Sabanci University

Faculty of Engineering and Natural Sciences
CS204 Advanced Programming
Summer 2022-2023

Take-Home Exam 1 – Space Colony Management
Due: 28 July 2023 11.55pm (SHARP)

DISCLAIMER:

Only checking the sample run cases might not be sufficient as your solution will be checked against a variety of samples different from the provided samples; however checking and studying these cases is highly encouraged and recommended.

You can NOT collaborate with your friends and discuss your solutions with each other. You have to write down the code on your own. Plagiarism will not be tolerated AND cooperation is not an excuse!

Aim

The aim of this take-home exam (THE) is to practice and recall some of the CS201 material. For that you are encouraged to review the following topics before you delve into the details of this THE:

- Structs
- Vector containers (1D and 2D)
- File handling
- Stringstream
- Functions: pass by value and pass by reference

Introduction

In the not-so-distant future, humans have successfully established a space colony on a distant planet. As part of the colony management team, you are responsible for developing a program that assists in managing the resources and infrastructure of the colony.

Your program will read data from 3 different text files, and your program should store all the data in appropriate data structures (containers) so that it can smoothly achieve certain tasks.

In a nutshell, you have a list of resources, each with its available quantity. Also, a list of building types that can be built in this colony along with a list of quantities corresponding to the amount of each resource that this building type needs for its construction. Finally, you are provided with an initial setup of the colony which you may continue your program with.

Note on implementation: You ***must*** use containers of type `vector` in this THE. For example, you can store the colony in a vector of vectors of some `struct` from your own design. Also, the stock and the consumption information needs to be stored in `vector` containers from your own design.

You must use, store, and manipulate all the available data using `vector` containers. Otherwise -if you avoid this restriction- your submission will be graded as zero (0).

Input files and User Inputs

Your program will read inputs from 3 different text files, usually having the following naming conventions:

1. *stockX.txt*
    - e.g., stock1.txt, stock2.txt ….
    - This type of file includes available resources in the following format

    ```
    resource1_name resource1_quantity
    resource2_name resource2_quantity
    .
    .
    resourceN_name resourceN_quantity
    ```

    - In each line of this file, every resource_name is a one-word string, while every resource_quantitiy is a non-negative integer number.
    - In each line of this file, every resource_name and resource_quantitiy pair on the same line are separated with a whitespace.
    - There won’t be any duplicate resource names or duplicate pairs of resources_names and resource quantities. i.e., every resource name (with its corresponding quantity) will appear only once in a single file.

2. *consumptionX.txt*

- e.g., consumption1.txt, consumption2.txt …
- This type of text file has the types of buildings that can be built in the current colony, along with lists of *N* non-negative integers that indicate the amount of each of the *N* resources required to build a block (one cell on the colony) of that type of building.

```
buildingType1 required_quantity_resource1 … required_quantity_resourceN
buildingType2 required_quantity_resource1 … required_quantity_resourceN
.
.
```

- Each `buildingType` is a single uppercase English letter.
- For each `buildingType`, its corresponding list of *N* non-negative integers indicates the amount of each of the *N* resources required to build a block of that given building type (*following the same order of resource names given in the first text file that contains the available resources information (i.e., stockX.txt)*).
- There won't be any duplicate building types or duplicate pairs of building types and required resources quantities. i.e., every building type (with its corresponding required resources quantities) will appear only once in a single file.

3. *colonyX.txt*
    - e.g., colony1.txt, colony2.txt,....
    - This type of text file will have a matrix representation of the buildings that will occupy some blocks (cells) of the colony matrix when the colony is to be loaded.
    - Each cell in the matrix is going to be either '-', which indicates that this specific cell is empty, or any other uppercase letter, which indicates that this cell is occupied by a block of that building type.
        - All the uppercase letters that are found in the colony file are also mentioned in the second file (*consumptionX.txt*), where their resources consumption was mentioned.

None of the files will have extra whitespaces or empty lines. That is, you don't have to check for such irregularities.

Note on User Inputs case-sensitivity:

After successfully reading, and displaying the contents of the three files, your program should provide the user with a menu-driven interface (details are in the sections below). During the user interaction with the menu, the user will enter some inputs. Your program should treat all the input characters (letters) and strings in a case-sensitive manner. I.e., if the program expects an uppercase character ‘D’, and the user accidentally enters 'd’, your program should consider ‘d’ as invalid input. Please refer to sample runs for more examples.

