

COP3503

Project 1 – Number Statistics

Overview:

Objective:

Show understanding of basic programming structures along with how to use methods and built in Java classes.

Description:

The number statistics program will prompt the user to input a list of numbers separated by spaces. Then the program will display a list of options for the user to pick from.

Options:

- Display List Statistics
- Display List Ordered
- Count Number of Even/Odd
- Count Number of Primes
- Enter New Number List
- Quit Program

The display statistics option must print the list's minimum, maximum, count, range, median, mean, mode, population variance, and population standard deviation (see sample output). The display list ordered option needs to print the list but ordered from least to greatest. This can be achieved by using the Arrays class do not create you own sorting algorithm. The count number of even/odds options needs to print out the number of even and the number of odd numbers in the list. The count number of primes option must print out how many numbers are prime in the list. The enter new number list option must allow the user to enter a new list of numbers separated by spaces. The last option allows the user to quit the program and exit meaning the program needs to continue to run until this option is selected.

Note: Each option should have at least 1 method associated to it. Some of those methods like the one for display list statistics should have supporting methods.

Requirements:

Documentation Requirements: 30 points

- 5 points: Write pseudo code that outlines the process of the “Display List Statistics” option
- 10 points: Create a flowchart that outlines the Project 1 Number Statistics program
 - This flowchart needs to accurately reflect your implementation of Project 1
- 15 points: Follow the guidelines given in the Programming and Commenting guides
 - These guides can be found on the Project 1 canvas page

Minimum Implementation Requirements: - 100 points

- The program must display the following list of options for the user to pick from
 - Display List Statistics
 - Display List Ordered
 - Count Number of Even/Odd
 - Count Number of Primes
 - Enter New Number List
 - Quit Program
- The program must continue to run until the user selects the “Quit Program” option
- The program must read in a list of numbers separated by spaces
- Do not use global variables or class member variables

Failure to meet the requirements list above will result in a zero for the project.

Implementation Requirements: 60 points

Display List Statistics: 15 points

The following values need to be calculated and printed for this option. (See sample output)

- Minimum
- Maximum
- Count
- Range
- Median
- Mean
- Mode
- Population variance
- Population standard deviation

Most of these values can be found by using the prebuilt methods in the Math class. If you can’t get the value in a single line, then create a method to calculate that value. For example, finding the median. Also, notice that we need to calculate the population variance and standard deviation not the sample which is a different formula. When printing the variance and standard deviation values only print 2 decimal places.

Display List Ordered: 5 points

This can be achieved easily by using 2 methods available in the Arrays class `sort()` and `toString()`.

Count Number of Even/Odd: 10 points

For this option all you need to do is count the number of even and odds then print it out to the user. The modulo operation will come in handy for this part.

Count Number of Primers: 15 points

For this option you will need to look up an algorithm online for determining if a number is prime or not. You can use the example from the lecture but keep in mind that you are checking a list of numbers and not just one number.

Enter New Number List: 15 points

This option should allow the user to replace the list they entered when the program first executed. Meaning that the user can replace the list and then select any of the other options to get information about the list without needing to restart the program. A single method should be created to read in the list from the user. That method should be used to read in the initial list and for the Enter New Number List option. Do not have redundant code when a method can be used. No error checking is required.

Quit Program: 0 points (required)

This option should allow the user to exit the program. If this option is not selected the program should continue to execute.

Debugging Requirements: 10 points

Using an IDE such as eclipse place a breakpoint at one of the print statements for the Display List Statistics option. Then run the program in debug mode. Enter a list of numbers then select the option for displaying the list statistics. At this point the program execution should pause and the interface should switch into debug mode. Locate the variable explorer then take a screenshot that includes the variable explorer and the console output.

Submission:

Project 1 requires that you submit 4 files to Canvas as specified below.

- Image, Text, or Word file containing pseudo code of the “Display List Statistics” option.
 - Name: Pseudo_code_N# (replace N# with your UNF N#)
 - Example: Pseudo_code_n00123456.txt
- Image file containing flowchart that outlines the Number Statistics program.
 - Name: Flowchart_N# (replace N# with your UNF N#)
 - Example: Flowchart_n00123456.png
- Java source code file containing the source code for the Number Statistics program.
 - Name: Project1_N#.java (replace N# with your UNF N#)
 - Example: Project1_n00123456.java
- Image file containing the screenshot showing the variable explorer and console output during debugging.
 - Name: Debugging_N# (replace N# with your UNF N#)
 - Example: Debugging_n00123456.jpg

Sample Output:

Initial Run:

```
Enter List of Integers Separated by Spaces:
3 2 4 1 5
Please make a selection:
1) Display List Statistics
2) Display List Ordered
3) Number of Odd/Even
4) Check for Prime Numbers
5) Enter New List
exit) Quit Program
```

Display List Statistics:

```
1
Min: 1
Max: 5
Count: 5
Range: 4
Median: 3.0
Mean: 3.0
Mode: 1
Variance: 2.00
Standard Deviation: 1.41
```

Display List Ordered:

```
2
[0, 6, 7, 8, 9]
```

Number of Even/Odd:

```
3
Number Even: 2
Number Odd: 3
```

Count Number of Primes:

```
4
Number of Prime in list: 3
```

Program Execution:

```
Enter List of Integers Separated by Spaces:
3 2 4 1 5
Please make a selection:
1) Display List Statistics
2) Display List Ordered
3) Number of Odd/Even
4) Check for Prime Numbers
5) Enter New List
exit) Quit Program
```

1

Min: 1
Max: 5
Count: 5
Range: 4
Median: 3.0
Mean: 3.0
Mode: 1
Variance: 2.00
Standard Deviation: 1.41

Please make a selection:

- 1) Display List Statistics
- 2) Display List Ordered
- 3) Number of Odd/Even
- 4) Check for Prime Numbers
- 5) Enter New List
- exit) Quit Program

2

[1, 2, 3, 4, 5]

Please make a selection:

- 1) Display List Statistics
- 2) Display List Ordered
- 3) Number of Odd/Even
- 4) Check for Prime Numbers
- 5) Enter New List
- exit) Quit Program

3

Number Even: 2
Number Odd: 3

Please make a selection:

- 1) Display List Statistics
- 2) Display List Ordered
- 3) Number of Odd/Even
- 4) Check for Prime Numbers
- 5) Enter New List
- exit) Quit Program

4

Number of Prime in list: 3

Please make a selection:

- 1) Display List Statistics
- 2) Display List Ordered
- 3) Number of Odd/Even
- 4) Check for Prime Numbers
- 5) Enter New List
- exit) Quit Program

5

Enter List of Integers Separated by Spaces:

8 6 9 7 0

Please make a selection:

- 1) Display List Statistics
- 2) Display List Ordered
- 3) Number of Odd/Even
- 4) Check for Prime Numbers

5) Enter New List
exit) Quit Program

2

[0, 6, 7, 8, 9]

Please make a selection:

1) Display List Statistics
2) Display List Ordered
3) Number of Odd/Even
4) Check for Prime Numbers
5) Enter New List
exit) Quit Program

exit

Program Exiting