

Modeling & Exact Optimization

Lab 1



Class conduct



Class conduct

- a QUIZ on March 11 during the lab time (no laptops, no phones, no books)



Class conduct

- a QUIZ on March 11 during the lab time (no laptops, no phones, no books)
- be courteous to your fellow students and instructors.



Class conduct

- a **QUIZ** on March 11 during the lab time (**no laptops, no phones, no books**)
- be courteous to your fellow students and instructors.
- turn off or put your cellular phones on silent mode. If you must answer the phone, please quietly leave the classroom.



Class conduct

- a **QUIZ** on March 11 during the lab time (**no laptops, no phones, no books**)
- be courteous to your fellow students and instructors.
- turn off or put your cellular phones on silent mode. If you must answer the phone, please quietly leave the classroom.
- do not use laptops or tablets during class if you intend to play movies, video games, do other homework, etc.



Class conduct

- a **QUIZ** on March 11 during the lab time (**no laptops, no phones, no books**)
- be courteous to your fellow students and instructors.
- turn off or put your cellular phones on silent mode. If you must answer the phone, please quietly leave the classroom.
- do not use laptops or tablets during class if you intend to play movies, video games, do other homework, etc.
- do not chatter with friends or purposely disturb other students.



Class conduct

- a **QUIZ on March 11 during the lab time (no laptops, no phones, no books)**
- be courteous to your fellow students and instructors.
- turn off or put your cellular phones on silent mode. If you must answer the phone, please quietly leave the classroom.
- do not use laptops or tablets during class if you intend to play movies, video games, do other homework, etc.
- do not chatter with friends or purposely disturb other students.
- leave and re-enter the classroom quietly if you must use the restroom.



Class conduct

- a **QUIZ on March 11 during the lab time (no laptops, no phones, no books)**
- be courteous to your fellow students and instructors.
- turn off or put your cellular phones on silent mode. If you must answer the phone, please quietly leave the classroom.
- do not use laptops or tablets during class if you intend to play movies, video games, do other homework, etc.
- do not chatter with friends or purposely disturb other students.
- leave and re-enter the classroom quietly if you must use the restroom.
- any type type of misconduct won't be tolerated, e.g., cheating, dishonesty, ...

Evolution Has a Rule: Use It or Lose It

Evolution Has a Rule: Use It or Lose It

- Examples:
 - Humans lost functional tails → now only the coccyx remains.
 - Wisdom teeth shrinking → softer diets, smaller jaws.
 - Ear muscles mostly inactive → no survival pressure to rotate ears.

Evolution Has a Rule: Use It or Lose It

- Examples:
 - Humans lost functional tails → now only the coccyx remains.
 - Wisdom teeth shrinking → softer diets, smaller jaws.
 - Ear muscles mostly inactive → no survival pressure to rotate ears.
- Same principle applies to your brain:
 - If AI generates your reasoning, you skip the mental reps.
 - Skipping mental reps reduces long-term retention.
 - Dependency increases, independent ability decreases.
 - Over time, your baseline thinking ability can weaken.

Introduction to Mathematical Programming

Zoo Problem

- 300 kids need to travel to the New York Zoo. The school may rent 40 seats and 30 seats buses for \$500 and \$400 . How many buses of each size do you use to minimize cost?
- What is the objective (function)?
- What is the decision to make?
- What are the constraints?



Introduction to Mathematical Programming

Zoo Problem

- 300 kids need to travel to the New York Zoo. The school may rent 40 seats and 30 seats buses for \$500 and \$400 . How many buses of each size do you use to minimize cost?
- What is the objective (function)?
Minimize the cost
- What is the decision to make?
- What are the constraints?



Introduction to Mathematical Programming

Zoo Problem

- 300 kids need to travel to the New York Zoo. The school may rent 40 seats and 30 seats buses for \$500 and \$400 . How many buses of each size do you use to minimize cost?
- What is the objective (function)?
Minimize the cost
- What is the decision to make?
How many of which buses to select
- What are the constraints?



Introduction to Mathematical Programming

Zoo Problem

- 300 kids need to travel to the New York Zoo. The school may rent 40 seats and 30 seats buses for \$500 and \$400 . How many buses of each size do you use to minimize cost?
Minimize the cost
- What is the decision to make?
How many of which buses to select
- What are the constraints?
Cannot exceed the bus seat capacities



Introduction to Mathematical Programming

Zoo Problem

- 300 kids need to travel to the New York Zoo. The school may rent 40 seats and 30 seats buses for \$500 and \$400 . How many buses of each size do you use to minimize cost?
Minimize the cost
- What is the decision to make?
How many of which buses to select
- What are the constraints?
Cannot exceed the bus seat capacities



Intuitive solution: A seat costs \$12.50 for 40 seats buses and \$13.33 for 30 seats buses. Seven 40-seat and one 30-seat. Spend \$3900.

Introduction to Mathematical Programming

Zoo Problem

- 300 kids need to travel to the New York Zoo. The school may rent 40 seats and 30 seats buses for \$500 and \$400 . How many buses of each size do you use to minimize cost?
Minimize the cost
- What is the decision to make?
How many of which buses to select
- What are the constraints?
Cannot exceed the bus seat capacities



Intuitive solution: A seat costs \$12.50 for 40 seats buses and \$13.33 for 30 seats buses. Seven 40-seat and one 30-seat. Spend \$3900.

Optimal solution: Six 40-seat and two 30-seat. Spend \$3800.

Introduction to Mathematical Programming

Knapsack Problem

- You are going for a vacation with your backpack (knapsack) and you could only carry 10kg with you. How do you choose which items to take with you?
- What is the objective (function)?
- What is the decision to make?
- What are the constraints?



Introduction to Mathematical Programming

Knapsack Problem

- You are going for a vacation with your backpack (knapsack) and you could only carry 10kg with you. How do you choose which items to take with you?
- What is the objective (function)?
Maximize the value of items
- What is the decision to make?
- What are the constraints?



Introduction to Mathematical Programming

Knapsack Problem

- You are going for a vacation with your backpack (knapsack) and you could only carry 10kg with you. How do you choose which items to take with you?
- What is the objective (function)?
Maximize the value of items
- What is the decision to make?
Which items to choose
- What are the constraints?



Introduction to Mathematical Programming

- You are going for a vacation with your backpack (knapsack) and you could only carry 10kg with you. How do you choose which items to take with you?
 - What is the objective (function)?
Maximize the value of items
 - What is the decision to make?
Which items to choose
 - What are the constraints?
Cannot exceed backpack capacityA cartoon illustration of a hiker standing in a mountainous landscape. The hiker is wearing an orange jacket, brown pants, and a brown backpack. A thought bubble above the hiker's head contains icons of a tent, a rope, a camera, a gold bar, a pair of boots, and a book, all connected by lines to a large question mark, symbolizing the decision-making process of what items to take on a trip.



Introduction to Mathematical Programming

Knapsack Problem

- You are going for a vacation with your backpack (knapsack) and you could only carry 10kg with you. How do you choose which items to take with you?

Cap =10kg	Weight	Value
A	6	30
B	3	14
C	4	16
D	2	9
E	5	20



Introduction to Mathematical Programming

Knapsack Problem

- You are going for a vacation with your backpack (knapsack) and you could only carry 10kg with you. How do you choose which items to take with you?

Cap =10kg	Weight	Value	Value/Weight
A	6	30	5.00
B	3	14	4.67
C	4	16	4.00
D	2	9	4.50
E	5	20	4.00



Introduction to Mathematical Programming

Knapsack Problem

- You are going for a vacation with your backpack (knapsack) and you could only carry 10kg with you. How do you choose which items to take with you?

Cap =10kg	Weight	Value	Value/Weight
A	6	30	5.00
B	3	14	4.67
C	4	16	4.00
D	2	9	4.50
E	5	20	4.00



Intuitive solution: Sort by their Value/Weight ratio and select until the capacity is reached. Choose items A and B with the total value of 44.

Introduction to Mathematical Programming

Knapsack Problem

- You are going for a vacation with your backpack (knapsack) and you could only carry 10kg with you. How do you choose which items to take with you?

Cap =10kg	Weight	Value	Value/Weight
A	6	30	5.00
B	3	14	4.67
C	4	16	4.00
D	2	9	4.50
E	5	20	4.00



Intuitive solution: Sort by their Value/Weight ratio and select until the capacity is reached. Choose items A and B with the total value of 44.

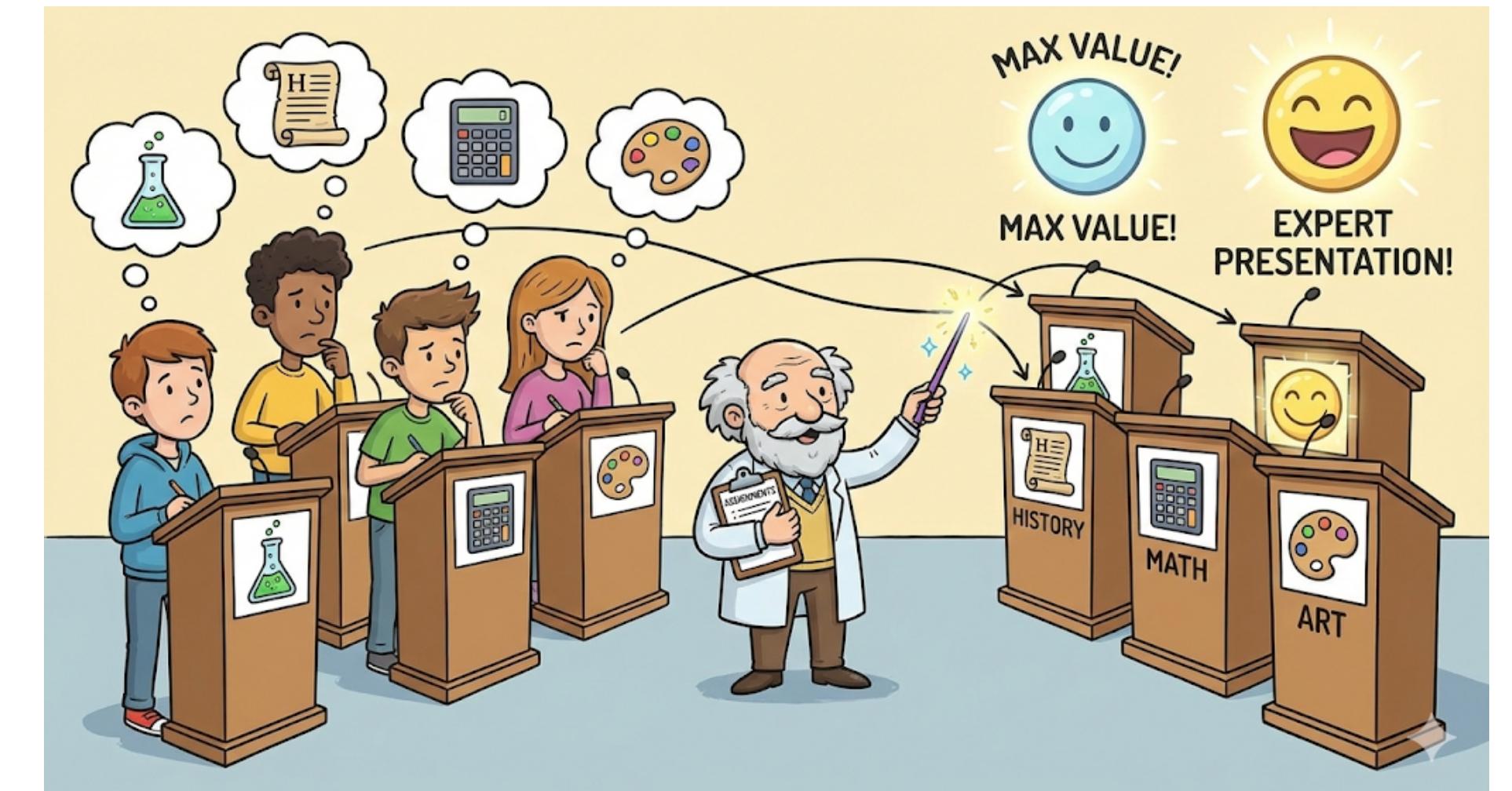
Optimal solution: Choose A and C with the total value of 46.

Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?
- What is the objective (function)?
- What is the decision to make?
- What are the constraints?

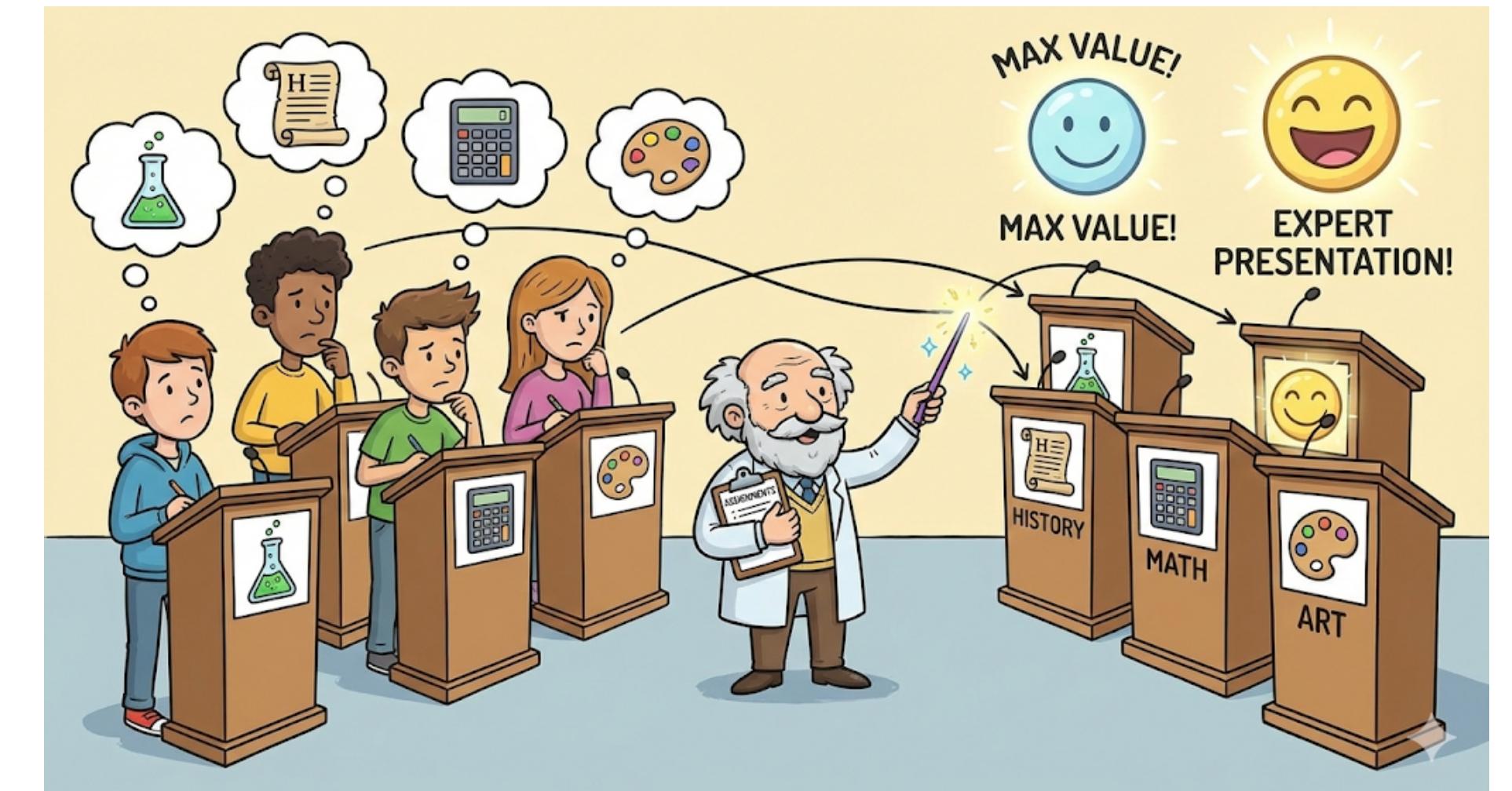


Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?
- What is the objective (function)?
Maximize the total value
- What is the decision to make?
- What are the constraints?