Program Flow

<u>You are encouraged to carefully check the sample runs as they complement the textual description, demonstrate different cases that your code should handle, and they also show the expected behavior of your program.</u>

After printing the welcoming message to the screen, your program needs to read data from 3 text files mentioned above.
The process for the first two files are almost the same. Your program basically should:

- start by asking the user to enter the file name (keep asking if the file name is not correct)
- open this file, read and store its content in an appropriate container,
- print a specific message to indicate that the file has been read,
- and finally, print the content of the file.

The above steps need to be followed for reading the first two files’ contents (i.e., *stockX.txt*, and *consumptionX.txt*). For distinguishing messages for each of the two files, please refer to the sample runs.

Then, we come to the 3rd file (*colonyX.txt*), which contains the colony’s initial setup. When reading the content of this file, and trying to construct the buildings, your program will be using (consuming) resources from the resources stock, and therefore, you are limited to the initial resource quantities that you read from the resource stock (*stockX.txt*) in the very beginning. Thus, every building to be constructed will consume a portion of the resources that it requires to be built (consumption quantities of resources for each building type are given in the *consumptionX.txt* text file).

If any of the resources was not sufficient for constructing any building in the given initial colony, your program should print specific error messages and exit without further executing the rest of the code.

If, on the other hand, your program manages to load the colony (i.e. there were enough resources for all the buildings in the colony), then your program should proceed with the following:

- Displaying the colony
    - You should display the colony as a 2D matrix shape, and in order to have a well-structured output, please use the following code snippet (from `iomanip`) while displaying the colony:
        - `std::setw(5) << std::left`
- Printing the resource stock
    - The structure of the output follows the same structure as the resource stock file (i.e., stockX.txt).
    - After loading the colony, you can notice the difference between the original and initial resource stock contents and the current resource stock content as a result of loading the colony and consuming some resources to build the buildings that were on the colony.

- Displaying the consumption of all resources
    - Here, you need to display as many matrix-shaped outputs as the number of resources that you have in the stock.
        - i.e, for every resource, you will print the structure of the colony, where each cell represents the consumption of this resource by the building on this cell, if any.
    - Again, in order to have a well-structured output, please use the following code snippet (from `iomanip`) while displaying the matrix shape:
        - `std::setw(5) << std::left`

Please note that, in all file opening procedures, if the user enters a wrong file name, your code should keep asking until the user provides a valid file name.

After reading the contents of each file, your program should print a message that indicates the data was read, and display that content of that file. Please refer to the sample runs for specific output messages and format.

While reading the files mentioned above, your program needs to store them in appropriate `vector` containers.

Note that, while loading the colony, your program will probably consume some resources that are required to load the buildings on that colony. So, your program needs to handle that and consider its effects.
Also, be assured that all the building types that are initially in the colony file are present in the corresponding consumption file. I.e., you don’t need to validate whether the building types in the initial colony setup are valid or not.

After that, your program should start displaying a menu with 7 different options (as shown in the sample runs). The user will enter an option number, and based on this number, the corresponding functionality needs to be performed.
Below is the list of 7 options with details.

1. Construct a new building on the colony.

    - If there is at least one empty cell on the colony, the user is allowed to choose a type of building from the list mentioned in the corresponding text file -*consumptionX.txt*.
        - If the user enters an invalid building type, your program should print a message indicating that, and ask the user to enter a valid building type. Please refer to the sample runs for the exact message.
    - Then, the user will be asked to enter the number of blocks (cells) that the building will occupy.
        - For every cell, the user will be asked to enter the row and column index of this cell.
    - Every construction on a block/cell will consume the corresponding amount of resources mentioned for the given building type in the *consumptionX.txt* file.

2. Destruct/Disassemble a building from the colony.

    - If there is at least one occupied cell/block by a building in the colony, the user can choose to destruct/disassemble a building by entering its row and column index.
    - When destroying a building, the resources that were used to construct it in the first place have to be returned to the resources stock.

Note: All the row and column indices entered by the user as input during the execution of the above two options have to be validated. Please refer to the sample runs for examples.

3. Print the colony.

    - This functionality has been explained above.

4. Print the consumption of all resources by each building in the colony.

    - This functionality has been explained above.

