CS204 – Fall 2021 - Sabancı University
Homework #1 – Minesweeper
Due: October 14, Thursday, 23:55

Description

Your first homework is about developing a simplified version of the Minesweeper game which is familiar to almost all of you. Minesweeper is a single-player puzzle video game. The objective of the game is to clear a rectangular board containing hidden "mines" or “bombs” without detonating any of them, with the help from clues about the number of mines in the neighboring cells. The game originates from the 1960s, and it has been written for many computing platforms in use today. It has many variations and offshoots. The aim of this assignments is to practice matrices (vectors of vectors).

Some versions of Minesweeper set up the board so that the solution does not require guessing. If a square (cell) containing a mine is revealed (opened), the player loses the game. If no mine is revealed, a digit is instead displayed in the square, indicating how many adjacent (neighboring) cells contain mines; if no mines are adjacent, the square becomes blank, and all adjacent squares will be recursively revealed. The player uses this information to deduce the contents of other squares and may either safely reveal each square or mark the square as containing a mine.

In the sections below we give a more detailed explanation on the program that you should submit, accompanied with the corresponding sample runs.

1) Initializing the mined field (board)

The rectangular mined field (board) **must be represented by a matrix** (vector of vectors). The user at the beginning of the program gives the dimensions of the matrix, namely he/she enters the number of rows *row* and the number of columns *col*. You can assume that the dimensions are given in a proper format (i.e. both *row* and *col* are entered as non-zero positive integer numbers) by the user, thus you don’t have to do any further checks when the matrix dimensions are entered.

After declaring the matrix with the given dimensions, the user enters the number of mines (bombs) - *nrMines*. The number of mines (*nrMines*) should be greater than zero and smaller than the total number of cells in the matrix, thus these inequalities should hold: $0 < nrMines < row \cdot col$. While you can assume that the user always enters integer data types for the number of mines, yet it is up to the program to check whether the inequalities holds. If any of them doesn’t hold, then the user is repeatedly asked to give the number of mines up until both of the inequalities are satisfied.

After getting the proper *nrMines* you should randomly distribute all of the *nrMines* mines throughout the matrix. You can do this by using the *randgen* library from CS201. Make sure that in each cell there is no more than one mine, thus a cell cannot have two or more mines. Furthermore, for each cell you should calculate the number of mines in all of the adjacent (neighboring) cells. While calculating the number of neighboring mines of a certain cell, if that cell also has a mine (bomb), it is not counted towards the number of neighboring mines.

You are allowed to use multiple matrices in case you need them or use a single matrix of a data structure that you will define. In case that you are going to use more than one matrix, it is up to you to decide on the purpose of each matrix. In case that you are going to use a single matrix of structures, it is up to you to define the content of the structure.

2) Program flow

While running the program, the user constantly (in each iteration) selects one of the 3 choices up until the game is finished. These choices are as follows:

1) **Displaying the surrounding (number of mines in the neighborhood) of a cell**: It **displays (prints) <u>only once</u> the number of mines of a certain cell** whose coordinates are given by the user. If the user gives a coordinate (row or column index) which is outside of the range, the program repeatedly asks the user to re-enter the coordinates until they are in the range. If the selected cell has a bomb, the game will still continue. Since the surrounding of a cell is displayed only at the first time, in the subsequent displays that cell will be covered up until it is permanently opened if the user chooses the second choice (see below). In the subsequent iterations the user can choose to display the surrounding of a cell that has already been displayed before or opened before (see choice 2).

2) **Open a cell**: It **permanently displays (opens) a cell entered by the user till the end of the game**. If the selected (opened) cell is a bomb, the game is over. If it is not, then it is an integer which shows the number of bombs in the neighborhood of the selected cell. If the user gives a coordinate (row or column index) which is outside of the range, the program repeatedly asks the user to re-enter the coordinates until they are in the range. If the user chooses to open a cell that has already been opened, the program re-opens that cell as if it is opened for the first time.

3) **Exit program**: Automatically exists the program.

