COMP 482 Project 0: Warm Up

Due: Never

Points: 0 points possible

Purpose: Students often have trouble understanding (believing?) the requirements given in a description are achievable and mandatory. The point of this project (that you will not submit) is intended to show you that they are both. You are welcome to use any portion of the professor's solution to this project in any future project.

Overview: You will write a java program which determines the median of a set of integer data items. The median of a set of data items is the "middle" value (ie the item which has an equal number of items smaller and larger). In the case where the number of data items is odd n = 2k-1, the result is the k^{th} smallest (which will also be the k^{th} largest). In the case where the number of data items is even, the result will be the average of the k^{th} and $(k+1)^{th}$ smallest (which will also be the average of the k^{th} and $(k+1)^{th}$ largest).

Details: The input will come from a file called input.txt which will be placed in the same directory as your java file. I'd suggest using a Scanner object to read the data, but note the file name is input.txt and nothing more (ie it is not src/data.txt or InPut.txt or myTestData.txt and is definitely not C:/Users/StudentName/482Projects/Project0/input.txt).

The first line of input.txt will have a single value which will be the number of data items. The remaining lines will be whitespace seperated integer values.

For some projects you might be constrained in how you compute/determine the result. For example, you might have to use a particular algorithm described in class or use a particular algorithm design technique or have a certain time/space efficiency. However, if nothing is mentioned (like for this project) any reasonable algorithm is permitted.

Your output will consist of a single line stating the median with 1 decimal place of accuracy. No other text or output should appear. While working on the project,you may want to have additional output to help you debug, but do not include this additional output in your final project.

You can discuss the algorithm to be used with anyone and consult any source (books, internet, etc). However, you are expected to write the code on your own with limited or no assistance from the professor, no assistance from others, and limited or no assistance from other sources (books, internet, etc). You should not share your source code with others. You should not submit source code written by someone else (with the possible exception of using some code in the professor's solution). In particular, you should not submit code written by students in prior semesters. In addition to being a clear example of academic dishonesty, it likely won't satisfy the project specifications (since my projects change each term).

I would strongly suggest you read through the professor's code to see how it follows the specifications given. The reason you can copy portions from the professor's solution for projects is because many students will claim they cannot understand or follow the specifications.

Picky, but required specifications: Your project must:

- be submitted via canvas (well, in this case there is no submission).
- consist of 1 or more dot-java files (no class files, zip files, input files or other files should be submitted). Each file must have your name and which project you are submitting as comments on the first 2 lines.
- not be placed into any package (for the java pedants, it must be in the default package).
- have one file called Project0.java (which must contain the main method).
- compile using the command 'javac Project0.java'.
- run using the command 'java Project0',

- accept input from a file called input.txt in the same directory as the java file(s) formatted precisely as described above.
- be submitted on time (early and multiple times is fine - do not worry if Canvas renames the file by appending a hyphen and digit - eg Project0-4.java).

If you fail to follow one of the above items, expect to receive a zero (recall that you have 1 and only 1 opportunity this semester to replace a project that gets a score of zero). If you follow the above items, your score will be graded based upon whether it

- is designed and formatted reasonably (correct indentation, no excessively long lines, no excessively long methods, has useful method/variable names, etc) and
- accomplishes the goal of the project. In other words, the output should be the correct answer, computed in the correct way, formatted correctly.

Sample execution: If input.txt contains

```
15
82 22      36
 11 31 4 31
    29
119 9 -2 57
29 73 93
```

the the output should be

```
31
```

If input.txt contains

```
50
 1  3  5  7  9 11 13 15 17 19 21 24 25 27
99 98 97 96 95 94 93 92 91 90 89 88 87 86
51 59 52 58 53 57 54 56 55

 2  4  6 28 26 24 22 20 18  8 10 12 14
```

Then the output should be

```
26.5
```

Stray Thoughts:
I will be using a recent version of Java (likely the current version, but if a new version is released I may upgrade).
You are generally allowed to use the standard features, classes, methods in Java. For example, I expect nearly all students will sometimes want to use either an array or java.util.ArrayList and the built in sort routine (either for arrays or ArrayLists). This is allowed as long as it doesn't violate a project requirement.
You can use as many or as few files as you feel appropriate, but the main method should be located in a file called Project0.java. Otherwise the project won't compile/run with the required commands.
Some IDEs default to placing java files into packages. This will likely cause the commands 'javac Project0.java' and/or 'java Project0' to fail. Either use an IDE that does not place java files into packages OR learn your preferred IDE well enough to avoid this issue OR delete any package lines before submission.
Students often decide to change or modify the format of the input or ouput. Sometimes it makes the project easier for them. Other times a student thinks it is an improved design. You may or may not be right, but don't change the input or output format anyway.
It is likely that many students won't read this far, but there is no need to let me know you've read this.

I suggest you finish you project several days in advance. This way you have time and opportunity to ask any last questions, you don't get caught by a power outage or similar issue, and verify that what you upload satisfies the requirements.

Your project should be written and understood by you. Helping or receiving help from others to figure out what is allowed/required is fine, but copying code is not. Significant shared source code indicates that you either did not write/understand what you submitted or you assisted another in submitting code they did not write or understand.

These project description files are getting excessively long, but I find that failing to state things which are common sense yields complaints.