5. Print the consumption of a specific resource by each building in the colony.

   - Ask the user to enter a valid resource name, and then display the map (layer) of its consumption by each block in the colony.
     - If the user entered invalid resource name, your program should print a specific message indicating that the resource was not found in the resources stock, and then skip the current option and continue the (7-options) loop of asking to enter a new option number. Again, for the exact messages, please refer to the sample runs.

6. Print the resources stock.

   - This functionality has been explained above.

7. Exit the program.

   - Your program should show a couple of ending messages and then terminate the program.

Sample Runs

Below, we provide some sample runs of the program that you will develop. The **bold** phrases are the standard input (cin) taken from the user (i.e., like **this**). You have to display the required information in the same order and with the same words/spaces as here; in other words, there must be an exact match!

You will submit your code through CodeRunner, and you can always use the “Show differences” feature to make it easier for you to figure out the mismatches in your output, if any.

Sample run 1:

```
Welcome to the colony management system
Please enter file name for resources stock:
stock5.txt
Available resources loaded from stock5.txt
Resource stock:
Iron 500
Copper 0
Silicon 1000
Please enter file name for resource consumption per building type:
consumption5.txt
Resources consumption per building type loaded from consumption5.txt
Resources consumption per building type:
H 500 1 0
S 2 0 100
C 501 0 0
B 1 1 1
A 0 0 0
E 500 0 999
R 1 0 1001
N 500 0 1000
P 499 0 999
Please enter file name for colony:
colony5.txt
Colony loaded from colony5.txt
Colony:
-       -       -       -       -       -       -       -       -
-       -       -       -       -       -       -       -       -
-       -       -       -       -       -       -       -       -
-       -       -       -       -       -       -       -       -
-       -       -       -       -       -       -       -       -
-       -       -       -       -       -       -       -       -
-       -       -       -       -       -       -       -       -
-       -       -       -       -       -       -       -       -
-       -       -       -       -       -       -       -       -
-       -       -       -       -       -       -       -       -
Resources stock after loading the colony:
Resource stock:
Iron 500
Copper 0
Silicon 1000
Consumption of resource Iron by each building in the colony:
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
```

```
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
Consumption of resource Copper by each building in the colony:
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
Consumption of resource Silicon by each building in the colony:
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
0       0       0       0       0       0       0       0       0
Please enter an option number:
1. Construct a new building on the colony.
2. Destruct/Disassemble a building from the colony.
3. Print the colony.
4. Print the consumption of all resources by each building in the colony.
5. Print the consumption of a specific resource by each building in the colony.
6. Print the resources stock.
7. Exit the program.
3
Colony:
–       –       –       –       –       –       –       –       –
–       –       –       –       –       –       –       –       –
–       –       –       –       –       –       –       –       –
–       –       –       –       –       –       –       –       –
–       –       –       –       –       –       –       –       –
–       –       –       –       –       –       –       –       –
–       –       –       –       –       –       –       –       –
–       –       –       –       –       –       –       –       –
–       –       –       –       –       –       –       –       –
–       –       –       –       –       –       –       –       –
```

```
Please enter an option number:
7
Thank you for using the colony management system. The program will terminate.
Goodbye!
```