While displaying (printing) the mined field (matrix), if in a certain cell there is a mine (bomb), it is denoted by letter **‘B’**. If it is not opened yet, it is denoted by character **‘X’**. When the surrounding of a cell is displayed temporary (using choice 1) or permanently (using choice 2 and if there is not a bomb) then it is an integer which shows the number of neighboring bombs. Don’t forget to convert those integers that show the number of the neighboring bombs into char when keeping them at the matrix.

3) Program exit

The program exits in one of the following three cases:

1) The user chooses the third choice during the runtime (as it is explained in the previous section). In this case a corresponding message will be printed to the console and the program will exit.

2) The user “steps” on a bomb when choosing the second option during the runtime. In this case a message telling the user that he lost due to stepping in a bomb will be printed to the console and the program will exit.

3) The user opens all of the cells (by using choice 2) that don't contain any bomb, thus only bombs remain un-opened. In this case a message telling the user that he won due to opening all of the non-bomb cells will be printed to the console and the program will exit.

4) Other remarks

You can use any library that you find useful. In that case, besides your main cpp file, you should include those libraries in your zipped file that you are going to submit to SUCourse+. You can use functions as needed (it is up to you to decide how many and what they do). In order to avoid any inconsistencies, for any inquiries/questions related to the workflow and the general concept of the assignment you should contact your TA Seyedpouya Seyedkazemi by his email (seyedpouya@sabanciuniv.edu) or you can ask your questions through the course's WhatsApp group, where either the instructor or Seyedpouya will answer your inquiries. Of course, for technical help while coding your assignment, you are free to ask any of the TAs/LAs or the instructor during the online office hours.

VERY IMPORTANT!

Your programs will be compiled, executed and evaluated automatically; therefore you should definitely follow the rules for prompts, inputs and outputs. See **Sample Runs** section for some examples.

- **Order of inputs and outputs** must be in the abovementioned format.

Following these rules is crucial for grading, otherwise our software will not be able to process your outputs and you will lose some grades in the best scenario.

IMPORTANT!

If your code does not compile, you will get **zero**. Please be careful about this and double check your code before submission.

Sample Runs

Some sample runs are given below, but these are not comprehensive, therefore you must consider different kind of cases to check your code performance and get full mark. Note that at the beginning of each sample, after the proper initialization of the matrix, its final initialized state is printed in bold letters colored with red. You don't need to do this while coding your assignment. It is just done for illustrative purposes, so as to make it easier to understand the flow of the program in the iterations that will follow.

You shouldn't focus a lot on stuff such as the number of new lines, empty spaces between letters, etc. Most important is the flow and the correct behavior of the program. Also, user inputs are displayed in bold in the sample runs below.

Sample 1:

```
Give the dimensions of the matrix: 4 4

How many bombs: 16

The number of bombs can not be greater than the whole number of cells minus one. Please
give the number of bombs again: 0

The number of bombs could not be less than one. Please give the number of bombs again: 4


X   X   X   X      // B   2   0   0

X   X   X   X      // B   3   1   1

X   X   X   X      // 2   3   B   1

X   X   X   X      // 1   B   2   1


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

6

Your input is wrong. Please select one of the options: 1, 2 or 3.

@

Your input is wrong. Please select one of the options: 1, 2 or 3.

1

Give the coordinates: 1 5

It is out of range. Please give a valid coordinates: 8 2

It is out of range. Please give a valid coordinates: 1 1
```

```
Displaying the surrounding of (1,1):

X  X  X   X

X  3  X   X

X  X  X   X

X  X  X   X

Around (1,1) you have 3 bomb(s)


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

1

Give the coordinates: 0 0


Displaying the surrounding of (0,0):

1  X  X   X

X  X  X   X

X  X  X   X

X  X  X   X

Around (0,0) you have 1 bomb(s)


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

1

Give the coordinates: 0 0

Displaying the surrounding of (0,0):

1  X  X   X

X  X  X   X

X  X  X   X

X  X  X   X
```

```
Around (0,0) you have 1 bomb(s)


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

2

Give the coordinates: 2 3


Opening cell (2,3):

X   X   X   X

X   X   X   X

X   X   X   1

X   X   X   X

Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

2

Give the coordinates: 3 3


Opening cell (3,3):

X   X   X   X

X   X   X   X

X   X   X   1

X   X   X   1


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

2
```

```
Give the coordinates: 2 2

Opening cell (2,2):

X   X   X   X

X   X   X   X

X   X   B   1

X   X   X   1


Unfortunately, you stepped on a mine

Arrangement of mines:

B   2   0   0

B   3   1   1

2   3   B   1

1   B   2   1

>>>>> Game Over! <<<<<
```

