrow x = 2, \text{ graphically:}$$

$$\text{opt soln: } (2, 2), \text{ opt val: } 58 < \text{L-bound}$$

$\Rightarrow$  no branching, expand  $S_6$

$$S_6: x \geq 3, x = 3 \Rightarrow 10x + 7y = 40$$

$$\Rightarrow 7y = 10 \Rightarrow y = \frac{10}{7}$$

$$\text{opt soln: } (3, \frac{10}{7}), \text{ opt val: } 68.143$$

$\Rightarrow$  branch for  $y$

$$S_7: y \leq 1, y = 1 \Rightarrow 10x + 7y = 40$$

$$\Rightarrow 10x = 33 \Rightarrow x = 3.3, \text{ opt soln: } (3.3, 1)$$

$$\text{opt val: } 68.1$$

$\Rightarrow$  branch for  $x$

$S_8: y \geq 2$ , infeasible  $\Rightarrow$  prune

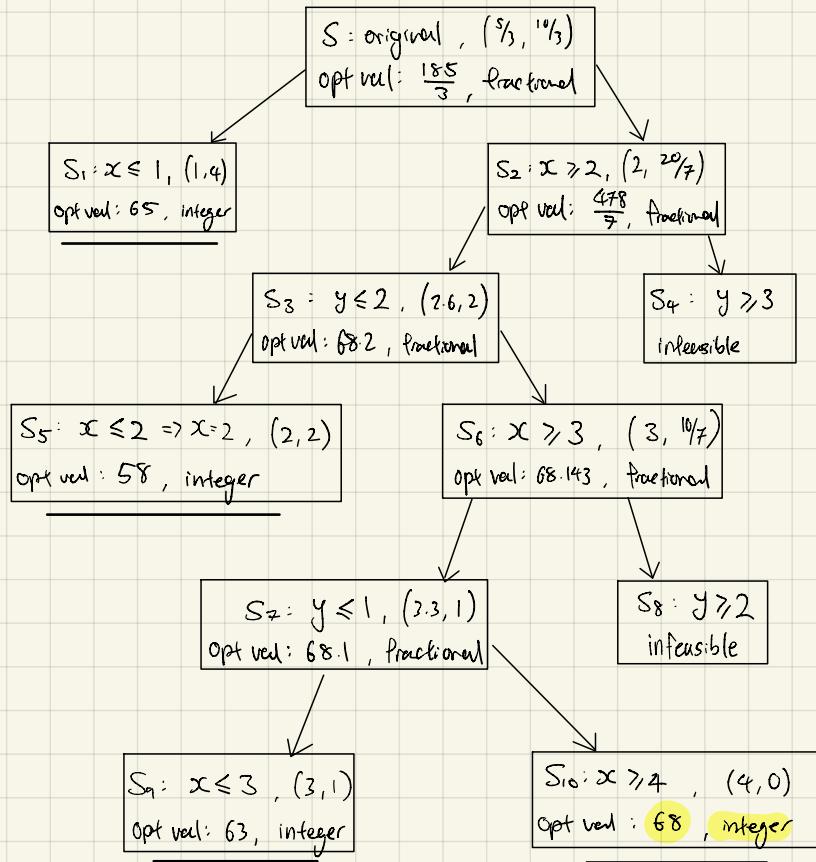
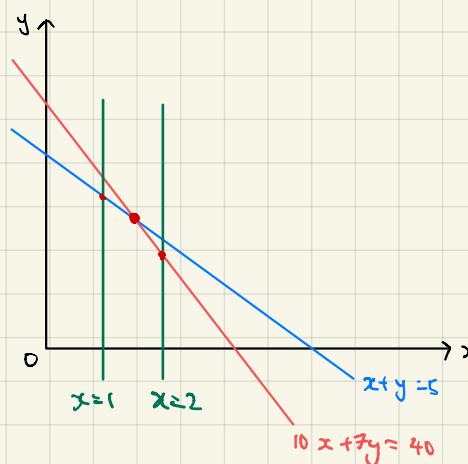
$$S_9: x \leq 3 \text{ with } (S_6) \Rightarrow x = 3$$

