



Search ...

Computer Science Department

Introduction to Computer Science

RUTGERS School of Arts and Sciences

Arrays – 50 course points

The purpose of this assignment is to practice array manipulation.

Refer to our Programming [Assignments FAQ](#) for instructions on how to install VScode, how to use the command line and how to submit your assignments.

Programming

Write 2 programs and submit on [Autolab](#).

- We provide a zip file (find it under BusStop on Autolab) containing *BusStop.java* and *EgyptianPyramid.java*.
- For each problem UPDATE and SUBMIT the corresponding file.

Observe the following rules:

DO NOT add any import statements
DO NOT add the project statement
DO NOT change the class name
DO NOT change the headers of ANY of the given methods
DO NOT add any new class fields
DO NOT as a result use static variables
DO NOT use `System.exit()`

ONLY print the result as specified by each problem.

Observe the examples' output and display only what the problem is asking for.

DO NOT print other messages; follow the examples for each problem.

1. ***RU BusStop (25 points)***. Serena, a freshmen at Rutgers just finished her *Intro to Logic* class on College Ave and needs to return to Busch for her *Intro to Computer Science* class. She is currently standing at the Yard Bus Stop,

waiting patiently for her desired bus to arrive. Your job is to help Serena find in which order that bus will arrive!

Because we only covered **char** at this point in the semester all busses routes' names are 1 character in length. We cannot have B-He or REXB routes.

See Rutgers bus routes below, where some are fictitious.

- A Route
- B Route
- C Route
- E Route
- F Route
- H Route
- L Route
- R Route
- X Route

Write the program *BusStop.java* that reads **n** char arguments from the command line.

- The first **n-1** characters refer to the order in which the buses will arrive at the bus stop.
- The last char refers to the bus Serena is waiting for.

The program is expected to display the order in which the bus Serena is waiting for will arrive. If the bus Serena is waiting for does *not stop at the bus stop* then the program **displays -1**.

To read the first character from the command line use:

- `char c = args[0].charAt(0);`

To read the nth character from the command line use:

- `char c = args[n-1].charAt(0);`

You will have to read an undetermined number of characters, so, you will need a loop to read all characters from the command line.

Hint:

1. Read the first n-1 values and store them into a **char 1D array**.
2. Then, read the last char in the command line, and
3. Search for that value in the char array.

Example 1:

```
java BusStop C L H H
3
```

The first bus to arrive is C, followed by L, then H. The program displays 3 because H is the third bus to arrive.

Example 2:

```
java BusStop B R X B
1
```

The first bus to arrive is B, followed by R, then X. The program displays 1 because B is the first bus to arrive.

Example 3:

```
java BusStop B C E A B H L
-1
```

The L bus does not stop at the bus stop, the program displays -1.

Assume: all the inputs are **char** values.

Problem by Manya Sood

2. Egyptian Pyramid (25 points). Let's build a pyramid with the help of a 2-dimensional (2D) array.

Long ago, some advanced civilizations such as the Egyptians modeled their architecture to be of pyramid form: 3-Dimensional polyhedra with some polygon base and triangle-shaped flat sides meeting to a point at the top. For the sake of this assignment, we will be considering 2-D pyramids, which take the form of triangles. You will be tasked with understanding and implementing 2D array(s) to create a 2-Dimensional pyramid of your own.

Overview

Oh no! During Professor Centeno's 2D Arrays lecture, a mysterious portal opened up right into the projected picture of the Egyptian pyramids, sucking all the CS111 students into another time dimension. It looks like the students are trapped in 2580 B.C. After some conversing (let's assume due to advanced language translation technology that students happened to bring), they discover that the Egyptians need assistance in building their pyramids. Since Professor Centeno did not time travel with the CS111 students, it's now up to them to help these Egyptians out.

The students need to code a program that reads two command line arguments from the terminal representing:

1. the overall size of their figure
2. the number of bricks provided-to build a pyramid.

Programming

The students are expected to use a 2D array to construct the pyramid. The pyramid is constructed using the characters 'X' (bricks) and '=' (no brick).