Sample run 2:

```
Welcome to the colony management system
Please enter file name for resources stock:
stok3
Unable to open the file stok3 for reading.
Please enter the correct file name:
stook3.txt
Unable to open the file stook3.txt for reading.
Please enter the correct file name:
stock3.text
Unable to open the file stock3.text for reading.
Please enter the correct file name:
stock3.txt
Available resources loaded from stock3.txt
Resource stock:
Biofuel 2000
Self-ReplicatingBots 5
SolarPanels 800
Hydrogel 300
Nanomaterials 150
Please enter file name for resource consumption per building type:
consump3.txt
Unable to open the file consump3.txt for reading.
Please enter the correct file name:
consumption3
Unable to open the file consumption3 for reading.
Please enter the correct file name:
consumption3.text
Unable to open the file consumption3.text for reading.
Please enter the correct file name:
consumption3.txt
Resources consumption per building type loaded from consumption3.txt
Resources consumption per building type:
U 2500 21 45 19 0
E 50 6 16 29 34
J 2000 5 800 300 150
M 100 5 500 200 50
D 0 0 0 0 0
```

```
Please enter file name for colony:
coolony3.txt
Unable to open the file coolony3.txt for reading.
Please enter the correct file name:
colony3.txtt
Unable to open the file colony3.txtt for reading.
Please enter the correct file name:
colony3.txt
Colony loaded from colony3.txt
Colony:
-    -    -    -    -    D
Resources stock after loading the colony:
Resource stock:
Biofuel 2000
Self-ReplicatingBots 5
SolarPanels 800
Hydrogel 300
Nanomaterials 150
Consumption of resource Biofuel by each building in the colony:
0    0    0    0    0    0
Consumption of resource Self-ReplicatingBots by each building in the colony:
0    0    0    0    0    0
Consumption of resource SolarPanels by each building in the colony:
0    0    0    0    0    0
Consumption of resource Hydrogel by each building in the colony:
0    0    0    0    0    0
Consumption of resource Nanomaterials by each building in the colony:
0    0    0    0    0    0
Please enter an option number:
1. Construct a new building on the colony.
2. Destruct/Disassemble a building from the colony.
3. Print the colony.
4. Print the consumption of all resources by each building in the colony.
5. Print the consumption of a specific resource by each building in the colony.
6. Print the resources stock.
7. Exit the program.
2
Please enter the row and the column index of the cell that you want to remove:
1 1
Invalid row or column index, please enter a valid row and column index:
0 6
Invalid row or column index, please enter a valid row and column index:
6 0
Invalid row or column index, please enter a valid row and column index:
```

```
0 0
The cell is already empty, please enter the row and the column index of another cell:
0 5
The building is removed, and the corresponding resources are added back to the stock.
Please enter an option number:
3
Colony:
-    -    -    -    -    -
Please enter an option number:
4
Consumption of resource Biofuel by each building in the colony:
0    0    0    0    0    0
Consumption of resource Self-ReplicatingBots by each building in the colony:
0    0    0    0    0    0
Consumption of resource SolarPanels by each building in the colony:
0    0    0    0    0    0
Consumption of resource Hydrogel by each building in the colony:
0    0    0    0    0    0
Consumption of resource Nanomaterials by each building in the colony:
0    0    0    0    0    0
Please enter an option number:
9
Invalid option number.Please enter an option number:
6
Resource stock:
Biofuel 2000
Self-ReplicatingBots 5
SolarPanels 800
Hydrogel 300
Nanomaterials 150
Please enter an option number:
5
Please enter the type of the resource:
Bioofuel
The resource Bioofuel was not found in the resources stock.
Please enter an option number:
5
Please enter the type of the resource:
Biofuel
Consumption of resource Biofuel by each building in the colony:
0    0    0    0    0    0
Please enter an option number:
0
Invalid option number.Please enter an option number:
```

```
n
Invalid option number.Please enter an option number:
M
Invalid option number.Please enter an option number:
1
Please enter the type of the building that you want to construct:
d
Invalid building type, please enter a valid building type:
D
Please enter the number of cells that the building will occupy:
0
Invalid number of cells, please enter a valid number of cells:
7
Invalid number of cells, please enter a valid number of cells:
6
Please enter the row and the column index of the cell number 1:
0 6
Invalid row or column index, please enter a valid row and column index:
0 0
Please enter the row and the column index of the cell number 2:
0 1
Please enter the row and the column index of the cell number 3:
0 2
Please enter the row and the column index of the cell number 4:
0 3
Please enter the row and the column index of the cell number 5:
0 4
Please enter the row and the column index of the cell number 6:
5 0
Invalid row or column index, please enter a valid row and column index:
0 5
The building is added to the colony.
Please enter an option number:
3
Colony:
D    D    D    D    D    D
Please enter an option number:
6
Resource stock:
Biofuel 2000
Self-ReplicatingBots 5
SolarPanels 800
Hydrogel 300
Nanomaterials 150
```

```
Please enter an option number:
4
Consumption of resource Biofuel by each building in the colony:
0    0    0    0    0    0
Consumption of resource Self-ReplicatingBots by each building in the colony:
0    0    0    0    0    0
Consumption of resource SolarPanels by each building in the colony:
0    0    0    0    0    0
Consumption of resource Hydrogel by each building in the colony:
0    0    0    0    0    0
Consumption of resource Nanomaterials by each building in the colony:
0    0    0    0    0    0
Please enter an option number:
1
There are no empty cells in the colony. Can not add a new building.
Please enter an option number:
2
Please enter the row and the column index of the cell that you want to remove:
0 6
Invalid row or column index, please enter a valid row and column index:
0 5
The building is removed, and the corresponding resources are added back to the stock.
Please enter an option number:
3
Colony:
D    D    D    D    D    -
Please enter an option number:
1
Please enter the type of the building that you want to construct:
u
Invalid building type, please enter a valid building type:
J
Please enter the number of cells that the building will occupy:
2
Invalid number of cells, please enter a valid number of cells:
0
Invalid number of cells, please enter a valid number of cells:
14
Invalid number of cells, please enter a valid number of cells:
13
Invalid number of cells, please enter a valid number of cells:
1
Please enter the row and the column index of the cell number 1:
0 4
```

```
The cell is not empty, please enter the row and the column index of another cell:
0 9
Invalid row or column index, please enter a valid row and column index:
0 5
The building is added to the colony.
Please enter an option number:
3
Colony:
D    D    D    D    D    J
Please enter an option number:
5
Please enter the type of the resource:
Solarpanels
The resource Solarpanels was not found in the resources stock.
Please enter an option number:
5
Please enter the type of the resource:
SolarPanels
Consumption of resource SolarPanels by each building in the colony:
0    0    0    0    0    800
Please enter an option number:
5
Please enter the type of the resource:
Self-ReplicatingBots
Consumption of resource Self-ReplicatingBots by each building in the colony:
0    0    0    0    0    5
Please enter an option number:
4
Consumption of resource Biofuel by each building in the colony:
0    0    0    0    0    2000
Consumption of resource Self-ReplicatingBots by each building in the colony:
0    0    0    0    0    5
Consumption of resource SolarPanels by each building in the colony:
0    0    0    0    0    800
Consumption of resource Hydrogel by each building in the colony:
0    0    0    0    0    300
Consumption of resource Nanomaterials by each building in the colony:
0    0    0    0    0    150
Please enter an option number:
6
Resource stock:
Biofuel 0
Self-ReplicatingBots 0
SolarPanels 0
```

```
Hydrogel 0
Nanomaterials 0
Please enter an option number:
1
There are no empty cells in the colony. Can not add a new building.
Please enter an option number:
2
Please enter the row and the column index of the cell that you want to remove:
1 0
Invalid row or column index, please enter a valid row and column index:
0 0
The building is removed, and the corresponding resources are added back to the stock.
Please enter an option number:
3
Colony:
-    D    D    D    D    J
Please enter an option number:
4
Consumption of resource Biofuel by each building in the colony:
0    0    0    0    0    2000
Consumption of resource Self-ReplicatingBots by each building in the colony:
0    0    0    0    0    5
Consumption of resource SolarPanels by each building in the colony:
0    0    0    0    0    800
Consumption of resource Hydrogel by each building in the colony:
0    0    0    0    0    300
Consumption of resource Nanomaterials by each building in the colony:
0    0    0    0    0    150
Please enter an option number:
6
Resource stock:
Biofuel 0
Self-ReplicatingBots 0
SolarPanels 0
Hydrogel 0
Nanomaterials 0
Please enter an option number:
1
Please enter the type of the building that you want to construct:
M
Please enter the number of cells that the building will occupy:
2
Invalid number of cells, please enter a valid number of cells:
1
```

```
Not enough Biofuel to build this building.
Please enter an option number:
6
Resource stock:
Biofuel 0
Self-ReplicatingBots 0
SolarPanels 0
Hydrogel 0
Nanomaterials 0
Please enter an option number:
3
Colony:
-    D    D    D    D    J
Please enter an option number:
7
Thank you for using the colony management system. The program will terminate.
Goodbye!
```