Sample 2:

```
Give the dimensions of the matrix: 4 4

How many bombs: 3


X  X  X   X      // 0  1  1   1

X  X  X   X      // 1  2  B   1

X  X  X   X      // B  3  2   1

X  X  X   X      // 2  B  1   0


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

3

Program exiting ...
```

Sample 3:

```
Give the dimensions of the matrix: 3 6

How many bombs: 6
```

```
X   X   X   X   X   X       // 2   B   2   1   2   1

X   X   X   X   X   X       // 2   B   3   2   B   B

X   X   X   X   X   X       // 1   1   2   B   4   B


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

2

Give the coordinates: 1 1


Opening cell (1,1):

X   X   X   X   X   X

X   B   X   X   X   X

X   X   X   X   X   X


Unfortunately, you stepped on a mine

Arrangement of mines:

2   B   2   1   2   2

2   B   3   2   B   B

1   1   2   B   4   B


>>>>> Game Over! <<<<<
```

Sample 4:

```
Give the dimensions of the matrix: 5 5

How many bombs: 7


X   X   X   X   X   // 1   B   2   3   B

X   X   X   X   X   // 2   4   B   4   B

X   X   X   X   X   // 1   B   B   4   2

X   X   X   X   X   // 1   2   3   B   1
```

```
X   X   X   X   X   // 0   0   1   1   1


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

1

Give the coordinates: 3 3


Displaying the surrounding of (3,3):

X   X   X   X   X

X   X   X   X   X

X   X   X   X   X

X   X   X   1   X

X   X   X   X   X


Around (3,3) you have 1 bomb(s)


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

2

Give the coordinates: 1 1


Opening cell (1,1):

X   X   X   X   X

X   4   X   X   X

X   X   X   X   X

X   X   X   X   X

X   X   X   X   X
```

```
Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

2

Give the coordinates: 1 1


Opening cell (1,1):

X   X   X   X   X

X   4   X   X   X

X   X   X   X   X

X   X   X   X   X

X   X   X   X   X


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

1

Give the coordinates: 2 1


Displaying the surrounding of (2,1):

X   X   X   X   X

X   4   X   X   X

X   2   X   X   X

X   X   X   X   X

X   X   X   X   X


Around (2,1) you have 2 bomb(s)
```

```
Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

2

Give the coordinates: 3 2


Opening cell (3,2):

X   X   X   X   X

X   4   X   X   X

X   X   X   X   X

X   X   3   X   X

X   X   X   X   X


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

2

Give the coordinates:  0 1


Opening cell (0,1):

X   B   X   X   X

X   4   X   X   X

X   X   X   X   X

X   X   3   X   X

X   X   X   X   X


Unfortunately, you stepped on a mine
```

```
Arrangement of mines:

1   B   2   3   B

2   4   B   4   B

1   B   B   4   2

1   2   3   B   1

0   0   1   1   1


>>>>> Game Over! <<<<<
```

Sample 5:

```
Give the dimensions of the matrix: 3 2

How many bombs: 4


X   X      //B     3

X   X      //B     B

X   X      //3     B


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

1

Give the coordinates: 2 1


Displaying the surrounding of (2,1):

X   X

X   X

X   2


Around (2,1) you have 2 bomb(s)
```

```
Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

2

Give the coordinates: 0 1


Opening cell (2,0):

X   3

X   X

X   X


Press:

1. If you want to find out the surrounding of a cell

2. If you want to open the cell

3. If you want to exit.

2

Give the coordinates: 2 0


Opening cell (2,0):

X   3

X   X

3   X

Congratulations! All the non-mined cells opened successfully

You won!


>>>>> Game Over! <<<<<
```

General Rules and Guidelines about Homeworks

The following rules and guidelines will be applicable to all homeworks, unless otherwise noted.

How to get help?