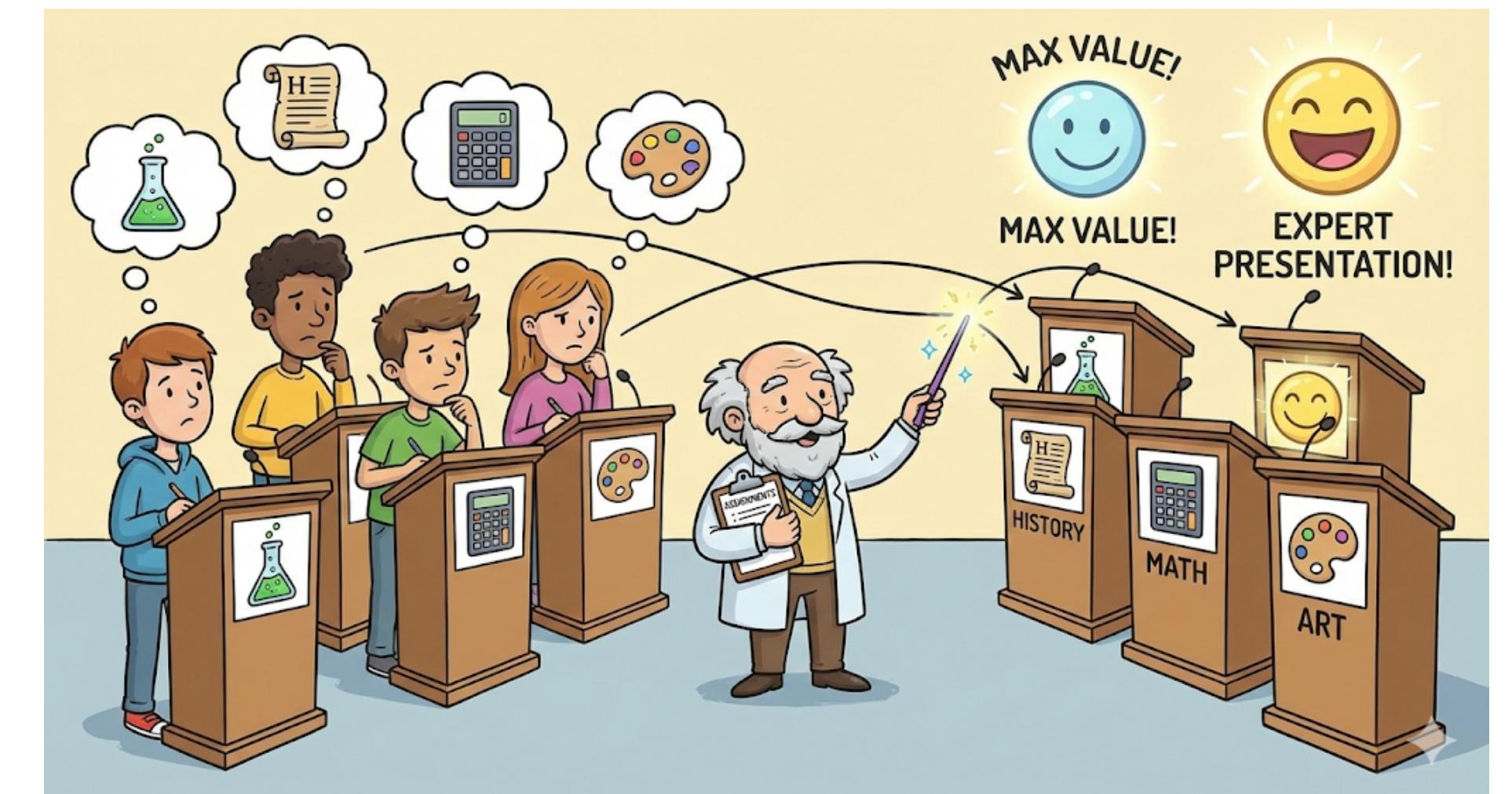


Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?
- What is the objective (function)?
Maximize the total value
- What is the decision to make?
Assignments to students
- What are the constraints?

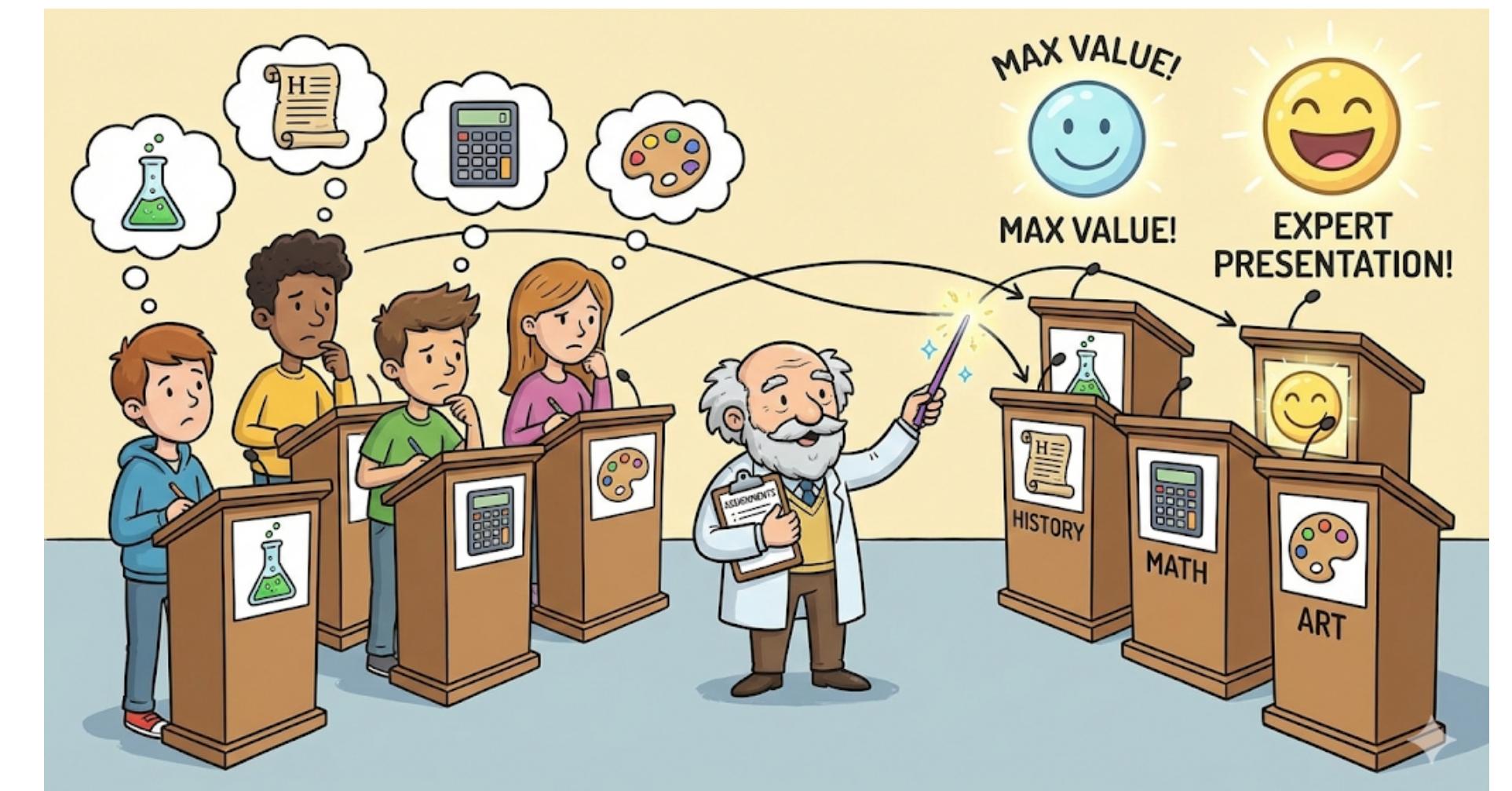


Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?
- What is the objective (function)?
Maximize the total value
- What is the decision to make?
Assignments to students
- What are the constraints?
 - **Each student gets one topic**

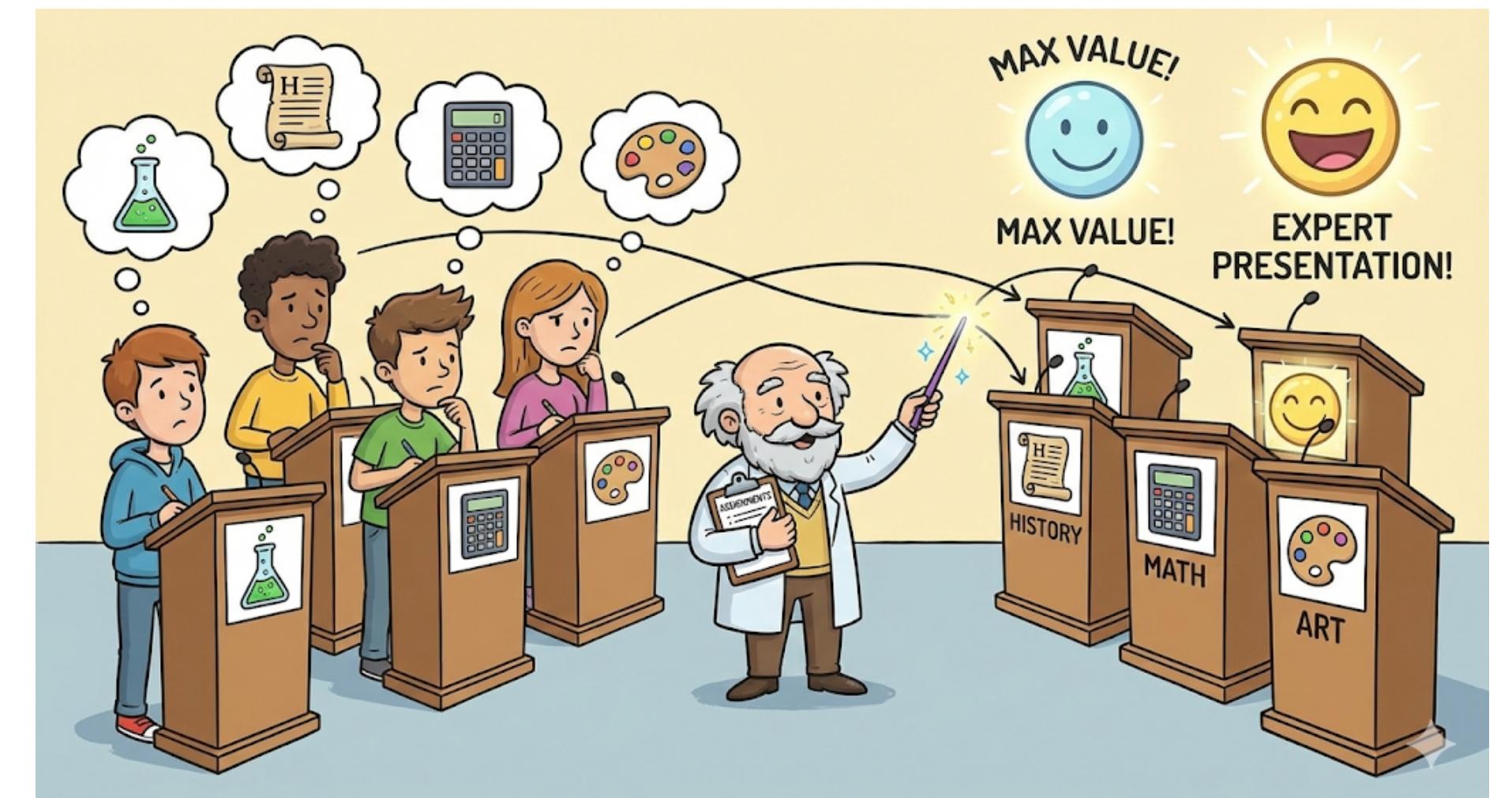


Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?
- What is the objective (function)?
Maximize the total value
- What is the decision to make?
Assignments to students
- What are the constraints?
 - Each student gets one topic
 - Every topic is assigned to one student