Sample run 3:

```
Welcome to the colony management system
Please enter file name for resources stock:
stock2.txt
Available resources loaded from stock2.txt
Resource stock:
Wood 100
Gas 7
Please enter file name for resource consumption per building type:
consumption2.txt
Resources consumption per building type loaded from consumption2.txt
Resources consumption per building type:
H 11 1
R 5 0
D 0 3
M 1 10
Please enter file name for colony:
colony2.txt
Colony loaded from colony2.txt
Colony:
H     H     H
D     R     H
Resources stock after loading the colony:
Resource stock:
Wood 51
```

```
Gas 0
Consumption of resource Wood by each building in the colony:
11    11    11
0     5     11
Consumption of resource Gas by each building in the colony:
1     1     1
3     0     1
Please enter an option number:
1. Construct a new building on the colony.
2. Destruct/Disassemble a building from the colony.
3. Print the colony.
4. Print the consumption of all resources by each building in the colony.
5. Print the consumption of a specific resource by each building in the colony.
6. Print the resources stock.
7. Exit the program.
1
There are no empty cells in the colony. Can not add a new building.
Please enter an option number:
1
There are no empty cells in the colony. Can not add a new building.
Please enter an option number:
2
Please enter the row and the column index of the cell that you want to remove:
1 2
The building is removed, and the corresponding resources are added back to the stock.
Please enter an option number:
3
Colony:
H    H    H
D    R    -
Please enter an option number:
4
Consumption of resource Wood by each building in the colony:
11    11    11
0     5     0
Consumption of resource Gas by each building in the colony:
1     1     1
3     0     0
Please enter an option number:
6
Resource stock:
Wood 62
Gas 1
Please enter an option number:
```

```
1
Please enter the type of the building that you want to construct:
m
Invalid building type, please enter a valid building type:
M
Please enter the number of cells that the building will occupy:
2
Invalid number of cells, please enter a valid number of cells:
1
Not enough Gas to build this building.
Please enter an option number:
3
Colony:
H    H    H
D    R    -
Please enter an option number:
4
Consumption of resource Wood by each building in the colony:
11   11   11
0    5    0
Consumption of resource Gas by each building in the colony:
1    1    1
3    0    0
Please enter an option number:
6
Resource stock:
Wood 62
Gas 1
Please enter an option number:
1
Please enter the type of the building that you want to construct:
D
Please enter the number of cells that the building will occupy:
1
Not enough Gas to build this building.
Please enter an option number:
1
Please enter the type of the building that you want to construct:
R
Please enter the number of cells that the building will occupy:
1
Please enter the row and the column index of the cell number 1:
0 0
The cell is not empty, please enter the row and the column index of another cell:
```

```
1 2
The building is added to the colony.
Please enter an option number:
3
Colony:
H    H    H
D    R    R
Please enter an option number:
4
Consumption of resource Wood by each building in the colony:
11   11   11
0    5    5
Consumption of resource Gas by each building in the colony:
1    1    1
3    0    0
Please enter an option number:
6
Resource stock:
Wood 57
Gas 1
Please enter an option number:
7
Thank you for using the colony management system. The program will terminate.
Goodbye!
```

