COP3530 Project 3 – Linked lists

Due Date: Friday, 10/22/2021 11:59 PM

Turn in:

Submit the zipped Eclipse project including at least Project3.java, State.java and States3.csv. The zip file should be named **<your last name>_Project3.zip** (for example, Liu_Project3.zip). The States3.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. Reuse your **State** class from Project 1 and 2 (make changes if necessary)
2. Create a class named **Stack** that will implement a stack of state objects using a **double-ended singly linked list**. Support the following methods.
    a. Constructor that creates the stack.
    b. A **push** method to push a state on the stack. **(This must be an O(1) method.)**
    c. A **pop** method to pop a state off the stack and return it. **(This must be an O(1) method.)**
    d. A ***recursive*** **printStack** method to print the stack from top of the stack to bottom. (This shan't change the stack.)
    e. An **isEmpty** method that returns true if the stack is empty, false otherwise
3. Create a class named **PriorityQ** that implements a **priority queue** of state objects using a **sorted double-ended doubly linked list**. The priority again is based on ***COVID-19 Death Rate (DR)***, the ***lower*** the DR, the ***higher*** the priority. Support the following methods:
    a. Constructor that creates the priority queue.
    b. An **insert** method to insert a State into the priority queue. **(This should be an O(N) method.)**
    c. A **remove** method to remove a State from the priority queue and return it. **(This should be an O(1) method.)**
    d. An **intervalDelete** method to delete from the priority queue state objects with COVID-19 DR that belong to the input interval. So this method takes two input parameters representing this interval, e.g., intervalDelete(80, 130) represent deleting states with DRs in the interval from 80 to 130, both inclusive. Return true if anything was found and deleted, and false otherwise.

   e. A ***recursive*** **printPriorityQ** method to print the priority queue from highest priority to lowest. (This shan't change the priority queue.)
   f. An **isEmpty** method that returns true if the priority queue is empty, false otherwise
4. We define five groups of states based on COVID-19 Death Rate: EXCELLENT states as those with DR <70, VGOOD no less than 70 but <100, GOOD no less than 100 but <150, FAIR no less than 150 but <250, POOR no less than 250.
5. Create a class named **Project3** that will:
   a. Read a file (csv) of states and create a single stack of state objects containing states from **FAIR**, **GOOD**, and **VGOOD** groups (only one stack is created; discard any state not in those groups; do not modify the input file.).
   b. Print the stack starting with the top of the stack.
   c. Create a priority queue. Pop items from the stack, one at a time, and insert them to the priority queue.
   d. Print the priority queue from highest priority to lowest.
   e. Repeatedly, prompt the user a short menu:
      1) Enter a DR interval for deletions.
      2) Print the priority queue.
      3) Exit program

      (Invalid user choices should be handled.)
   f. When option 1) is entered, validity of such interval should be checked too. If valid, say, the user gave interval [x, y], the intervalDelete method is called to delete states with **COVID-19 DR between x and y** in the priority queue.

The priority queues and the stacks should be printed in the following format:

```
Stack Contents:  or  Priority Queue Contents:

Name          MHI          VCR            CFR             Case Rate     Death Rate
-------------------------------------------------------------------------------
Alabama       50536        510.8          0.017373        14126.05      245.41
```

Provide comments in this form for the **State, Stack,** and Priority**Q** 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 Project3 class.

```
/**
 * COP 3530: Project 3 – Linked lists
 * <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 Project3
{
```

Example Output:

```
COP3530 Project 3

Enter the file name: States3.csv


Stack Contents:

Name           MHI          VCR            CFR             Case Rate     Death Rate
--------------------------------------------------------------------------------
. . .

Priority Queue Contents:

Name           MHI          VCR            CFR             Case Rate     Death Rate
--------------------------------------------------------------------------------
. . .

1. Enter a DR interval for deletions
2. Print priority queue
3. Exit
Enter your choice: 1
Enter DR interval like [x,y]: [X,10]
Invalid interval, enter numbers: [30,10]
Invalid interval, first number must be no bigger than the second: [80,130]

States of priority queue with DRs in [80,130] are deleted


1. Enter a DR interval for deletions
2. Print priority queue
3. Exit
Enter your choice: 2

Name           MHI          VCR            CFR             Case Rate     Death Rate
--------------------------------------------------------------------------------
. . .

1. Enter a DR interval for deletions
2. Print priority queue
3. Exit
Enter your choice: A
Invalid choice, enter 1-3: 8
Invalid choice, enter 1-3: 3

Have a good day!
```