$$\text{Graphically: opt soln: } (3, 1)$$

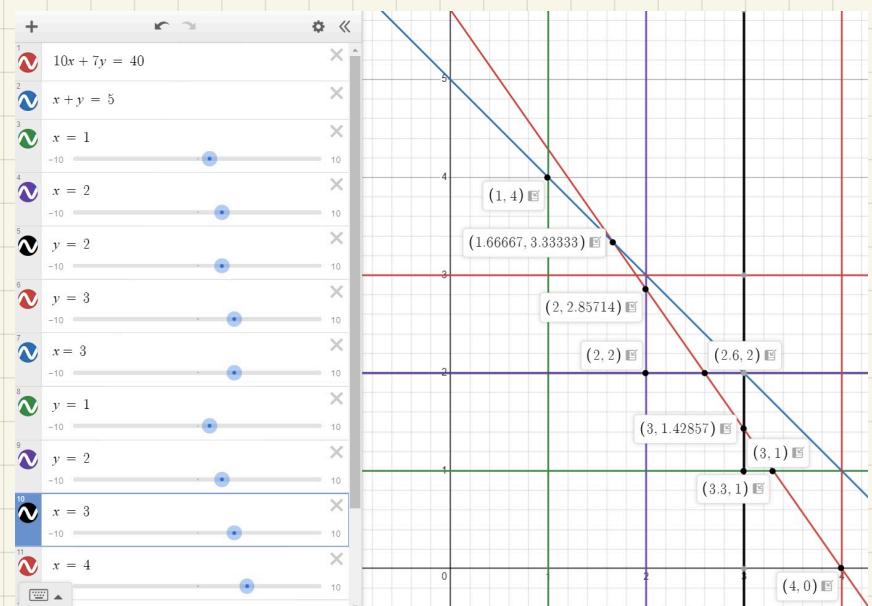
opt val: 63 < L-bound (stop, no branching)

$$S_{10}: x \geq 4, \text{ graphically, opt soln: } (4, 0)$$

$$\text{opt val: } 68 \Rightarrow \text{Final optimal soln } \checkmark$$



Optimized solution:  $x = 3, y = 1$   
Optimal value: 68



**Problem 4: Manufacturing Company (30 pts).**

A manufacturing company plans to build new factories (variables  $x_1$  and  $x_2$ ) and warehouses (variables  $x_3$  and  $x_4$ ) in Shenzhen and/or Beijing. The company wants to solve the following binary integer program to determine the location and number of the potential factories and warehouses:

$$\begin{array}{ll} \text{maximize} & 9x_1 + 5x_2 + 6x_3 + 4x_4 \\ \text{subject to} & 6x_1 + 3x_2 + 5x_3 + 2x_4 \leq 10 \\ & x_3 + x_4 \leq 1 \\ & x_3 - x_1 \leq 0 \\ & x_4 - x_2 \leq 0 \\ & x_1, x_2, x_3, x_4 \in \{0, 1\}. \end{array}$$

(a) Discuss and interpret the meaning of the constraints " $x_3 + x_4 \leq 1$ ", " $x_3 - x_1 \leq 0$ ", and " $x_4 - x_2 \leq 0$ ".

(b) Use the branch-and-bound method to solve the integer problem. You are allowed to use an LP solver to solve each of the relaxed linear programs. Please specify the branch-and-bound tree and what you did at each node.

a) " $x_3 + x_4 \leq 1$ "

$\Rightarrow$  implies that the company can only have one warehouse in Shenzhen or Beijing, but not both.

" $x_3 - x_1 \leq 0$ " i.e.  $x_3 = 1 \Rightarrow x_1 = 1$

$\Rightarrow$  if the warehouse is in Shenzhen, the factory must also be in Shenzhen

" $x_4 - x_2 \leq 0$ " i.e.  $x_4 = 1 \Rightarrow x_2 = 1$

$\Rightarrow$  if the warehouse is in Beijing, the factory must also be in Beijing.

b) Using an LP solver: Relax  $0 \leq x_1, x_2, x_3, x_4 \leq 1$

✓ S - Original

```
> ✓ # Code for Solving the Original Problem
...
**** S - Original Problem ****
Status: Optimal
x1 = 0.83333333
x2 = 1.0
x3 = 0.0
x4 = 1.0
Objective value: 16.49999997
```

S - Original:

- Fractional
- proceed to branch on  $x_1$
- add  $x_1 = 0$  (S1)
- add  $x_1 = 1$  (S2)

✓ S - 1 : Add  $x_1 = 0$

```
> ✓ # Code for Solving the Problem with Constraint S1 ...
...
**** S - 1 : x1 == 0 ****
Status: Optimal
x1 = 0.0
x2 = 1.0
x3 = 0.0
x4 = 1.0
Objective value: 9.0
```

S - 1  $x_1 = 0$ :

- Integer
- Current Lbound: 9
- no need for further branching

✓ S2 - Add  $x_1 = 1$

```
> ✓ # Code for Solving the Problem with Constraint S2 ...
...
**** S - 2 : x1 == 1 ****
Status: Optimal
x1 = 1.0
x2 = 0.8
x3 = 0.0
x4 = 0.8
Objective value: 16.2
```

S - 2  $x_1 = 1$ :

- Fractional
- Current Ubound (LP - relaxed): 16.2
- Branch on  $x_2$
- Add  $x_2 = 0$  (S3)
- Add  $x_2 = 1$  (S4)

✓ S3 - Add  $x_2 = 0$

```
> ✓ # Code for Solving the Problem with Constraint S3 ...
...
**** S - 3 : x2 == 0 ****
Status: Optimal
x1 = 1.0
x2 = 0.0
x3 = 0.8
x4 = 0.0
Objective value: 13.8
```