Sample run 4:

```
Welcome to the colony management system
Please enter file name for resources stock:
stock4.txt
Available resources loaded from stock4.txt
Resource stock:
Wood 100
Gas 7
Please enter file name for resource consumption per building type:
consumption4.txt
Resources consumption per building type loaded from consumption4.txt
Resources consumption per building type:
H 11 1
R 5 0
D 0 4
M 1 10
Please enter file name for colony:
colony4.txt
```

```
Not enough Gas to build this building.
Not enough resources to build this colony.
Thank you for using the colony management system. The program will terminate.
Goodbye!
```

Sample Run 5:

```
Welcome to the colony management system
Please enter file name for resources stock:
stock3.txt
Available resources loaded from stock3.txt
Resource stock:
Biofuel 2000
Self-ReplicatingBots 5
SolarPanels 800
Hydrogel 300
Nanomaterials 150
Please enter file name for resource consumption per building type:
consumption3.txt
Resources consumption per building type loaded from consumption3.txt
Resources consumption per building type:
U 2500 21 45 19 0
E 50 6 16 29 34
J 2000 5 800 300 150
M 100 5 500 200 50
D 0 0 0 0 0
Please enter file name for colony:
colony3.txt
Colony loaded from colony3.txt
Colony:
-    -    -    -    -    D
Resources stock after loading the colony:
Resource stock:
Biofuel 2000
Self-ReplicatingBots 5
SolarPanels 800
Hydrogel 300
Nanomaterials 150
Consumption of resource Biofuel by each building in the colony:
0    0    0    0    0    0
Consumption of resource Self-ReplicatingBots by each building in the colony:
0    0    0    0    0    0
Consumption of resource SolarPanels by each building in the colony:
0    0    0    0    0    0
```

```
Consumption of resource Hydrogel by each building in the colony:
0    0    0    0    0    0
Consumption of resource Nanomaterials by each building in the colony:
0    0    0    0    0    0
Please enter an option number:
1. Construct a new building on the colony.
2. Destruct/Disassemble a building from the colony.
3. Print the colony.
4. Print the consumption of all resources by each building in the colony.
5. Print the consumption of a specific resource by each building in the colony.
6. Print the resources stock.
7. Exit the program.
2
Please enter the row and the column index of the cell that you want to remove:
0 7
Invalid row or column index, please enter a valid row and column index:
0 6
Invalid row or column index, please enter a valid row and column index:
0 5
The building is removed, and the corresponding resources are added back to the stock.
Please enter an option number:
3
Colony:
-    -    -    -    -    -
Please enter an option number:
2
There are no buildings in the colony. Can not remove a building.
Please enter an option number:
4
Consumption of resource Biofuel by each building in the colony:
0    0    0    0    0    0
Consumption of resource Self-ReplicatingBots by each building in the colony:
0    0    0    0    0    0
Consumption of resource SolarPanels by each building in the colony:
0    0    0    0    0    0
Consumption of resource Hydrogel by each building in the colony:
0    0    0    0    0    0
Consumption of resource Nanomaterials by each building in the colony:
0    0    0    0    0    0
Please enter an option number:
6
Resource stock:
Biofuel 2000
Self-ReplicatingBots 5
```

```
SolarPanels 800
Hydrogel 300
Nanomaterials 150
Please enter an option number:
7
Thank you for using the colony management system. The program will terminate.
Goodbye!
```