You may ask questions to TAs (Teaching Assistants) and LAs (Learning assistants) of CS204. Office hours of TAs are at SUCourse+. Lectures and/or recitations will be partially dedicated to clarify the issues related to homework, so it is to your benefit to attend the lectures/recitations or watch the corresponding video lecture.
The style of submitting the assignments is similar to CS201.

What and Where to Submit

Please see the detailed instructions below/in the next page. The submission steps will get natural/easy for later homeworks.

Grading and Objections

Careful about the semi-automatic grading: Your programs might be graded using a semi-automated system. Therefore you should follow the guidelines about input and output order; moreover you should also use same prompts as given in the Sample Runs. Otherwise semi-automated grading process will fail for your homework, and you may get a zero, or in the best scenario you will lose points.

Grading:

- ❑ Late penalty is 10% off of the full grade and only one late day is allowed.
- ❑ **Having a correct program is necessary, but not sufficient to get the full grade. Comments, indentation, meaningful and understandable identifier names, informative introduction and prompts, and especially proper use of required functions, unnecessarily long program (which is bad) and unnecessary code duplications (which is also bad) will also affect your grade.**
- ❑ Please submit your own work only (even if it is not working). It is really easy to find out "similar" programs!
- ❑ For detailed rules and course policy on plagiarism, please check out http://people.sabanciuniv.edu/levi/cs204/policy_plagiarism.html. and keep in mind that

Plagiarism will not be tolerated!

Grade announcements: Grades will be posted in SUCourse+, and you will get an Announcement at the same time. You will find the grading policy and/or test cases in that announcement.

Grade objections: It is your right to object to your grade if you think there is a problem, but before making an objection please try the steps below and if you still think there is a problem, contact the TA that graded your homework from the email address provided in the comment section of your announced homework grade or attend the specified objection hour in your grade announcement.

- Check the comment section in the homework tab to see the problem with your homework.
- Download the .zip file you submitted to SUCourse+ and try to compile it.
- Check the test cases in the announcement and try them with your code.
- Compare your results with the given results in the announcement.

What and where to submit (IMPORTANT)

Submissions guidelines are below. Students are expected to strictly follow these guidelines in order to have a smooth grading process. If you do not follow these guidelines, depending on the severity of the problem created during the grading process, 5 or more penalty points are to be deducted from the grade.

<u>Add your name to the program:</u> It is a good practice to write your name and last name somewhere in the beginning program (as a comment line of course).