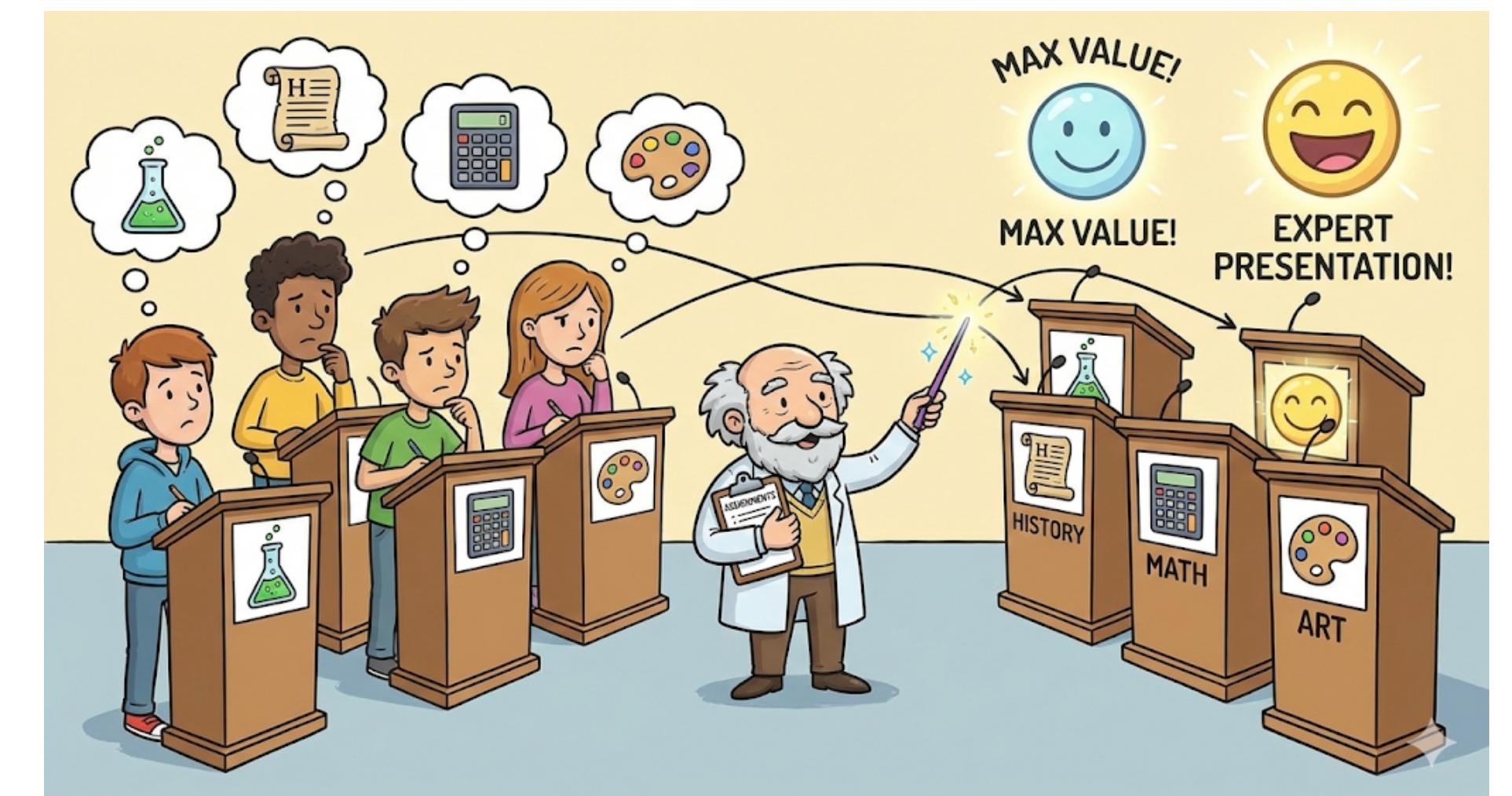
Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?

	Topic A	Topic B	Topic C	Topic D
S1	50	49	1	1
S2	48	2	47	1
S3	2	46	45	1
S4	1	1	100	44



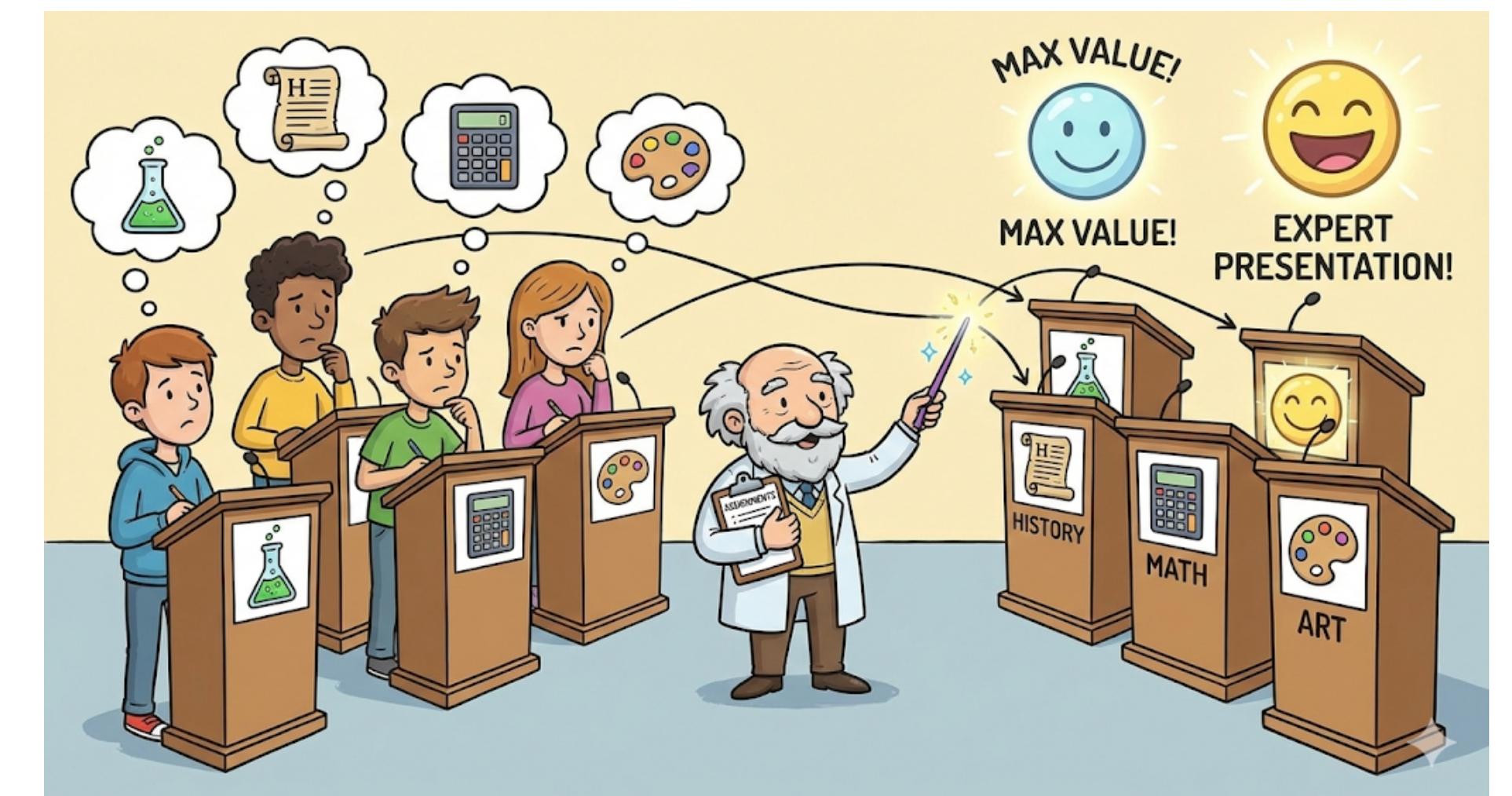
Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?

	Topic A	Topic B	Topic C	Topic D
S1	50	49	1	1
S2	48	2	47	1
S3	2	46	45	1
S4	1	1	100	44



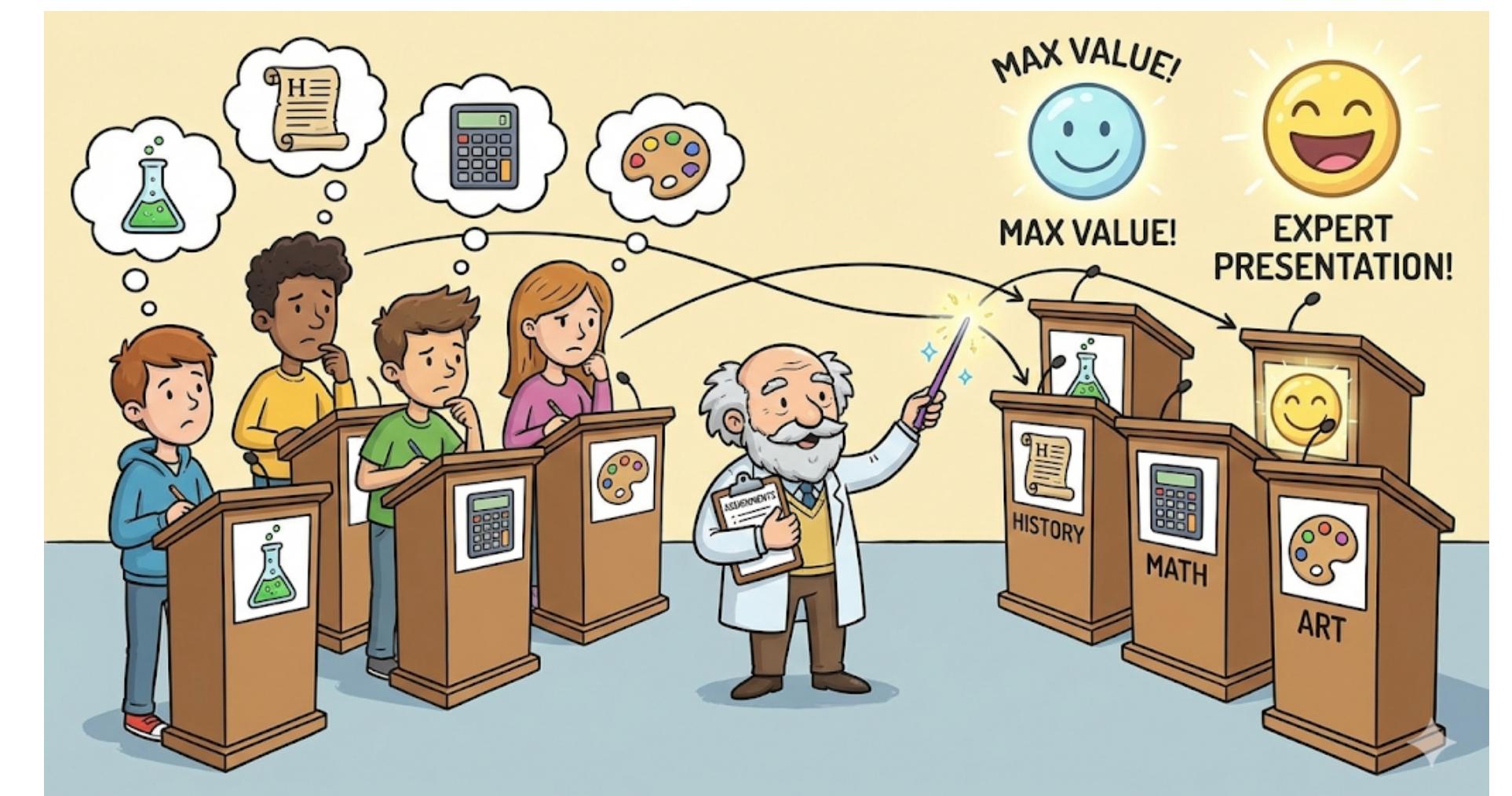
Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?

	Topic A	Topic B	Topic C	Topic D
S1	50	49	1	1
S2	48	2	47	1
S3	2	46	45	1
S4	1	1	100	44



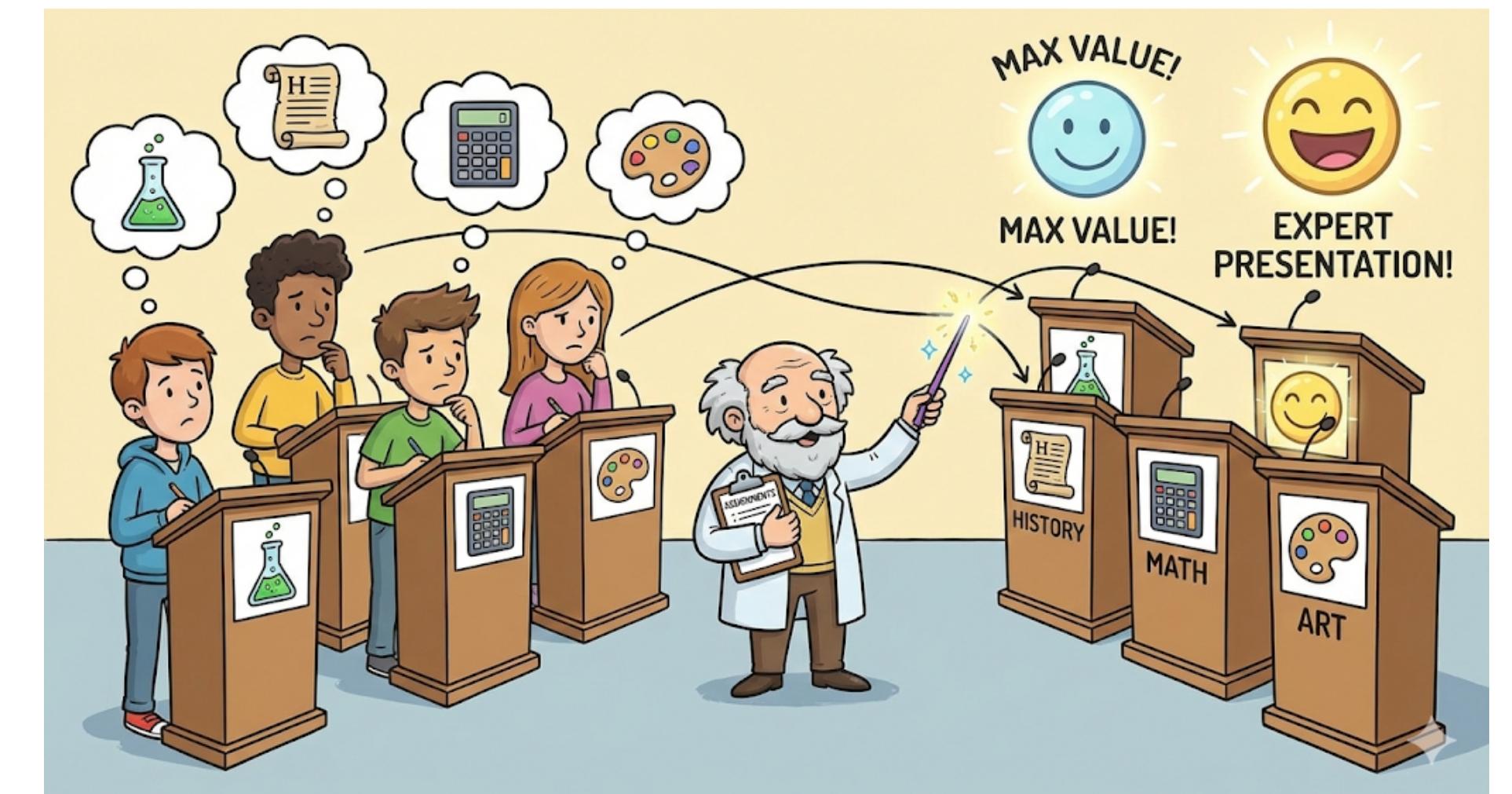
Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?