Some Important Rules

In order to get full credit, your program must be efficient, modular (with the use of functions), well commented and properly indented. Besides, you also have to use understandable identifier names. Presence of any redundant computation, bad indentation, meaningless identifiers or missing/irrelevant comments may decrease your grade in case that we detect them. When we grade your THEs, we pay attention to these issues. Moreover, **we may test your programs with very large test cases**. Hence, take into consideration the efficiency of your algorithms other than correctness.

Sample runs give a good estimate of how correct your implementation is, however, we will test your programs with different test cases and **your final grade may conflict with what you have seen on CodeRunner**. We will also **manually** check your code, indentations and so on, hence do not object to your grade based on the **CodeRunner** results, but rather, consider every detail on this documentation. **So please make sure that you have read this documentation carefully and covered all possible cases, even some other cases you may not have seen on CodeRunner or the sample runs**. The cases that you *do not need* to consider are also given throughout this documentation.
Submit via CodeRunner on SUCourse ONLY! Paper, e-mail or any other methods are not acceptable.

The internal clock of SUCourse might be a couple of minutes skewed, so make sure you do not leave the submission to the last minute. In the case of failing to submit your THE on time:
"No successful submission on SUCourse on time = A grade of zero (0) directly."

What and where to submit (PLEASE READ, IMPORTANT)

It'd be a good idea to write your name and last name in the program (as a comment line of course). Do not use any Turkish characters anywhere in your code (not even in comment parts). If your full name is "Duygu Karaoğlan Altop", and if you want to write it as comment; then you must type it as follows:

// Duygu Karaoglan Altop

You should copy the full content of the .cpp file and paste it into the specified “Answer” area in the relevant assignment submission page on SUCourse. ***Please note that the warnings may be considered as errors on CodeRunner, which means that you should have a compiling and warning-free program***.

Since the grading process will be automatic, you are expected to strictly follow these guidelines. If you do not follow these guidelines, your grade will be zero (0). Any tiny change in the output format will result in your grade being zero (0), so please test your programs yourself, and against the sample runs that are available at the relevant assignment submission page on SUCourse.

In the CodeRunner, there are some visible and invisible (hidden) test cases. You will know whether your code has successfully passed all the test cases or not before submitting your code. There is no re-submission. You don't have to complete your task in one time, you can continue from where you left last time but you should not press submit before finalizing it. Therefore, you should make sure that it’s your final solution version before you submit it. Also, we still do not suggest that you develop your solution on CodeRunner but rather on your IDE on your computer.

You may visit the office hours if you have any questions regarding submissions.

How to get help?

You may ask your questions to TAs or to the instructor. Information regarding the office hours of the TAs and the instructor are available at SUCourse.

Plagiarism

Plagiarism is checked by automated tools, and we are very capable of detecting such cases. Be careful with that. Exchange of abstract ideas are totally okay but once you start sharing the code with each other, it is very probable to get caught by plagiarism. So, do NOT send any part of your code to your friends by any means or you might be charged as well, although you have done your THE by yourself. THEs are to be done personally and you have to submit your own work. **Cooperation will NOT be counted as an excuse.**

In case of plagiarism, the rules on the Syllabus apply.

Good Luck!
Ahmed Salem, Duygu Karaoğlan Altop