COP3530 Project 2 – Stacks and Priority Queues

Due Date: Friday, 10/8/2021 11:59 PM

Turn in:

Submit the zipped Eclipse program including at least Project2.java, Stack.java, Priority.java, State.java and States2.csv. The zip file should be named <your last name>_Project2.zip (for example, Liu_Project2.zip). The States2.csv file contains information about all 50 States in the USA. For each State, in the CSV are its name, capitol, region, US House seats and population (according to the *2020 Census* found in *Wikipedia* at https://en.wikipedia.org/wiki/List_of_states_and_territories_of_the_United_States_by_population), numbers of COVID-19 total cases and deaths (as of Sept 2, 2021, according to *Centers for Disease Control and Prevention* at https://covid.cdc.gov/covid-data-tracker/), median household income (according to the *2020 World Population Review* at *https://worldpopulationreview.com/state-rankings/median-household-income-by-state*), and violent crime rates (according to the *2019 FBI report* at https://ucr.fbi.gov/crime-in-the-u.s/2019/crime-in-the-u.s.-2019, did not find 2020 FBI data for this).

The program should be well documented in the format of doc comments in Java. Detailed formats are found at http://www.oracle.com/technetwork/articles/java/index-137868.html.

Requirements:

1. Feel free to reuse your **State** class from Project 1.
2. Create a class named **Stack** that will implement a stack of State objects using an array. Support the following methods.
   a. Constructor that creates the stack array based on a size provided.
   b. A **push** method to push a State on the stack.
   c. A **pop** method to pop a State off the stack and return it.
   d. A **printStack** method to print the stack from top down to bottom of it. (This method should not modify content of stack.)
   e. An **isEmpty** method that returns true if the stack is empty, false otherwise.
   f. An **isFull** method that returns true if the stack is full, false otherwise.
3. Create a class named **PriorityQ** that implements a priority queue of State objects using a **sorted array**, based on ***COVID-19 Death Rate (DR)***, the ***lower*** the DR, the ***higher*** the priority. The definition of DR is the same as in Project 1. Support the following methods:
   a. Constructor that creates the priority queue array based on a size provided.
   b. An **insert** method to insert a State into the queue. **(This should be an O(N) method.)**
   c. A **remove** method to remove a State from the queue and return it. **(This should be an O(1) method.)**
   d. A **printQueue** method to print the priority queue from beginning to end of it. (This method should not modify content of priority queue.)
   e. An **isEmpty** method that returns true if the priority queue is empty, false otherwise.
   f. An **isFull** method that returns true if the priority queue is full, false otherwise.
4. Create a class named **Project2** that will:

a. Prompt user to enter the name of the CSV file, e.g., States2.csv, as input to the system.
b. Create an empty stack and an empty priority queue.
c. Read the csv file to insert State objects, in the order shown in the input file, into the stack and into the priority queue.
d. Offer repetitively the user the following menu:
    1) Print stack
    2) Pop a state object from stack
    3) Push a state object onto stack
    4) Print priority queue
    5) Remove a state object from priority queue
    6) Insert a state object into priority queue
    7) Exit

5. When user selects options 3 or 6, the program should ask the user to enter the nine data members listed in the State class.

The queues and the stack should be printed in the following format:

```
Stack or  Priority Queue Contents:

Name          MHI          VCR            CFR             Case Rate     Death Rate
----------------------------------------------------------------------------------
Alabama       50536        510.8          0.014340        8455.38       121.25
```

Provide comments in this form for the **State, Stack,** and **PriorityQ** classes:

Comments for the class:

```
/**
 * Detailed description of the class.
 *
 * @author <your name>
 * @version <date you last changed the class>
 */
```

Public method comments:

```
/**
 * Description of the purpose of the method, the meaning of the
 * input parameters (if any) and the meaning of the return values
 * (if any).
 *
 * @param  parameter  description of the parameter (one for each)
 * @return description of the return value
 */
```