	Topic A	Topic B	Topic C	Topic D
S1	50	49	1	1
S2	48	2	47	1
S3	2	46	45	1
S4	1	1	100	44



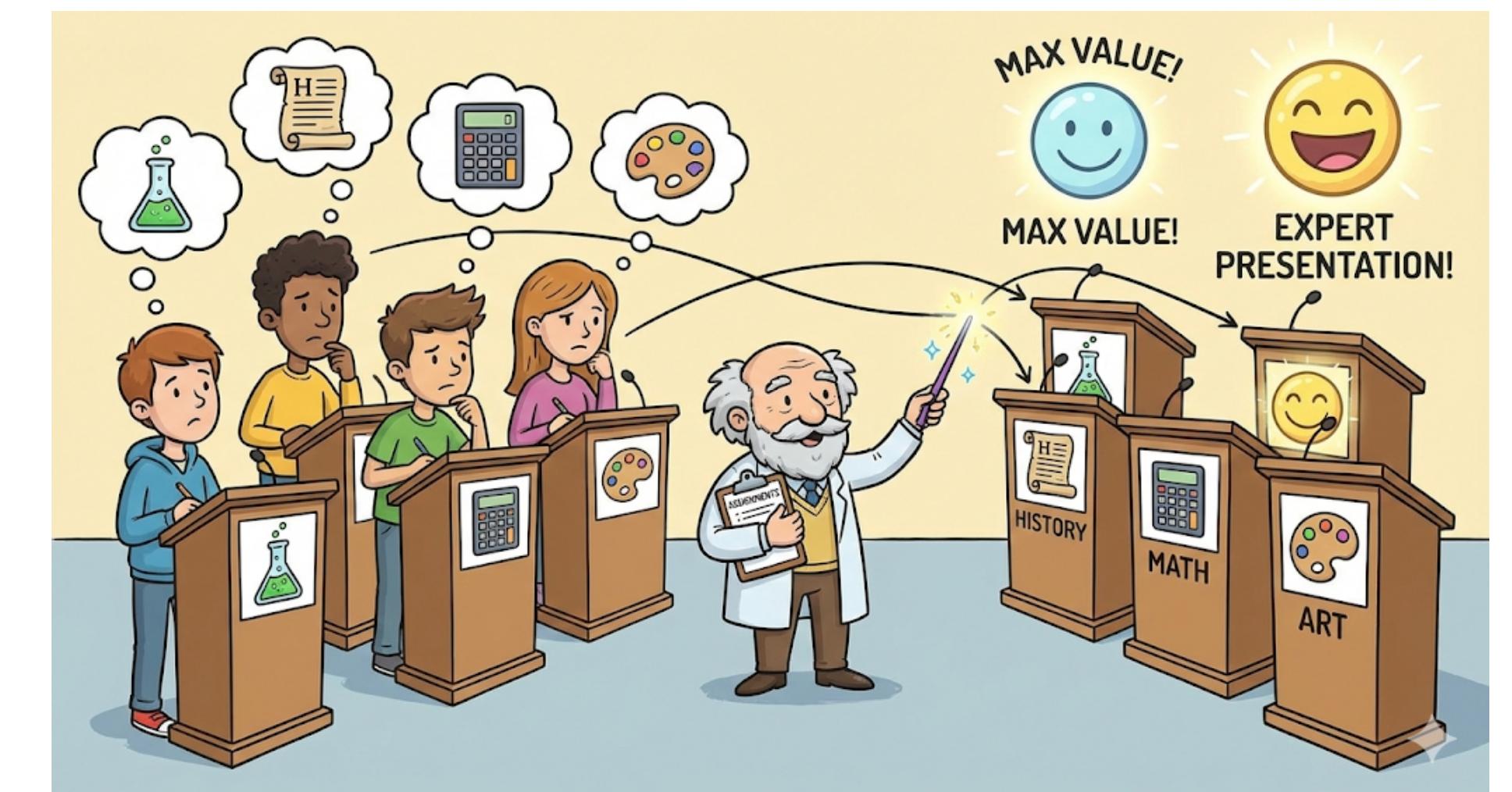
Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?

	Topic A	Topic B	Topic C	Topic D
S1	50	49	1	1
S2	48	2	47	1
S3	2	46	45	1
S4	1	1	100	44



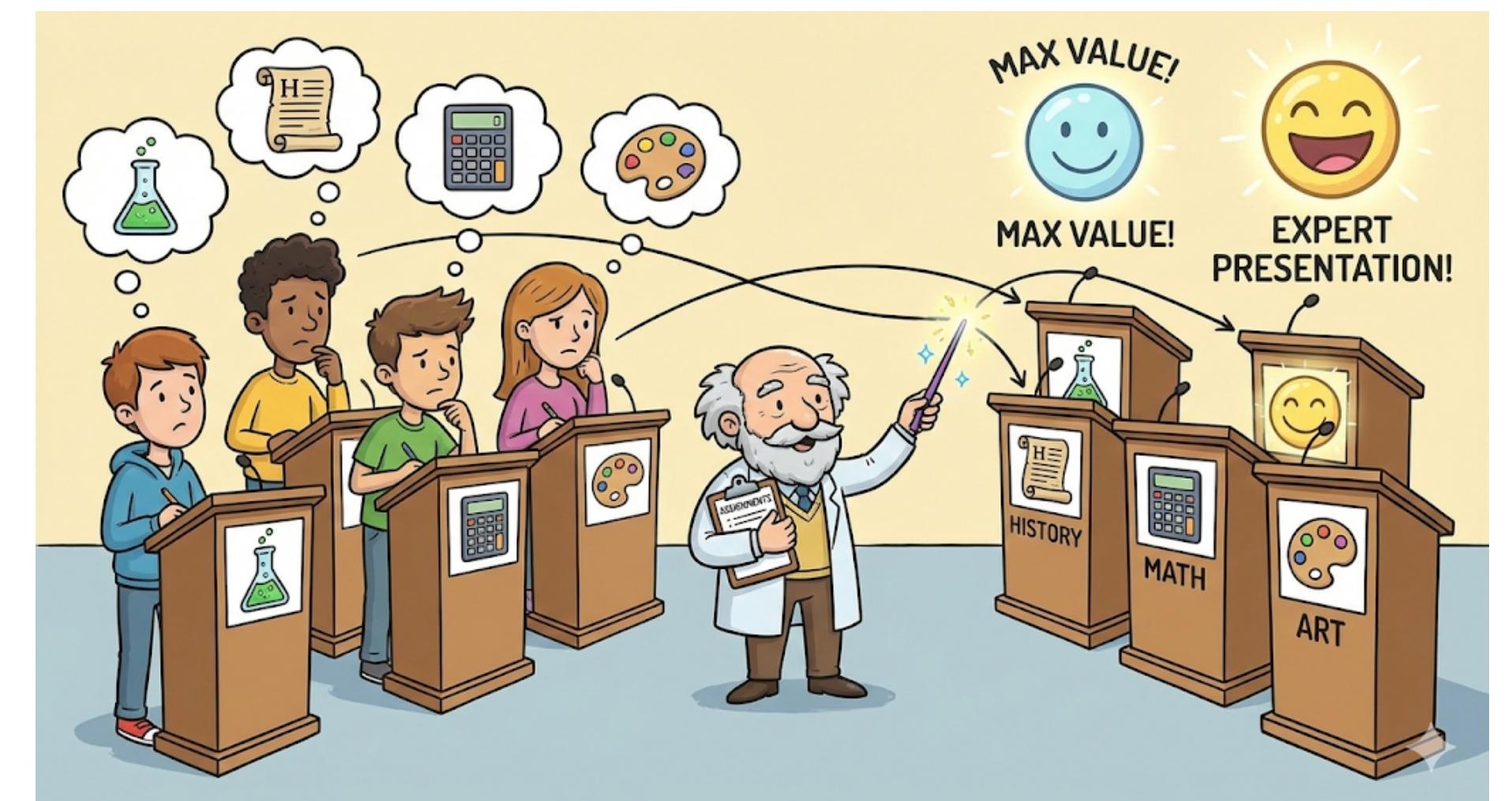
Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?

	Topic A	Topic B	Topic C	Topic D
S1	50	49	1	1
S2	48	2	47	1
S3	2	46	45	1
S4	1	1	100	44



Intuitive solution: Start from Student 1 and assign them to the highest scored topics: S1: A, S2: C, S3: B, S4: D. Total value = 187.

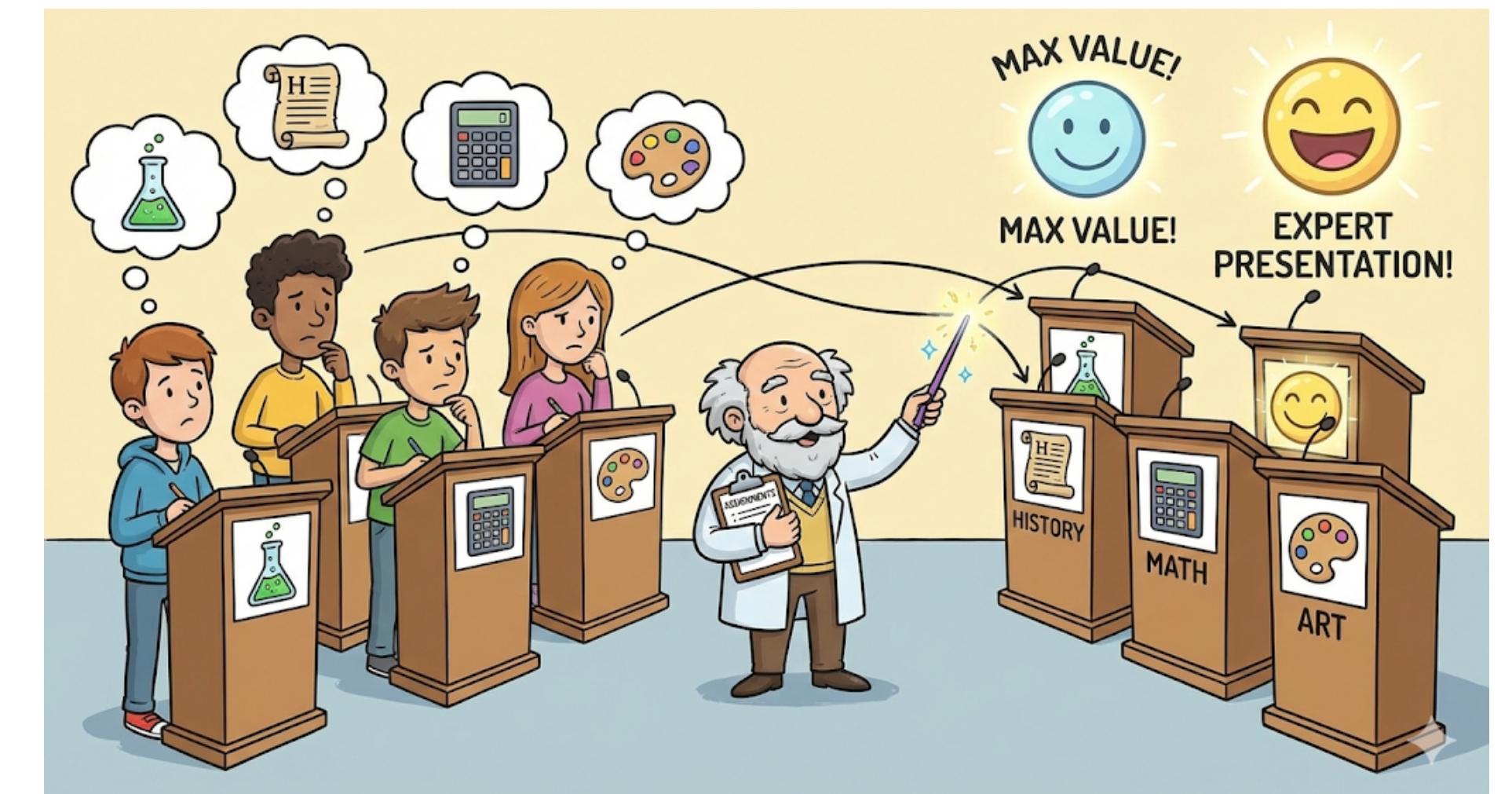
Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?

	Topic A	Topic B	Topic C	Topic D
S1	50	49	1	1
S2	48	2	47	1
S3	2	46	45	1
S4	1	1	100	44



Intuitive solution: Start from Student 1 and assign them to the highest scored topics: S1: A, S2: C, S3: B, S4: D. Total value = 187.

Optimal solution: S1: B, S2: A, S3: D, S4: C. Total value = 198.

