Midterm 2

CS 1324, Spring 2018

Name (printed legibly):

Student number:

Integrity Pledge: On my honor, I affirm that I have neither given nor received inappropriate aid in the completion of this exercise.

Signature:

Do not write anything on the back of any page.

If you need additional space, use the blank page at the end of the examination. If you pull your examination apart, put ***all of your examination pages in order*** when you turn them in and ***turn in all pages including blank pages***.

Answer all programming questions in Java.

Pay careful attention to what is requested: a code fragment (a few lines of code), a method, or a program. Students who write a whole program when a code fragment is requested will waste so much time that they may not complete the examination.

Unless otherwise indicated, each part of a problem is worth the same number of points. Show your work to receive partial credit.

When a variable has a comment like "value assigned elsewhere," it means that the value of the variable has already been created somewhere out of sight (like in Turingscraft). You do not need to prompt the user or get values from the user in this situation.

Since you do not have the whole API available during the examination, it is acceptable to guess at method names, parameters and return values. If your guesses are reasonable—even if not perfect—you will receive credit. For example, if you forget that the String class has a length() method and call it size(), that is fine. If, however, you make up new methods that are not reasonable for the class or that magically solve problems the class cannot solve, you will not get credit.

You do not need to import packages or throw exceptions in any methods.

1. (12 points; 4 points each) Find the logical value (true or false) of each of the statements below. If the code is not legal, say so. Assume the variables are declared and initialized as follows:

double candy = 3.9 ;
int bunny = 14;
int[] eggs = {5, 1, 4, 2, 7, 5};

    a)   candy >= 0.0 || candy < 5.0

    b) 12 <= bunny < 15

    c) eggs[0] != eggs[5] && eggs[1]==eggs[4] || eggs[2] == eggs[3] || eggs[0] == eggs[5]

2. (6 points) Write a single if/else statement that uses a logical operator that will perform the operation described below. Use the variables declared below.

I like to buy paper towels at Fred's Club, but I don't want to have so many paper towels that I can't store them in my cabinet. My cabinet has enough space for 12 rolls of paper towels, but I already have some rolls stored there. I also only want to buy the towels if they cost less than $0.75 per roll. Write a code fragment that prints out whether I should buy paper towels or not.
int rollsInCabinet;   // The number of rolls stored in my cabinet. Value given elsewhere.
double pricePerPackage; // The price of a package, which contains multiple rolls. Value given elsewhere.
int rollsPerPackage;   // The number of rolls in one package. Value given elsewhere.

3. (15 points; 10 points for index, count and temp, 5 points for the contents of the rain array) Trace the loop below in the tables. Remember to show initial and final values.

```java
double[] rain = {4.2, 3.1, 0.4, 0.0, 0.0, 1.4, 3.1};


// READ CAREFULLY—not the usual for loop
for(int index= 0; index<rain.length; index = index + 2)
{
      if (rain[index] < 2.0)
      {
            double temp = rain[index];
            rain[index] = rain[0];
            rain[0] = temp;
      }
}
```

index	temp	rain[-1]	rain[0]	rain[1]	rain[2]	rain[3]	rain[4]	rain[5]	rain[6]	rain[7]

4. (5 points) The paragraph below describes a static method.

The method will take a given positive integer and create an array that is initialized with integers (starting at 1) and increasing by one with each index.

For example, if the given positive integer is 5, an array containing {1, 2, 3, 4, 5} would be returned.

Write the signature of the method. The method signature contains a return type, the method name and any necessary parameter(s).

5. (5 points) Suppose there is a method called nCopies. This method is given two integer parameters (size and init) and an array reference and changes the first size values in the array to init. For example, if the array contained {1, 2, 3, 4, 5}, size is 3, and init is 7, the method would return an array that contains {7, 7, 7, 4, 5} since the first 3 values in the array would be changed to 7s. The signature of the method is below.

void initializeArray(int size, int init, int[] data)

Write a few lines of code that call this method as shown in the example above.

6. (5 points) What will the value of index be after the code below is run?

```
int[] array = {2, 4, 5, 1, 8, 9, 7};
Arrays.sort(array);
int index = Arrays.binarySearch(array, 3);
```

7. (12 points) Answer the questions in the box below about program execution. You may use a memory diagram to trace the code if you wish to, or may use your knowledge of passing by value and/or sharing.

```java
public class SecondMidterm {
    public static void main(String[] args) {
        int[] start = {9, 1, 7, 2};
        int factor = 2;
        mystery(start, factor);
    }

    public static void mystery(int[] data, int mult) {
        mult = mult + 1;

        for (int index=0; index < data.length; ++index) {
            data[index] = mult + index;
        }

        data = new int[mult];
        for (int index = 0; index < data.length; ++index){
            data[index] = mult;
        }
    }
}
```