Provide comments in this form for the Project2 class.

```
/**
 * COP 3530: Project 2 – Stacks and PriorityQ Queues
 * <p>
 * Description of the class using as many lines as needed
 * with <p> between paragraphs. Including descriptions of the
 * input required and output generated.
 *
 * @author <your name>
 * @version <the date you last modified the program>
 */
public class Project2
{
```

Example:

```
COP3530 Project 2
Instructor: Xudong Liu

Stacks and Priority Queues
Enter the file name: States2.csv


Stack created of 50 states.
Priority queue created of 50 states.

1) Print stack
2) Pop a state object from stack
3) Push a state object onto stack
4) Print priority queue
5) Remove a state object from priority queue
6) Insert a state object into priority queue
7) Exit
Enter your choice: 1

Name          MHI          VCR            CFR             Case Rate     Death Rate
----------------------------------------------------------------------------------
Ohio          56602        293.2          0.016992        10407.28      176.84
……

1) Print stack
2) Pop a state object from stack
3) Push a state object onto stack
4) Print priority queue
5) Remove a state object from priority queue
6) Insert a state object into priority queue
7) Exit
Enter your choice: 2

One state is popped from stack.


1) Print stack
2) Pop a state object from stack
3) Push a state object onto stack
4) Print priority queue
5) Remove a state object from priority queue
6) Insert a state object into priority queue
7) Exit
Enter your choice: 1

Name          MHI          VCR            CFR             Case Rate     Death Rate
----------------------------------------------------------------------------------
Delaware      68287        422.6          0.015701        12140.03      190.62
……
```

```
1) Print stack
2) Pop a state object from stack
3) Push a state object onto stack
4) Print priority queue
5) Remove a state object from priority queue
6) Insert a state object into priority queue
7) Exit
Enter your choice: 3

Enter name: New State
Enter capitol: Big City
Enter region: south
Enter US House Seats: 13
……
Enter VCR: 365.12

One state is pushed onto stack.

1) Print stack
2) Pop a state object from stack
3) Push a state object onto stack
4) Print priority queue
5) Remove a state object from priority queue
6) Insert a state object into priority queue
7) Exit
Enter your choice: 1

Name          MHI          VCR            CFR             Case Rate     Death Rate
----------------------------------------------------------------------------------
New State     ……
……

1) Print stack
2) Pop a state object from stack
3) Push a state object onto stack
4) Print priority queue
5) Remove a state object from priority queue
6) Insert a state object into priority queue
7) Exit
Enter your choice: 4

Name          MHI          VCR            CFR             Case Rate     Death Rate
----------------------------------------------------------------------------------
Hawaii        81275        285.5          0.009676        4246.63       41.09
……

1) Print stack
2) Pop a state object from stack
3) Push a state object onto stack
4) Print priority queue
5) Remove a state object from priority queue
6) Insert a state object into priority queue
7) Exit
Enter your choice: 6

Enter name: Another State
Enter capitol: Snow City
Enter region: North
Enter US House Seats: 23
……
Enter VCR: 205.48
```

```
One state is inserted onto priority queue.

1) Print stack
2) Pop a state object from stack
3) Push a state object onto stack
4) Print priority queue
5) Remove a state object from priority queue
6) Insert a state object into priority queue
7) Exit
Enter your choice: 5

One state is removed from priority queue.

1) Print stack
2) Pop a state object from stack
3) Push a state object onto stack
4) Print priority queue
5) Remove a state object from priority queue
6) Insert a state object into priority queue
7) Exit
Enter your choice: 4

Name          MHI          VCR            CFR             Case Rate     Death Rate
----------------------------------------------------------------------------------
(Here you should see that the state with highest priority is gone.)

1) Print stack
2) Pop a state object from stack
3) Push a state object onto stack
4) Print priority queue
5) Remove a state object from priority queue
6) Insert a state object into priority queue
7) Exit
Enter your choice: 8
Invalid choice.  Enter your choice: ABC
Invalid choice.  Enter your choice: 7

Have a good day!
```