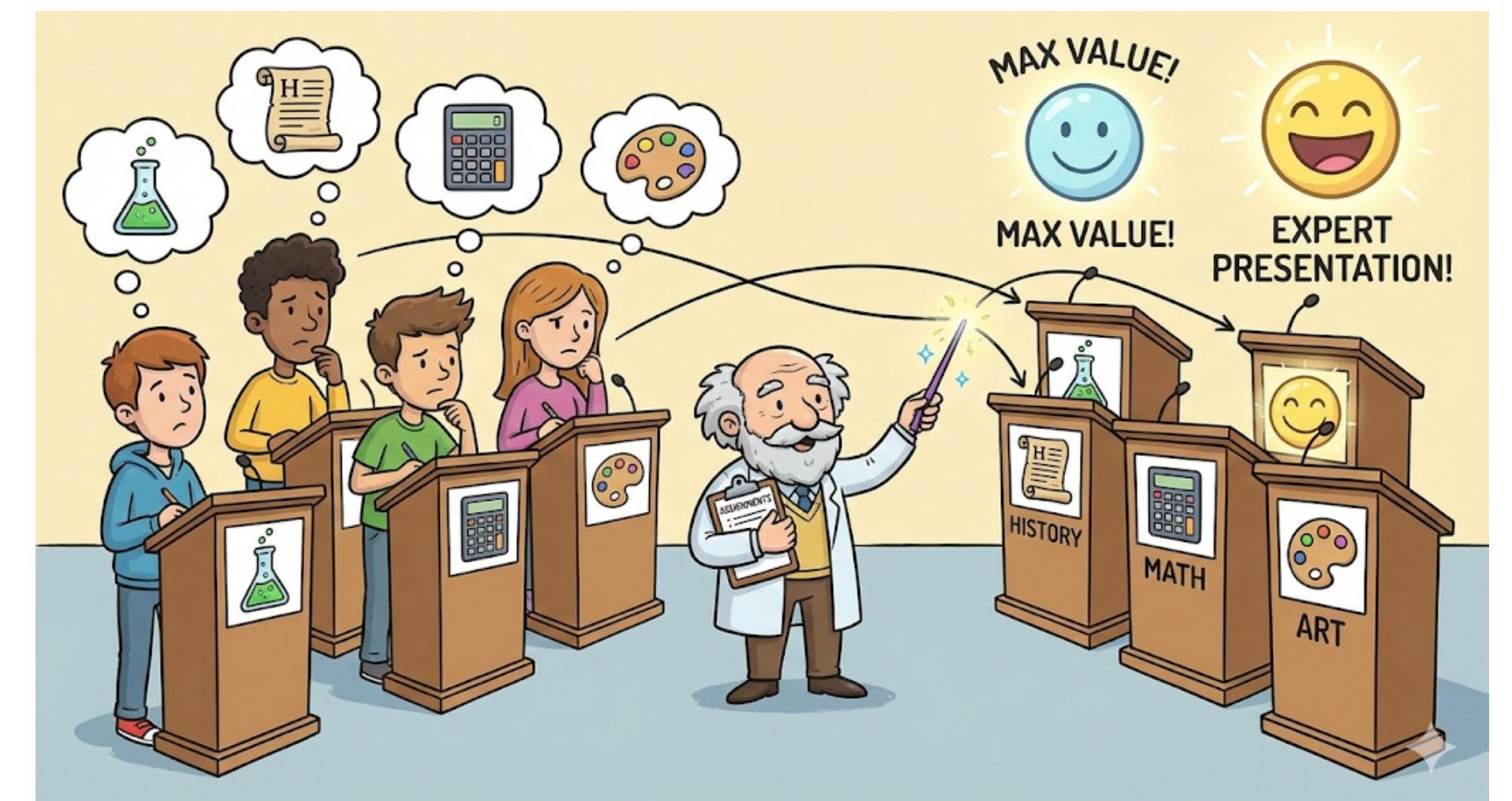
Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?

	Topic A	Topic B	Topic C	Topic D
S1	50	49	1	1
S2	48	2	47	1
S3	2	46	45	1
S4	1	1	100	44



🧠 What about Student 3, it will be a very bad presentation

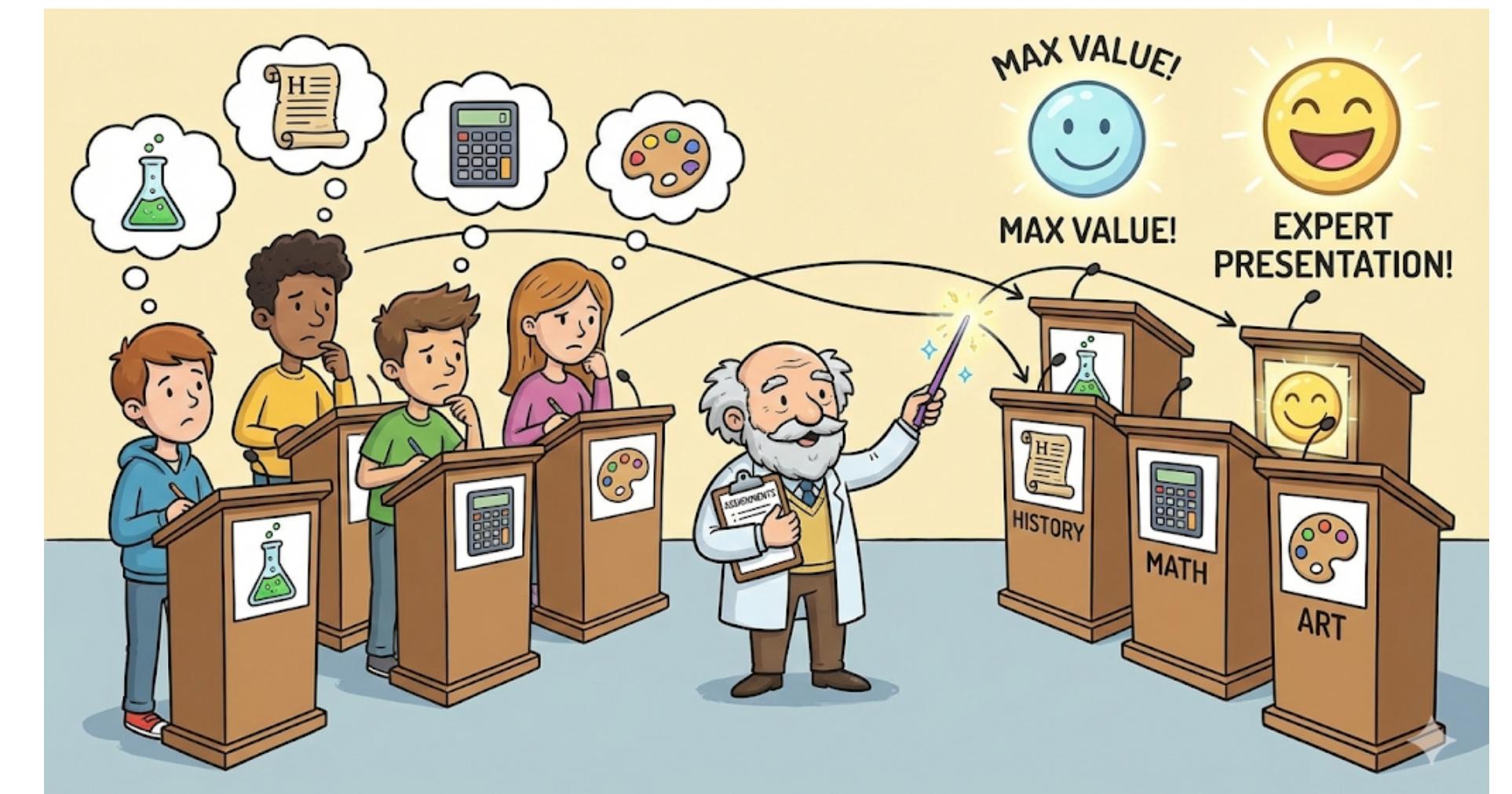
Introduction to Mathematical Programming

Assignment Problem



- 4 students must present 4 topics. Each student has an expertise in specific areas. Therefore, the value of the presentation will be better with the expert. How would you assign the topics to the students?

	Topic A	Topic B	Topic C	Topic D
S1	50	49	1	1
S2	48	2	47	1
S3	2	46	45	1
S4	1	1	100	44



🧠 What about Student 3, it will be a very bad presentation

💡 Change objective function to “maximize the minimum score”

How do we solve optimally?

Introduction to solvers

How do we solve optimally?

Introduction to solvers

- Our job is to translate these problems to a mathematical language

How do we solve optimally?

Introduction to solvers

- Our job is to translate these problems to a mathematical language
- The solvers will find the optimal for us
 - Solver example: Gurobi, Cplex, Fico Xpress, SCIP (open source)
 - Most of them are free for students

How do we solve optimally?

Introduction to solvers

- Our job is to translate these problems to a mathematical language
- The solvers will find the optimal for us
 - Solver example: Gurobi, Cplex, Fico Xpress, SCIP (open source)
 - Most of them are free for students
- Knapsack and Assignment Problem with Gurobi + Python

How do we solve optimally?

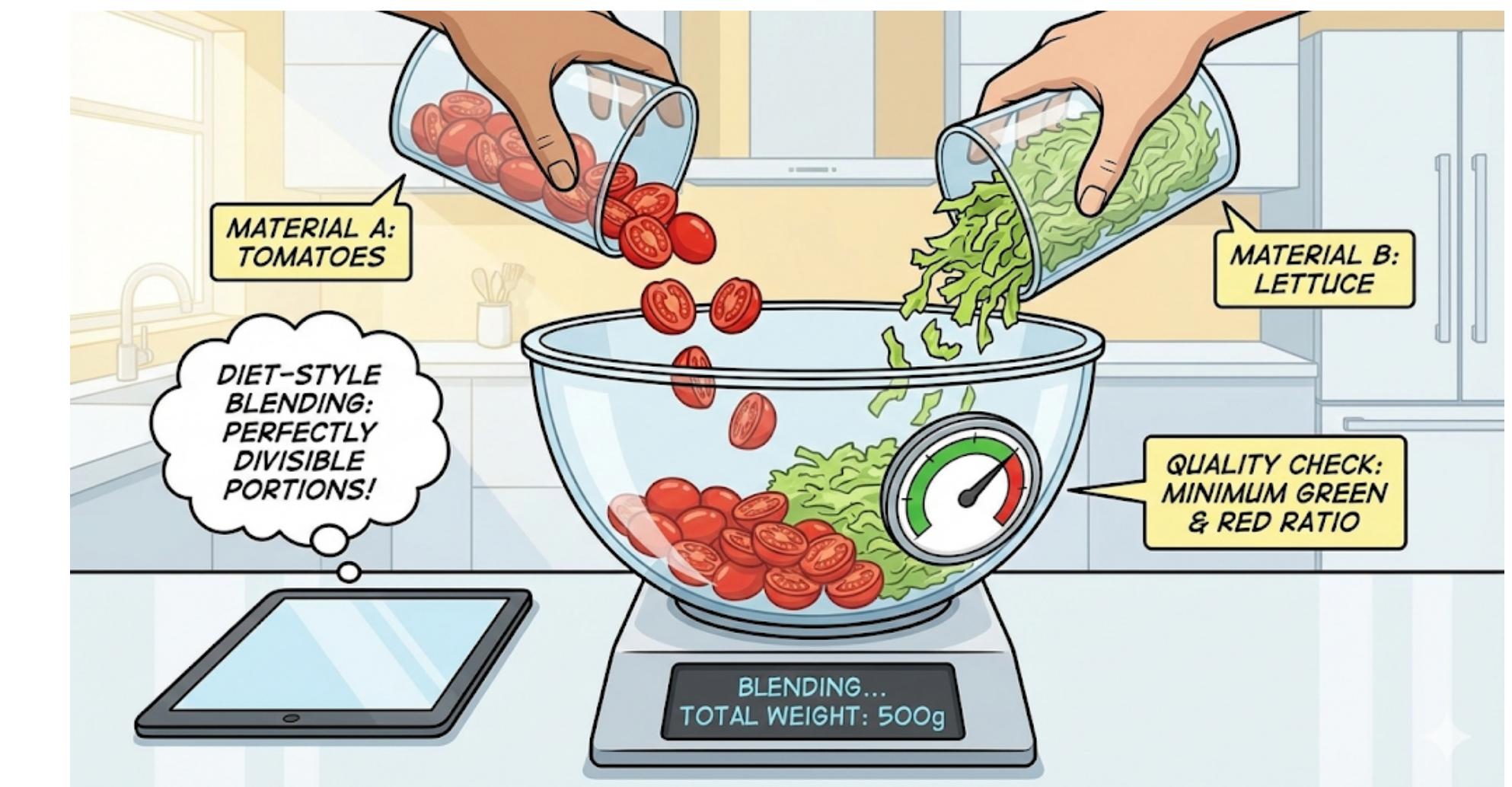
Introduction to solvers

- Our job is to translate these problems to a mathematical language
- The solvers will find the optimal for us
 - Solver example: Gurobi, Cplex, Fico Xpress, SCIP (open source)
 - Most of them are free for students
- Knapsack and Assignment Problem with Gurobi + Python
- So far, we have seen **binary** and **integer** variables!