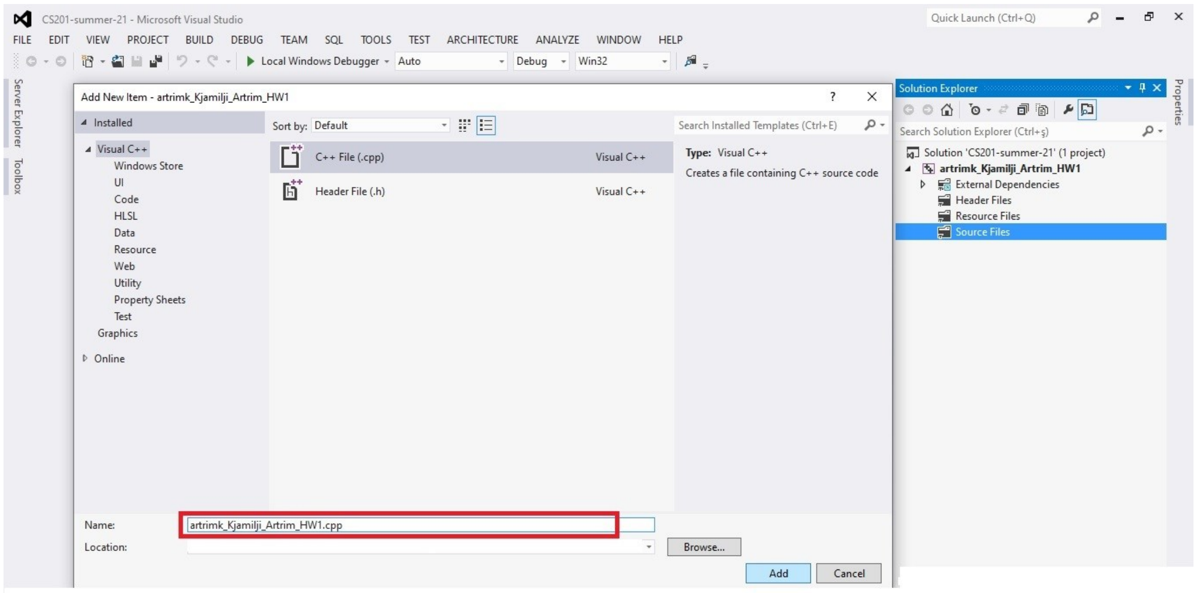
<u>Name your submission file:</u>

❑ <u>Use only English alphabet letters, digits or underscore in the file names. Do not use blank, Turkish characters or any other special symbols or characters.</u>

❑ Name your cpp file that contains your program as follows:

“**SUCourseUserName_YourLastname_YourName_HWnumber.cpp**”

❑ Your SUCourse+ user name is actually your SUNet user name which is used for checking sabanciuniv e-mails. Do NOT use any spaces, non-ASCII and Turkish characters in the file name. For example, if your SUCourse+ user name is artrimk, your name is Artrim, your last name is Kjamilji, and you are uploading the first homework (HW1) to the corresponding assignment in SUCourse +, then the file name must be: **artrimk_Kjamilji_Artrim_hw1.cpp**


CS201-summer-21 - Microsoft Visual Studio
FILE EDIT VIEW PROJECT BUILD DEBUG TEAM SQL TOOLS TEST ARCHITECTURE ANALYZE WINDOW HELP
Local Windows Debugger
Auto
Debug
Win32
Quick Launch (Ctrl+Q)
Add New Item - artrimk_Kjamilji_Artrim_HW1
Installed
Visual C++
Windows Store
UI
Code
HLSL
Data
Resource
Web
Utility
Property Sheets
Test
Graphics
Online
Sort by: Default
C++ File (.cpp) Visual C++
Header File (.h) Visual C++
Search Installed Templates (Ctrl+E)
Type: Visual C++
Creates a file containing C++ source code
Browse...
Add
Cancel
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'CS201-summer-21' (1 project)
artrimk_Kjamilji_Artrim_HW1
External Dependencies
Header Files
Resource Files
Source Files
Server Explorer
Toolbox
Properties

Name: artrimk_Kjamilji_Artrim_HW1.cpp

Location:

❑ Do not add any other character or phrase to the file name.

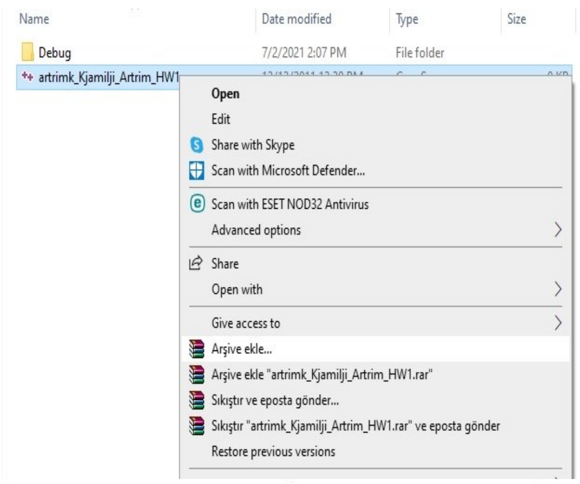
❑ Make sure that this file is the latest version of your homework program.

❑ Compress this cpp and, if necessary, other files such as libraries (both the .h and the corresponding .cpp files) that you might have used in your assignment using WINZIP or WINRAR programs. Please use "**<u>zip</u>**" compression. "rar" or another compression mechanism is NOT allowed. Our homework processing system works only with zip files. Therefore, make sure that the resulting compressed file has a zip extension.

- ❑ Check that your compressed file opens up correctly and it contains all of your **zipped** file(s). You will receive no credits if your compressed zip file does not expand or it does not contain the correct file(s).
- ❑ The naming convention of the zip file is the same as the cpp file (except the extension of the file of course). The name of the zip file should be as follows.

