

# NETB507 - Assignment Problem

Rumen Yavorov

F84719

February 2023

# Outline

NETB507 -  
Assignment  
Problem

Rumen  
Yavorov

## Overview

Initialize agents and  
tasks

Input weights for  
each task by agent

Steps of the  
algorithm

## 1 Overview

- Initialize agents and tasks
- Input weights for each task by agent
- Steps of the algorithm

# Initialize agents and tasks

NETB507 -  
Assignment  
Problem

Rumen  
Yavorov

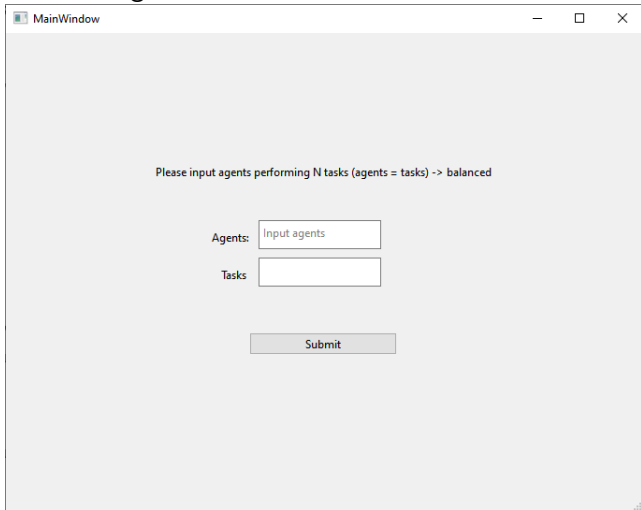
## Overview

Initialize agents and  
tasks

Input weights for  
each task by agent

Steps of the  
algorithm

## ■ Initialize agents and tasks



MainWindow

Please input agents performing N tasks (agents = tasks) -> balanced

Agents:

Tasks:

Submit

# Input weights for each task by agent I

NETB507 -  
Assignment  
Problem

Rumen  
Yavorov

## Overview

Initialize agents and  
tasks

Input weights for  
each task by agent

Steps of the  
algorithm

## ■ Input weights for each task by agent

Form

	Task - 1	Task - 2	Task - 3	Task - 4
Agent - 1	Please set cost	Please set cost	Please set cost	Please set cost
Agent - 2	Please set cost	Please set cost	Please set cost	Please set cost
Agent - 3	Please set cost	Please set cost	Please set cost	Please set cost
Agent - 4	Please set cost	Please set cost	Please set cost	Please set cost

Submit values

# Input weights for each task by agent II

NETB507 -  
Assignment  
Problem

Rumen  
Yavorov

## Overview

Initialize agents and  
tasks

Input weights for  
each task by agent

Steps of the  
algorithm

- Input weights, Then click 'Submit values'

	Task - 1	Task - 2	Task - 3	Task - 4
Agent - 1	10	12	19	11
Agent - 2	5	10	7	8
Agent - 3	12	14	13	11
Agent - 4	8	15	11	9

Submit values

# Input weights for each task by agent III

NETB507 -  
Assignment  
Problem

Rumen  
Yavorov

## Overview

Initialize agents and  
tasks

Input weights for  
each task by agent

Steps of the  
algorithm

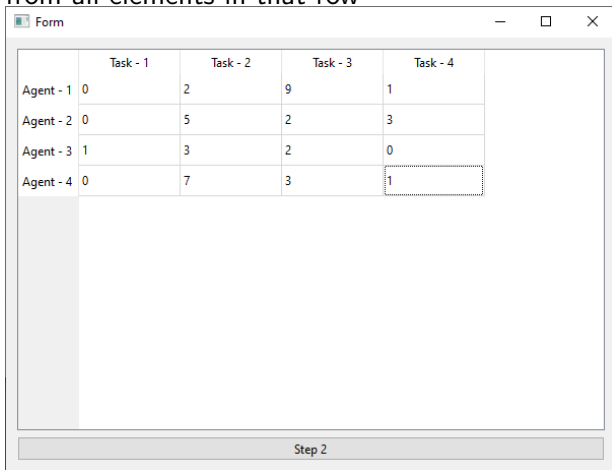
## ■ Step buttons will appear

	Task - 1	Task - 2	Task - 3	Task - 4
Agent - 1	10	12	19	11
Agent - 2	5	10	7	8
Agent - 3	12	14	13	11
Agent - 4	8	15	11	9