Introduction to Mathematical Programming

Food Production

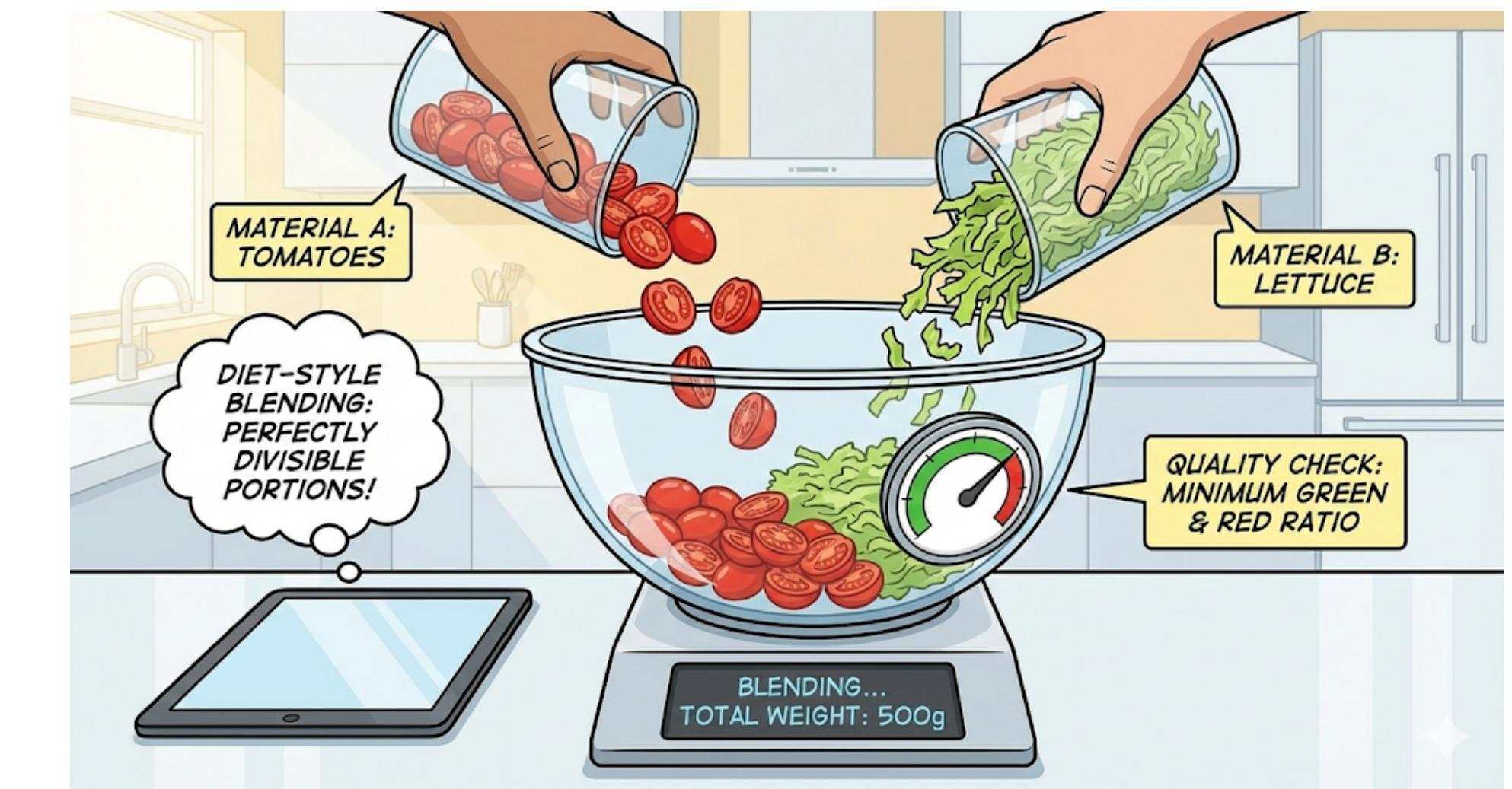
- You're mixing two raw materials to make a 300gr salad. You must meet a minimum quality of 1800. The salad cannot have more than 250gr of tomatoes. You want to minimize the cost.
- What is the objective (function)?
- What is the decision to make?
- What are the constraints?



Introduction to Mathematical Programming

Food Production

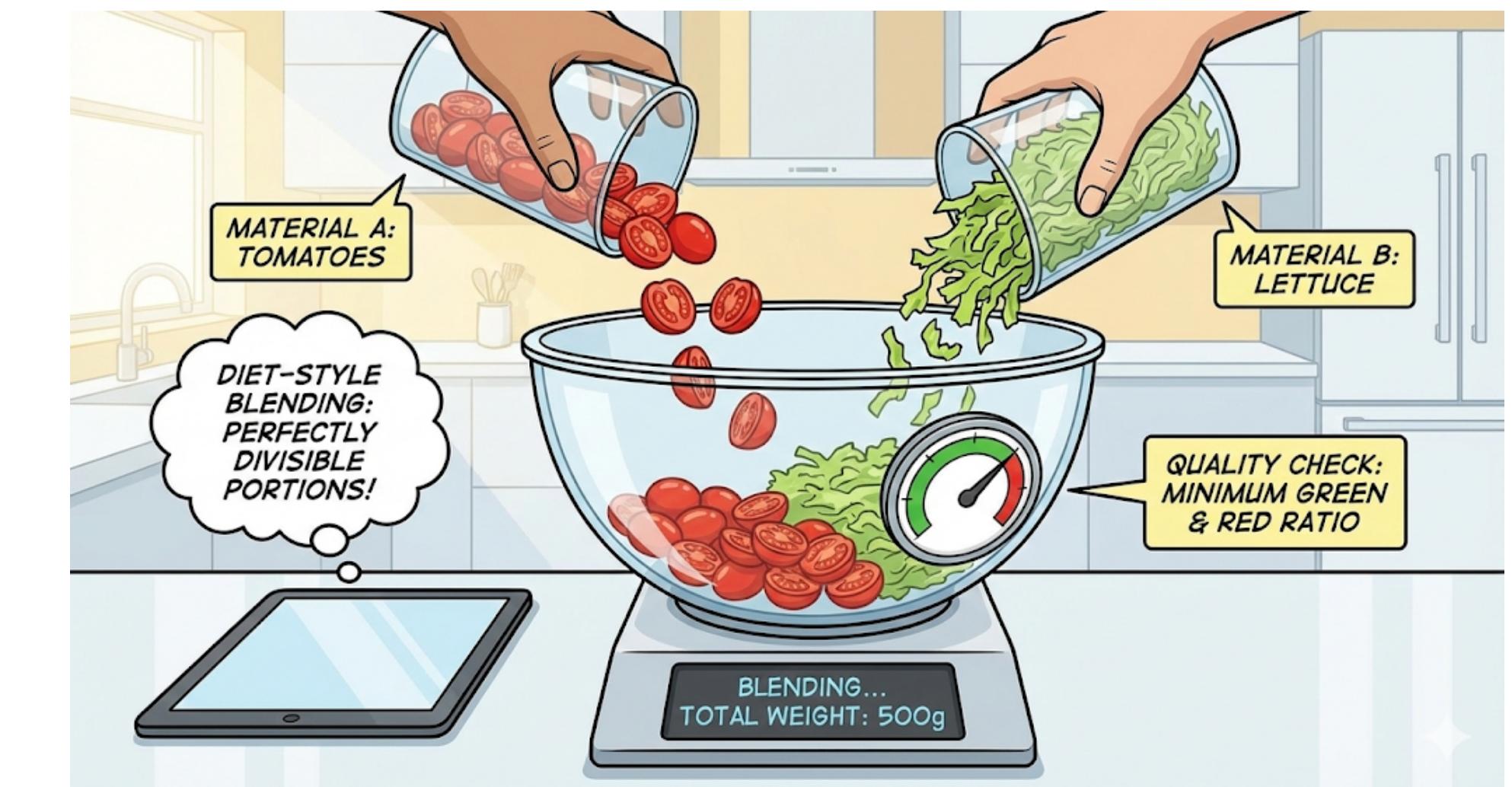
- You're mixing two raw materials to make a 300gr salad. You must meet a minimum quality of 1800. The salad cannot have more than 250gr of tomatoes. You want to minimize the cost.
- What is the objective (function)?
Minimize the total cost
- What is the decision to make?
- What are the constraints?



Introduction to Mathematical Programming

Food Production

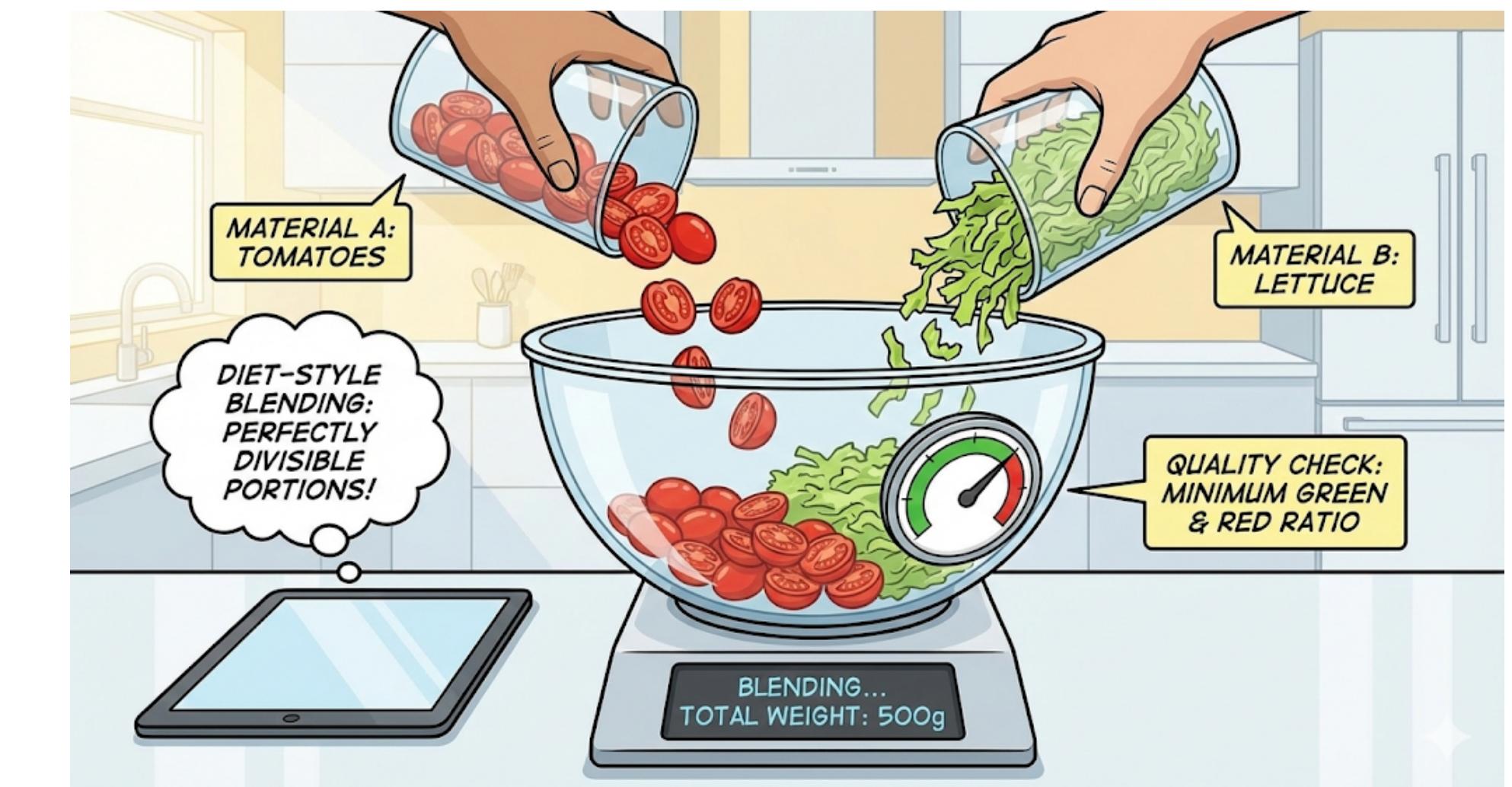
- You're mixing two raw materials to make a 300gr salad. You must meet a minimum quality of 1800. The salad cannot have more than 250gr of tomatoes. You want to minimize the cost.
- What is the objective (function)?
Minimize the total cost
- What is the decision to make?
How much to add from each material
- What are the constraints?



Introduction to Mathematical Programming

Food Production

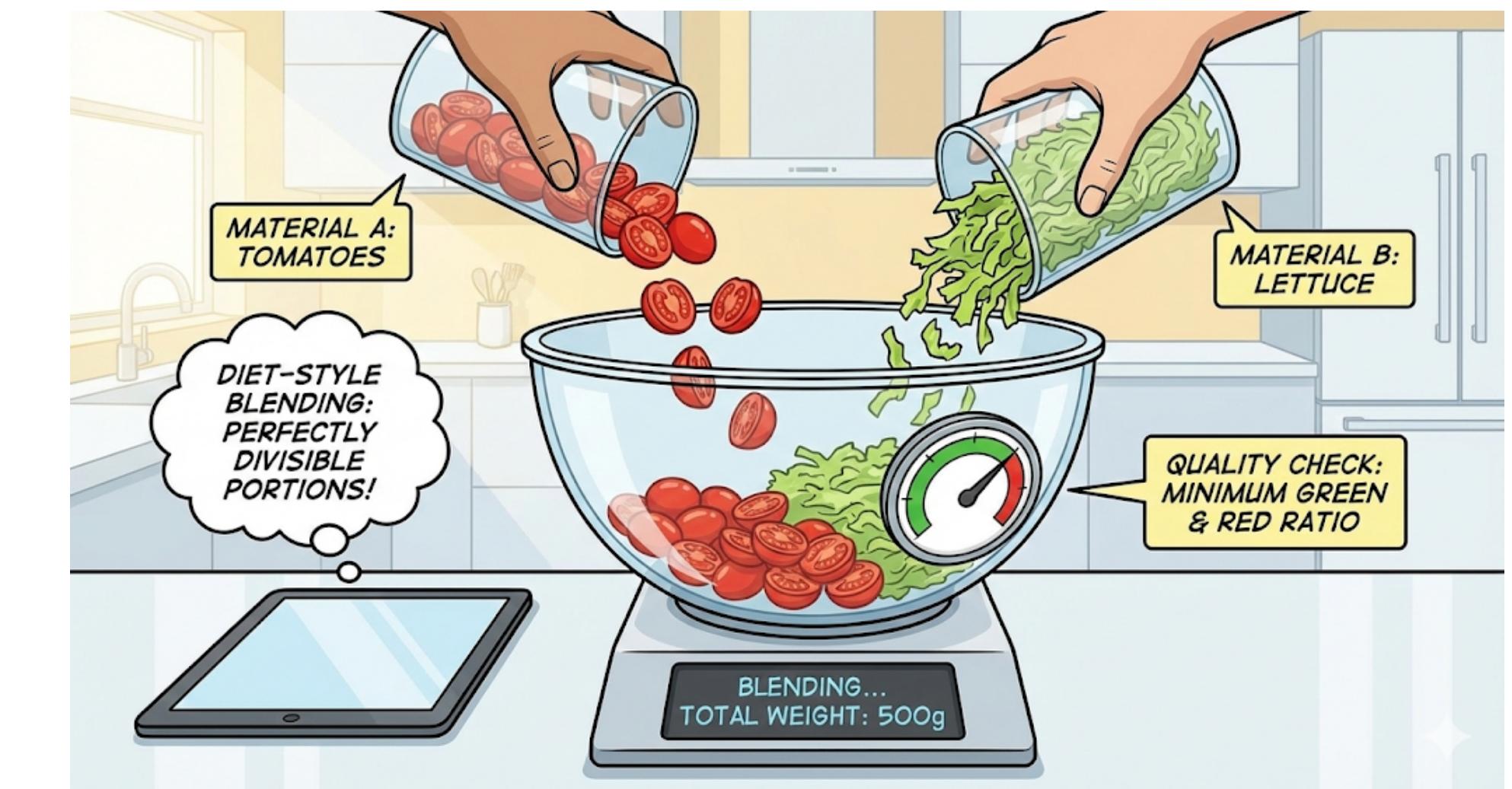
- You're mixing two raw materials to make a 300gr salad. You must meet a minimum quality of 1800. The salad cannot have more than 250gr of tomatoes. You want to minimize the cost.
- What is the objective (function)?
Minimize the total cost
!!!Fractional values are allowed
- What is the decision to make?
How much to add from each material
- What are the constraints?