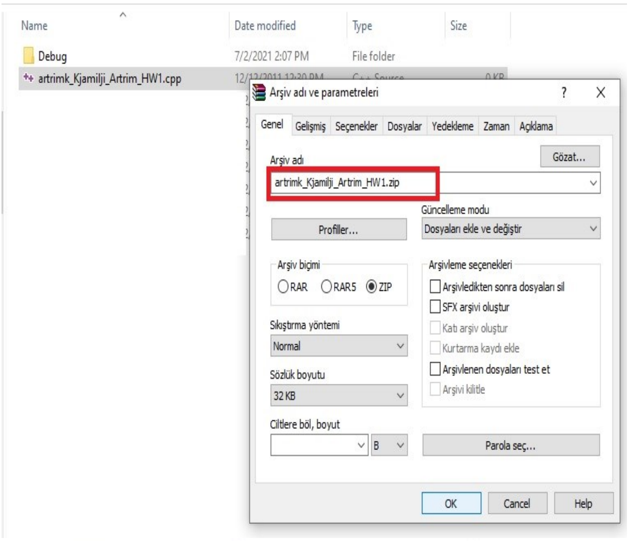
“**SUCourseUserName_YourLastname_YourName_HWnumber.zip**”

For example zubzipler_Zipleroglu_Zubeyir_hw1.zip is a valid name, but hw1_hoz_HasanOz.zip, HasanOzHoz.zip are NOT valid names.



Name	Date modified	Type	Size
Debug	7/2/2021 2:07 PM	File folder	
artrimk_Kjamilji_Artrim_HW1			

Open
Edit
Share with Skype
Scan with Microsoft Defender...
Scan with ESET NOD32 Antivirus
Advanced options
Share
Open with
Give access to
Arşive ekle...
Arşive ekle "artrimk_Kjamilji_Artrim_HW1.rar"
Sıkıştır ve eposta gönder...
Sıkıştır "artrimk_Kjamilji_Artrim_HW1.rar" ve eposta gönder
Restore previous versions



Name	Date modified	Type	Size
Debug	7/2/2021 2:07 PM	File folder	
artrimk_Kjamilji_Artrim_HW1.cpp			

Arşiv adı ve parametreleri

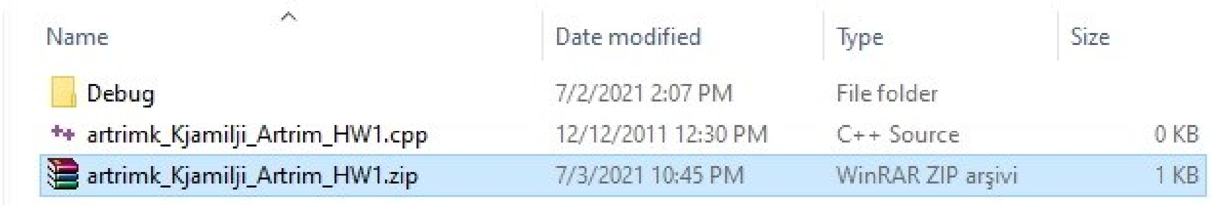
Arşiv adı: artrimk_Kjamilji_Artrim_HW1.zip

Güncelleme modu: Dosyaları ekle ve değiştir

Arşiv biçimi: ○ RAR ○ RAR5 [✓ ZIP]

Sıkıştırma yöntemi: Normal

Sözlük boyutu: 32 KB



Name	Date modified	Type	Size
Debug	7/2/2021 2:07 PM	File folder	
artrimk_Kjamilji_Artrim_HW1.cpp	12/12/2011 12:30 PM	C++ Source	0 KB
artrimk_Kjamilji_Artrim_HW1.zip	7/3/2021 10:45 PM	WinRAR ZIP arşivi	1 KB

<u>Submission:</u>

- ❑ Submit via SUCourse+ ONLY! You will receive no credits if you submit by other means (e-mail, paper, etc.). Click on "ASSIGNMENTS" at CS204 SUCourse+.

<u>Resubmission:</u>

- ❑ If you would like to resubmit your work, you should first remove the existing file(s). This step is very important. If you do not delete the old file(s), we will receive both files and the old one may be graded.

Successful submission is one of the requirements of the homework. If, for some reason, you cannot successfully submit your homework and we cannot grade it, your grade will be 0.

Good Luck!
Artrim Kjamilji and Seyedpouya Seyedkazemi