Step 1

# Steps of the algorithm I

- Step 1 - Find lowest element in each row and subtract it from all elements in that row



	Task - 1	Task - 2	Task - 3	Task - 4
Agent - 1	0	2	9	1
Agent - 2	0	5	2	3
Agent - 3	1	3	2	0
Agent - 4	0	7	3	1

Step 2

NETB507 -  
Assignment  
Problem

Rumen  
Yavorov

## Overview

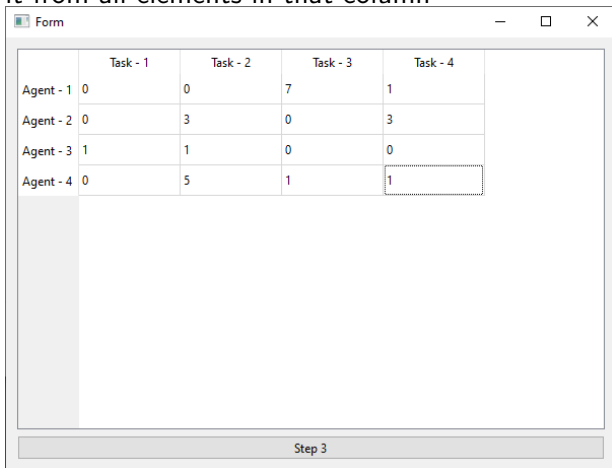
Initialize agents and  
tasks

Input weights for  
each task by agent

Steps of the  
algorithm

# Steps of the algorithm II

- Step 2 - Find lowest element in each column and subtract it from all elements in that column



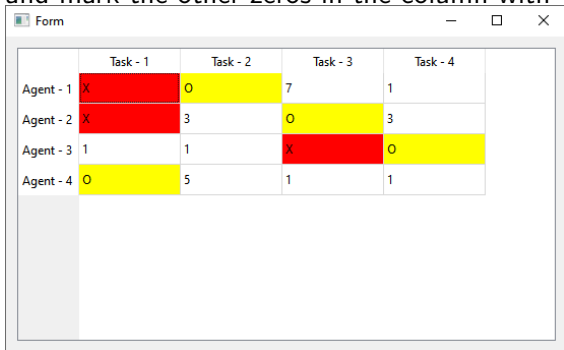
	Task - 1	Task - 2	Task - 3	Task - 4
Agent - 1	0	0	7	1
Agent - 2	0	3	0	3
Agent - 3	1	1	0	0
Agent - 4	0	5	1	1

Step 3



# Steps of the algorithm III

- Step 3 - Find the only zero in row and mark it with 'O' and mark the other zeros in the column with 'X'



The screenshot shows a window titled 'Form' containing a table with 4 rows (Agents) and 4 columns (Tasks). The table is as follows:

	Task - 1	Task - 2	Task - 3	Task - 4
Agent - 1	X	0	7	1
Agent - 2	X	3	0	3
Agent - 3	1	1	X	0
Agent - 4	0	5	1	1

In the table, 'X' marks are in cells (1,1), (2,1), and (3,3). '0' marks are in cells (1,2), (2,3), and (3,4). Cells (1,2) and (2,3) are highlighted in yellow. Cell (3,4) is highlighted in yellow. Cell (1,1) is highlighted in red. Cell (3,3) is highlighted in red.

- The algorithm should continue to execute 'Step 3' till it examines every row and column then return optimal schedule