Introduction to Mathematical Programming

Food Production

- You're mixing two raw materials to make a 300gr salad. You must meet a minimum quality of 1800. The salad cannot have more than 250gr of tomatoes. You want to minimize the cost.
- What is the objective (function)?
Minimize the total cost !!!Fractional values are allowed
- What is the decision to make?
How much to add from each material
- What are the constraints?
 - Meet the minimum quality
 - 300gr of end product
 - Max 250gr tomatoes

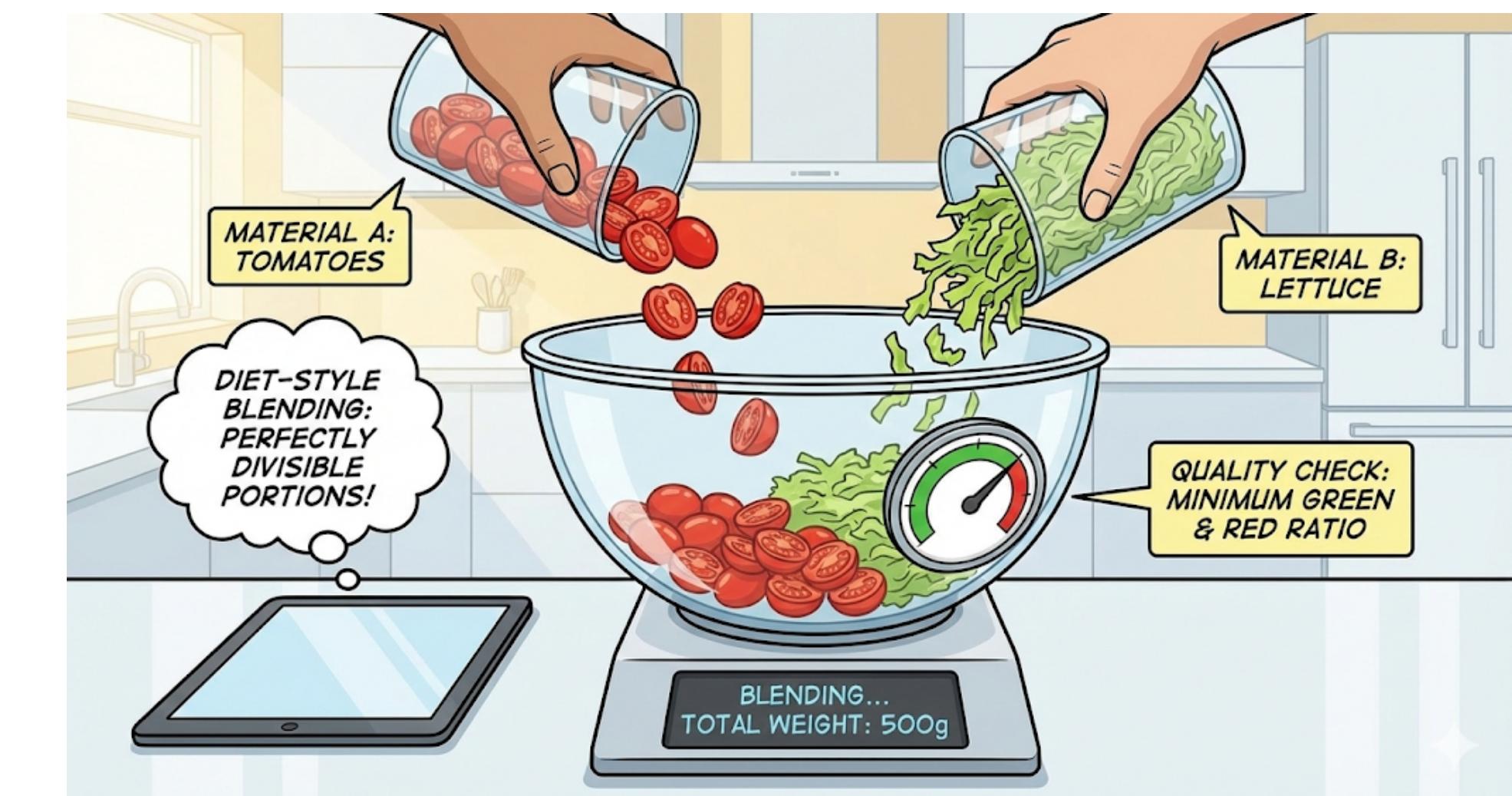


Introduction to Mathematical Programming

Food Production

- You're mixing two raw materials to make a 300gr salad. You must meet a minimum quality of 1800. The salad cannot have more than 250gr of tomatoes. You want to minimize the cost.

	Cost (€/kg)	Quality (unit/g)
Tomato	6	9
Lettuce	2	3



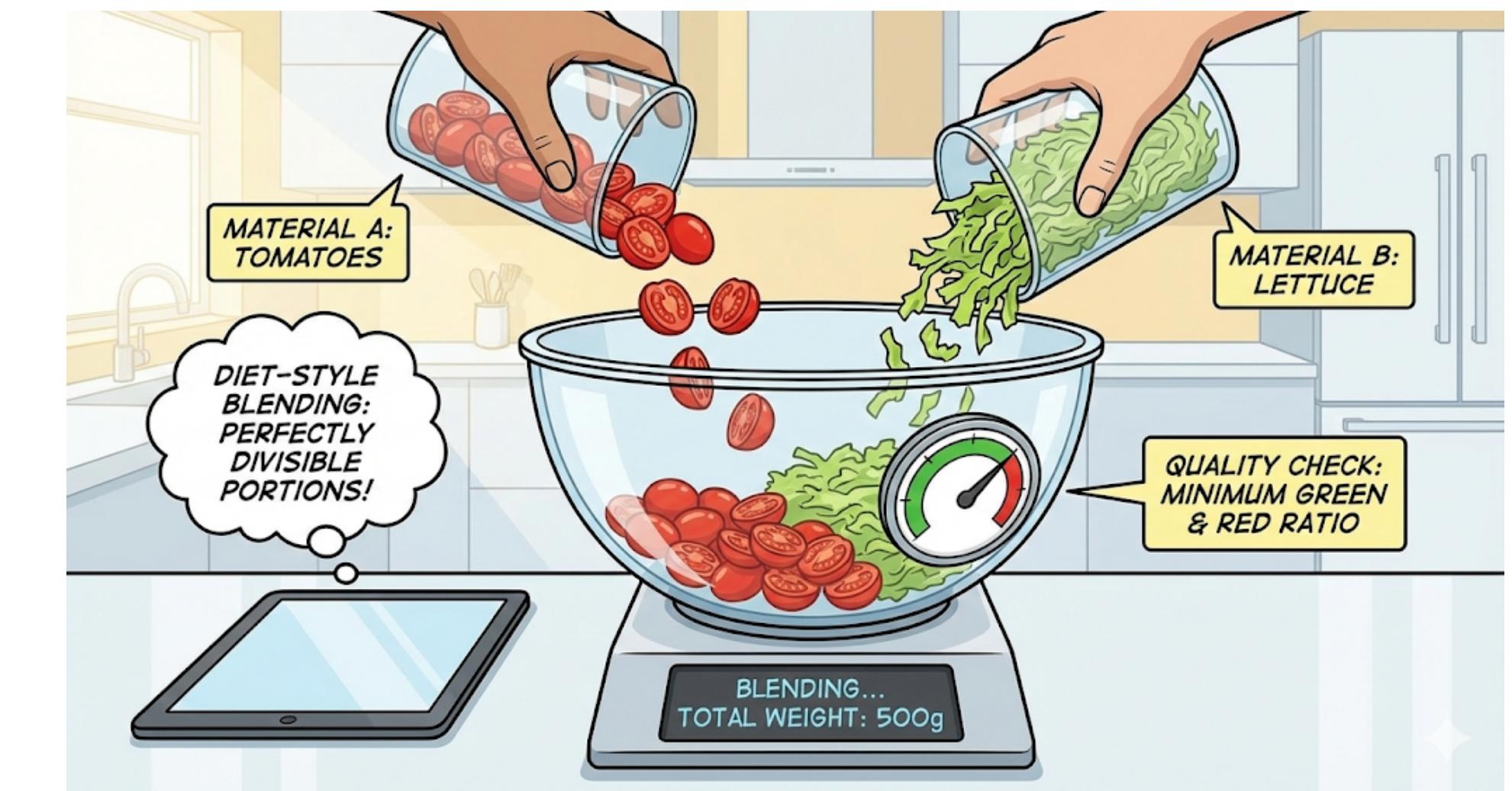
Introduction to Mathematical Programming

Food Production

- You're mixing two raw materials to make a 300gr salad. You must meet a minimum quality of 1800. The salad cannot have more than 250gr of tomatoes. You want to minimize the cost.

	Cost (€/kg)	Quality (unit/g)
Tomato	6	9
Lettuce	2	3

How do we solve this now?



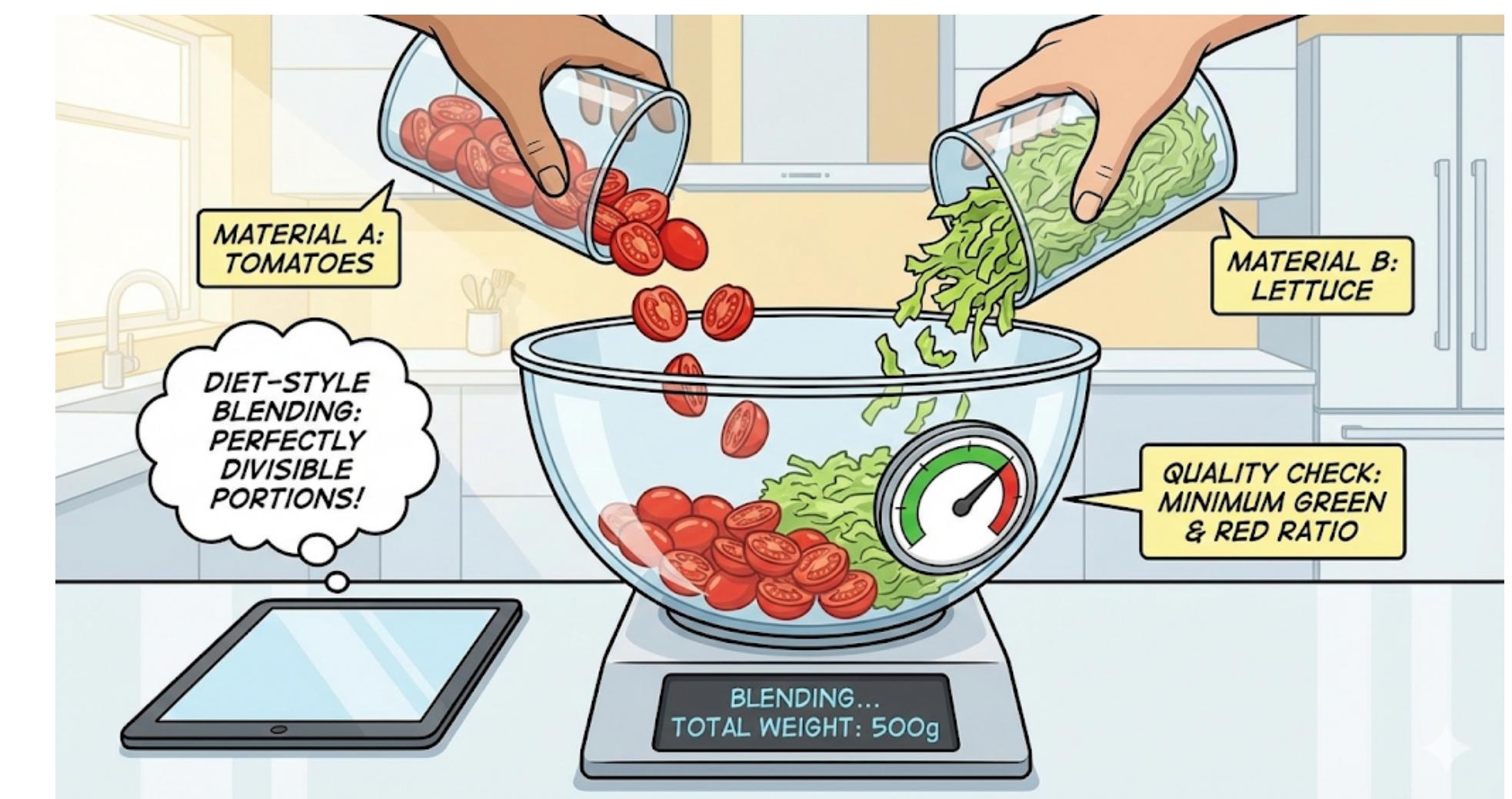
Introduction to Mathematical Programming

Food Production

- You're mixing two raw materials to make a 300gr salad. You must meet a minimum quality of 1800. The salad cannot have more than 250gr of tomatoes. You want to minimize the cost.

	Cost (€/kg)	Quality (unit/g)
Tomato	6	9
Lettuce	2	3

How do we solve this now?