At the end of the program,
The contents of array start (not the reference) are (8 points):
factor contains (4 points):

main stack frame

Identifier	Address	Contents
start	100	
factor	101	
	102	
	103	

mystery stack frame

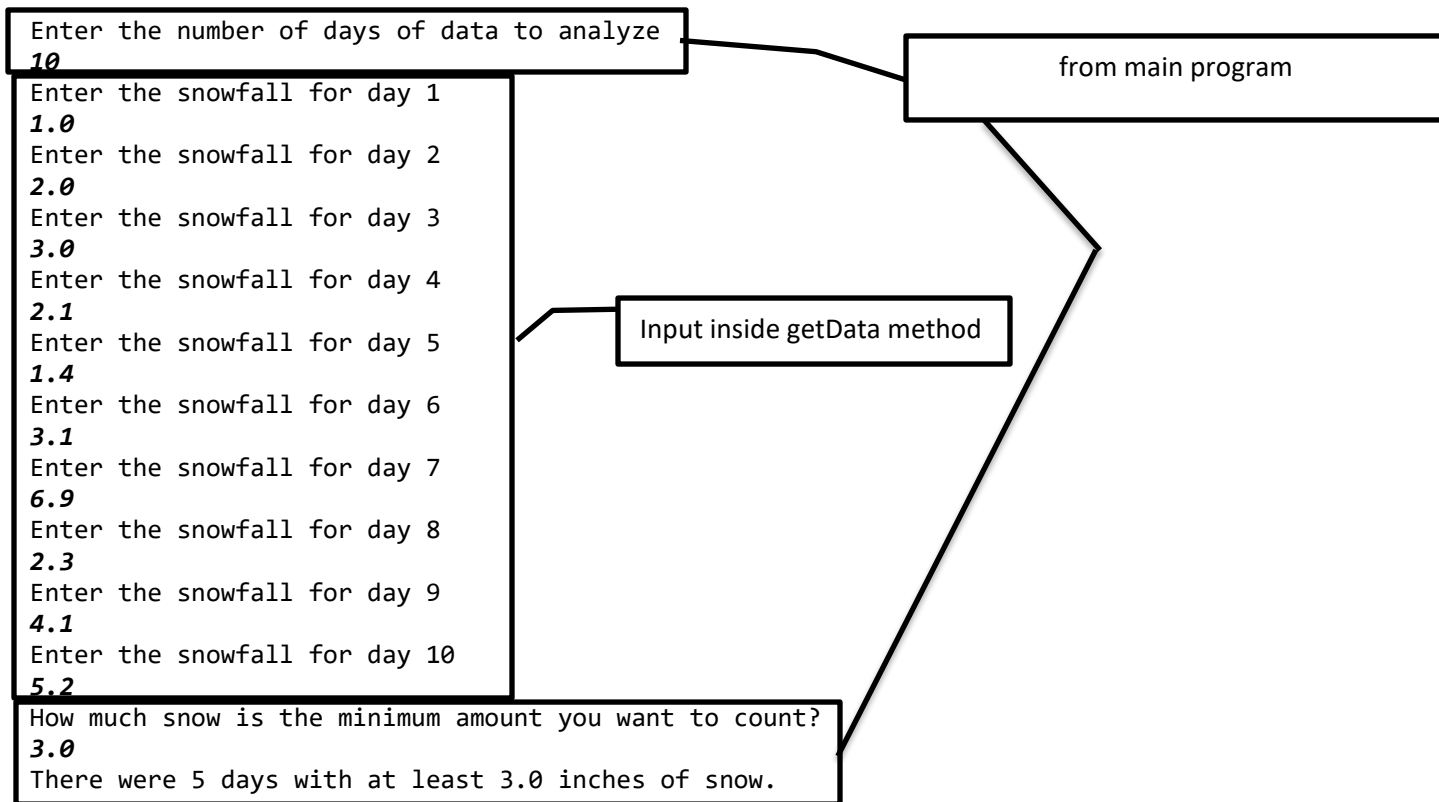| Identifier	Address	Contents
data	200	
mult	201	
	202	
	203	
	204	

heap

Identifier	Address	Contents
	1000	
	1001	
	1002	
	1003	
	1004	
	1005	
	1006	
	1007	
	1008	
	1009	
	1010	
	1011	
	1012	

Do not write anything on the back of any page

8. (40 points; 13 points for a), 13 points for b), 14 points for c)) Write three methods that together form a program to solve the problem below.

The east coast of the United States has recently experienced three strong storms (called Nor'easters) right in a row. This has piqued your interest in analyzing weather data.

You will write a program that allows the user to enter the amount of snow that fell on a given number of days. Your program should then allow the user to enter an amount of snow and have it count how many days had more snow than the entered amount.

A sample run of the program is below (the data in bold italics was entered by the user)

```
Enter the number of days of data to analyze
10
Enter the snowfall for day 1
1.0
Enter the snowfall for day 2
2.0
Enter the snowfall for day 3
3.0
Enter the snowfall for day 4
2.1
Enter the snowfall for day 5
1.4
Enter the snowfall for day 6
3.1
Enter the snowfall for day 7
6.9
Enter the snowfall for day 8
2.3
Enter the snowfall for day 9
4.1
Enter the snowfall for day 10
5.2
How much snow is the minimum amount you want to count?
3.0
There were 5 days with at least 3.0 inches of snow.
```

from main program

Input inside getData method

The program will have three methods:

public static double[] getData(Scanner keyboard, int days)

The data in the middle box above is entered inside this method.

public static int countMinimumSnowfall(double[] snow, double min)

This method has no input or output statements.

and a **main program**.

The number of days and minimum amount of snowfall are entered in the main program.

Do not write anything on the back of any page

a) Write the getData() method. This method allows the user to enter snowfall data for each day, storing the result in an array with days elements. The array reference should be returned.

public static double[] getData(Scanner keyboard, int days)

b) Write the countMinimumSnowfall method below. The method should take an array that contains daily snowfall and a minimum value. The method should return the number of days of snowfall that were greater than that minimum value.

public static int countMinimumSnowfall(double[] snow, double min)

c) **Write the main program.** The main program should call the methods you wrote in a) and b). Their signatures are below:

public static int countMinimumSnowfall(double[] snow, double min)

public static double[] getData(Scanner keyboard, int days)