S - 3  $x_2 = 0$ :

- Fractional
- Branch on  $x_3$
- Add  $x_3 = 0$  (S5)
- Add  $x_3 = 1$  (S6)

✓ S4 - Add  $x_2 = 1$

```
> ✓ # Code for Solving the Problem with Constraint S4 ...
...
**** S - 4 : x2 == 1 ****
Status: Optimal
x1 = 1.0
x2 = 1.0
x3 = 0.0
x4 = 0.5
Objective value: 16.0
```

S - 4  $x_2 = 1$ :

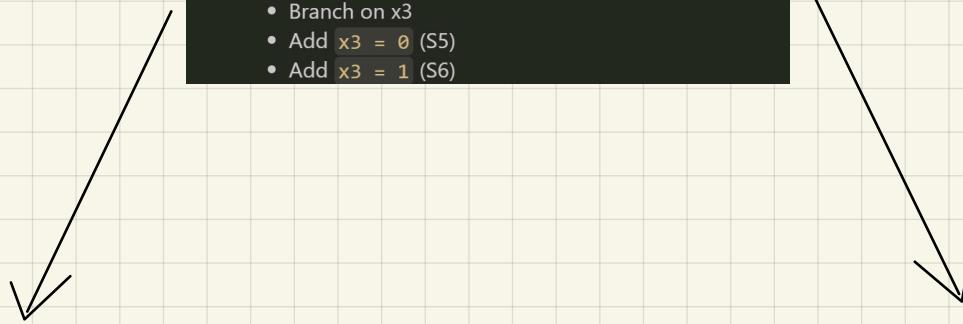
- Fractional
- Branch on  $x_4$
- Add  $x_4 = 0$  (S7)
- Add  $x_4 = 1$  (S8)

# Continuing from node S3

```
✓ S3 - Add x2 = 0

✓ # Code for Solving the Problem with Constraint S3 ...
...
**** S - 3 : x2 == 0 ****
Status: Optimal
x1 = 1.0
x2 = 0.0
x3 = 0.8
x4 = 0.0
Objective value: 13.8

S - 3 x2 = 0:
• Fractional
• Branch on x3
• Add x3 = 0 (S5)
• Add x3 = 1 (S6)
```



```
▷ ✓ # Code for Solving the Problem with Constraint S5 ...
...
**** S - 5 : x3 == 0 ****
Status: Optimal
x1 = 1.0
x2 = 0.0
x3 = 0.0
x4 = 0.0
Objective value: 9.0

S - 5 x3 = 0:
• Integer
• No need for further branching
```

```
▷ ✓ # Code for Solving the Problem with Constraint S6 ...
...
**** S - 6 : x3 == 1 ****
Status: Infeasible
x1 = 1.0
x2 = 0.0
x3 = 1.0
x4 = -0.5
Objective value: 13.0

S - 6 x3 = 1:
• Infeasible
• Prune this branch
```

Continuing from node S4

```
▽ S4 - Add x2 = 1

▷ ✓ # Code for Solving the Problem with Constraint S4 ...
...
**** S - 4 : x2 == 1 ****
Status: Optimal
x1 = 1.0
x2 = 1.0
x3 = 0.0
x4 = 0.5
Objective value: 16.0

S - 4 x2 = 1:


- Fractional
- Branch on x4
- Add x4 = 0 (S7)
- Add x4 = 1 (S8)

```

```
▷ ✓ # Code for Solving the Problem with Constraint S7 ...
...
**** S - 7 : x4 == 0 ****
Status: Optimal
x1 = 1.0
x2 = 1.0
x3 = 0.2
x4 = 0.0
Objective value: 15.2

S - 7 x4 = 0:


- Fractional
- Branch on x3
- Add x3 = 0 (S9)
- Add x3 = 1 (S10)

```

```
▷ ✓ # Code for Solving the Problem with Constraint S8 ...
...
**** S - 8 : x4 == 1 ****
Status: Infeasible
x1 = 1.0
x2 = 0.66666667
x3 = 0.0
x4 = 1.0
Objective value: 16.33333335

S - 8 x4 = 1:


- Infeasible
- Prune this branch

```

```
▷ ✓ # Code for Solving the Problem with Constraint S9 ...
...
**** S - 9 : x3 == 0 ****
Status: Optimal
x1 = 1.0
x2 = 1.0
x3 = 0.0
x4 = 0.0
Objective value: 14.0

S - 9 x3 = 0:


- Integer
- No further branching required

```

```
▷ ✓ # Code for Solving the Problem with Constraint S10 ...
...
**** S - 10 : x3 == 1 ****
Status: Infeasible
x1 = 1.0
x2 = -0.33333333
x3 = 1.0
x4 = 0.0
Objective value: 13.33333335

S - 10 x3 = 1:


- Infeasible
- Prune this branch

```

Optimal solution: (1, 1, 0, 0)

Optimal value: 14.0