Although it is not easy to solve intuitively, easier for solvers:

- The solution space has structure: no holes or disconnected pieces
- Linearity: no local traps
- Continuous variables: no enumerating combinations

Introduction to Mathematical Programming

Food Production

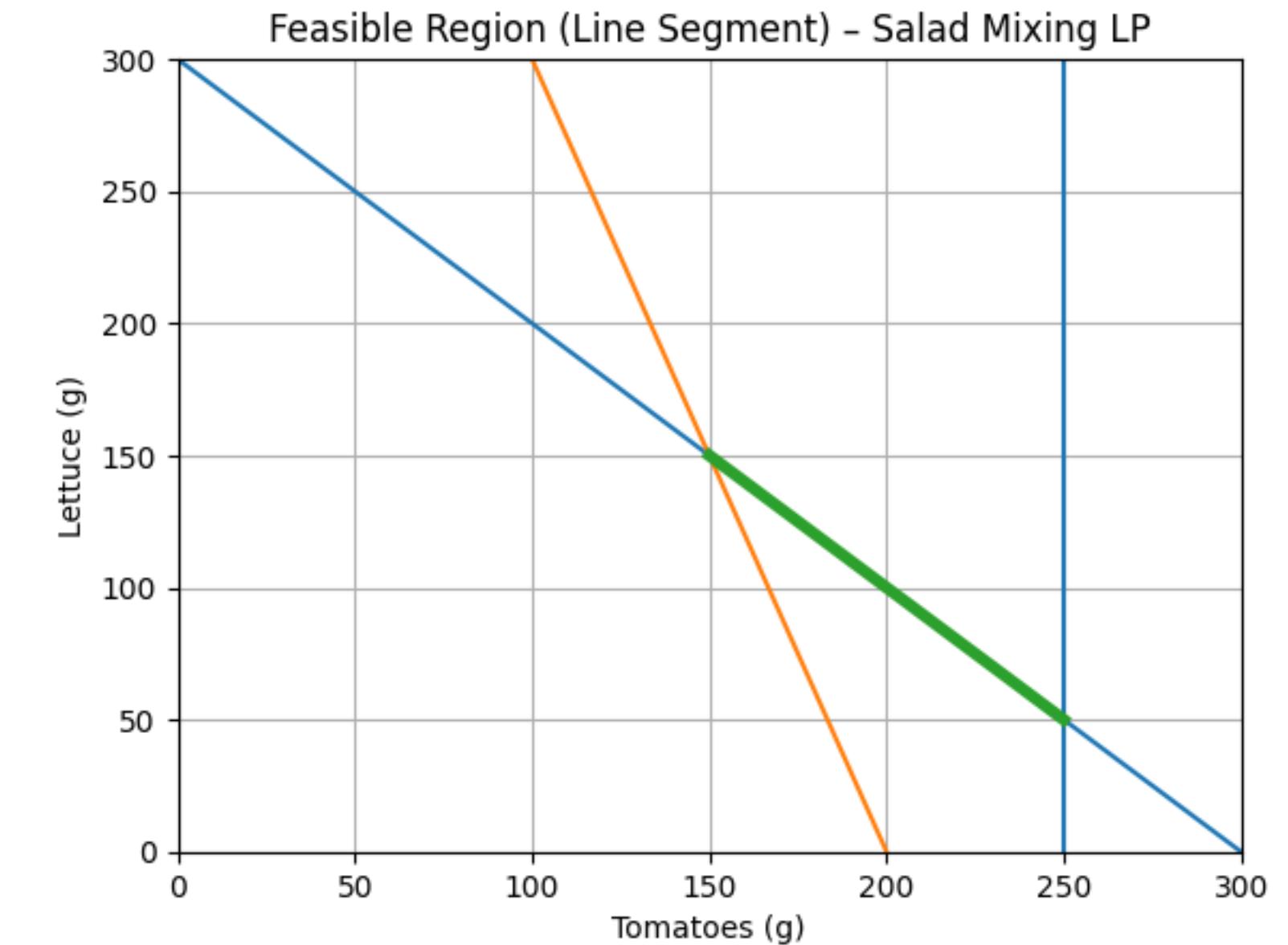
- You're mixing two raw materials to make a 300gr salad. You must meet a minimum quality of 1800. The salad cannot have more than 250gr of tomatoes. You want to minimize the cost.

	Cost (€/kg)	Quality (unit/g)
Tomato	6	9
Lettuce	2	3

How do we solve this now?

Although it is not easy to solve intuitively, easier for solvers:

- The solution space has structure: no holes or disconnected pieces
- Linearity: no local traps
- Continuous variables: no enumerating combinations



Introduction to Mathematical Programming

Food Production

at least

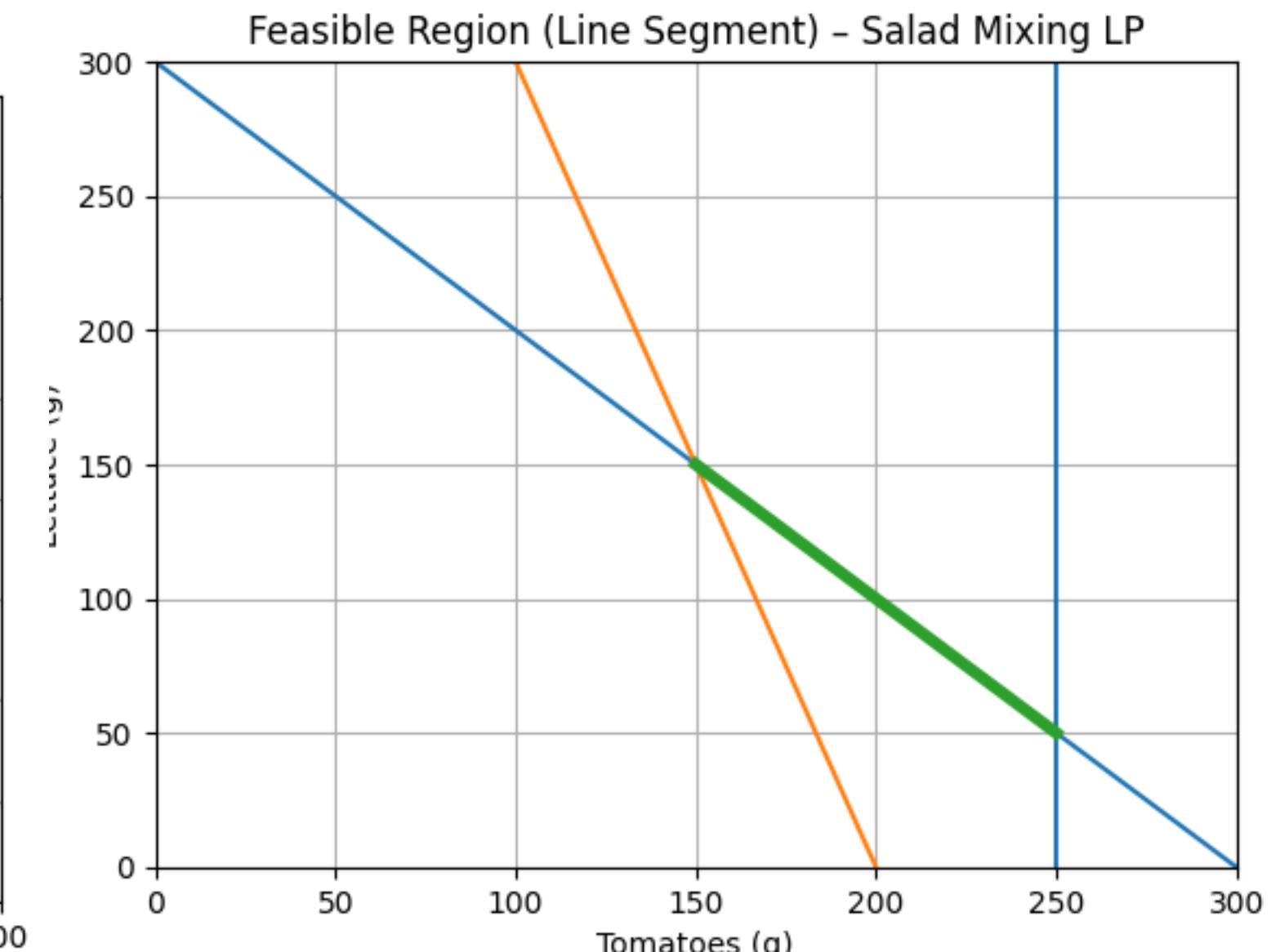
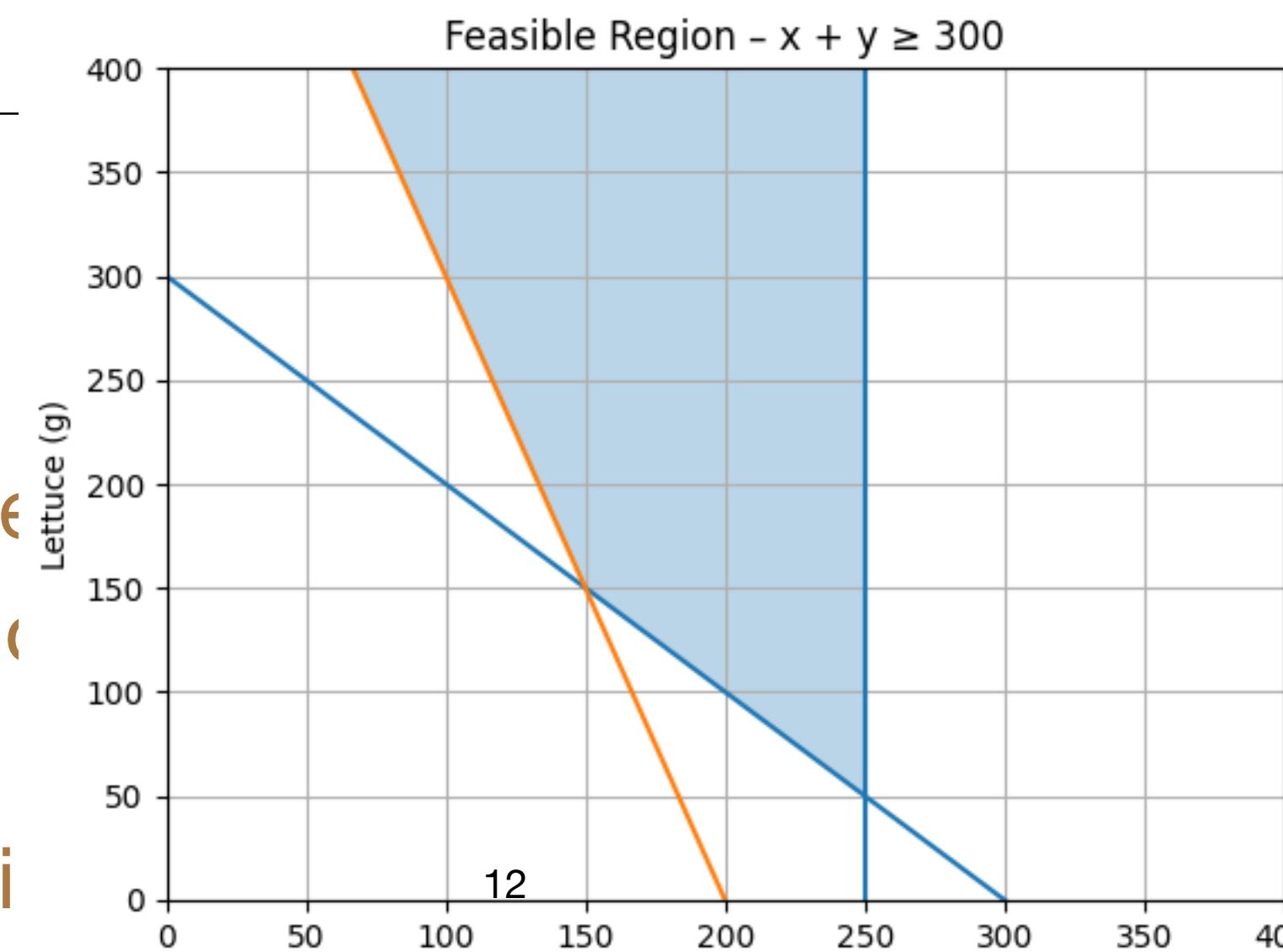
- You're mixing two raw materials to make a 300gr salad. You must meet a minimum quality of 1800. The salad cannot have more than 250gr of tomatoes. You want to minimize the cost.

	Cost (€/kg)	Quality (unit/g)
Tomato	6	9
Lettuce	2	

How do we solve this now?

Although it is not easy to solve intuitively

- The solution space has structure: no local optima
- Linearity: no local traps
- Continuous variables: no enumeration



Gurobi Setup



- Registration:
 - <https://tinyurl.com/4t45xhf7>
 - <https://portal.gurobi.com/iam/login/>
- Download: <https://www.gurobi.com/downloads/gurobi-software/>
- Installing & licensing
 - grbgetkey 6b085692-feba-11e8-83af-02e454ff9c50
- Download and install Python: <https://www.python.org/downloads/release/python-3116/>
- Install Gurobi Python pkg: `python -m pip install gurobipy`
- (Optional) to enable type hints: `python -m pip install gurobipy-stubs`

Gurobi Setup



- Download: Lab1 - gurobi examples.ipynb
- JupyterLab: <https://jupyter.org/install>
- In your terminal:
 - `jupyter lab`
- Go to the browser and open Lab1 - gurobi examples.ipynb
- Many other examples: <https://docs.gurobi.com/projects/examples/en/current/examples/python.html>

Gurobi with IDE



- Download VS Code: <https://code.visualstudio.com/Download>
- Download diet.py: <https://docs.gurobi.com/projects/examples/en/current/examples/python/diet.html> or copy + past to diet.py
- Open VS Code and open the diet.py file
- VS Code will suggest you to install Python Extension, accept it.
- Any problem about python issues with VS Code: https://code.visualstudio.com/docs/python/python-tutorial#_select-a-python-interpreter
- At the top right you will find the play symbol . Press it to execute your code. The Terminal will open and execute the code. You will directly see the output or any errors.

What to remember

Lab1

What to remember

Lab1

- How do we define a problem?
 - Objective function
 - Decisions
 - Constraints

What to remember

Lab1

- How do we define a problem?
 - Objective function
 - Decisions
 - Constraints
- Different objective functions: *min*, *max*, *max min*

What to remember

Lab1

- How do we define a problem?
 - Objective function
 - Decisions
 - Constraints
- Different objective functions: *min*, *max*, *max min*
- Different decision variables: *binary*, *integer*, *continuous*

What to remember

Lab1

- How do we define a problem?
 - Objective function
 - Decisions
 - Constraints
- Different objective functions: *min*, *max*, *max min*
- Different decision variables: *binary*, *integer*, *continuous*
- Different constraints types: \leq , $=$, \geq