- Bricks, denoted by the symbol 'X', are to be placed beginning at the bottom left and continue until either 1) there are no more bricks or 2) the pyramid is complete.
- The surrounding area is to be filled with the symbol '='.

Once the 2D array (pyramid) is constructed print:

- the 2D array starting at row and column zero [0,0].
- the number of bricks remaining, in the format listed below.

Note: Keep in mind that the output is expected be **PRINTED** into the terminal.

Programming

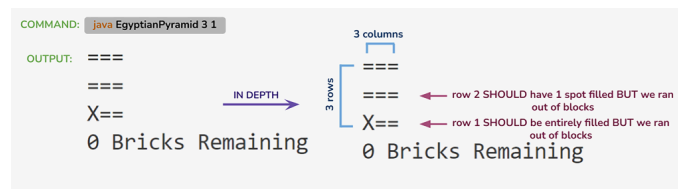
Example 1: Executing the command

```
javac EgyptianPyramid.java
java EgyptianPyramid 3 1
```

produces a 3x3 2D grid with 1 initial brick to construct the pyramid.



The output looks like this (Note the placement of the brick is on the left side of the last row):



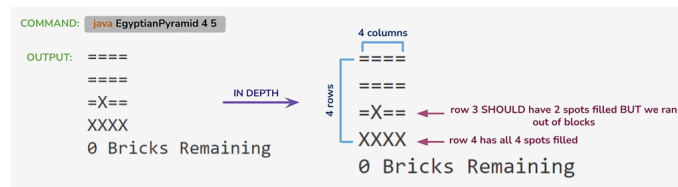
Example 2: Executing the command

```
javac EgyptianPyramid.java
java EgyptianPyramid 4 5
```

produces a 4×4 2D grid with 5 initial bricks to construct the pyramid.



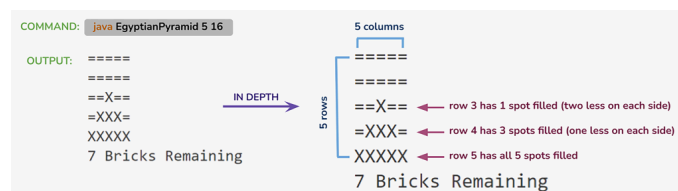
The output looks like this:



Example 3: Executing the command

```
javac EgyptianPyramid.java
java EgyptianPyramid 5 16
```

produces a 5×5 2D grid with 16 initial bricks to construct the pyramid. The output looks like this:



Problem by Kushi Sharma and Ayla Muminovic

Before submission

Collaboration policy. Read our collaboration policy [here](#).

Submitting the assignment. Submit *BusStop.java* and *EgyptianPyramid.java* separately via the web submission system called Autolab. To do this, click the *Assignments* link from the course website; click the *Submit* link for that assignment.

Getting help

If anything is unclear, don't hesitate to drop by office hours or post a question on Piazza.

- Find instructors office hours [here](#)
- Find tutors office hours in Canvas -> Tutoring, [RU CATS](#)

- Find head TAs office hours [here](#)
- **POST** on Piazza in Canvas -> Piazza
- In addition to office hours we have the [CSL](#) (Coding and Social Lounge), a community space staffed with lab assistants which are undergraduate students further along the CS major to answer questions.

Connect with Rutgers

Rutgers Home

Rutgers Today

myRutgers

Academic Calendar

Calendar of Events

SAS Events

Explore SAS

Departments & Degree-Granting Programs

Other Instructional Programs

Majors & Minors

Research Programs, Centers, & Institutes

International Programs

Division of Life Sciences

Explore CS

We are Hiring!

Research

News

Events

Resources

Search CS

Home

[Back to Top](#)

Copyright 2020, Rutgers, The State University of New Jersey. All rights reserved.
Rutgers is an equal access/equal opportunity institution. Individuals with disabilities are encouraged to direct suggestions, comments, or complaints concerning any accessibility issues

with Rutgers web sites to: accessibility@rutgers.edu or complete the **Report Accessibility Barrier or Provide Feedback Form**.