

Java for Non Majors

CGS 3416: Spring 2016

Homework 6: 100 points

Due date: 11:59 PM 03/29/2016

Objective

The purpose of this assignment is to test your proficiency with Java loops, methods and arrays. It will also serve to check if you can integrate older concepts with newer ones.

For this assignment you are required to write two programs. Please email your files "Palindromes.java" and Sort.java to jayarama@cs.fsu.edu

Problem 1 - Palindrome Checker

You have somehow been sucked into a cliched 80's action movie, where you're Arnold Schwarzenegger's sidekick. You are approaching the final boss fight. However, he has locked himself into a vault which can only be opened by speaking a series of palindromic strings. Fortunately for you, the villain's sidekick is not very competent and has left a large binder of text lying around. 80's Arnold, recognizing his less than perfect English diction, has asked you to identify all the palindromes in the binder and read them to the lock. You decide to write a Java console application to make the job of identifying the palindromes easier for you.

A palindrome is defined as a string that reads the same forward and backward. For example, "Race car" is a palindrome.

Specifications

- This program is worth 50 points and should be called Palindromes.java
- Create a class with the appropriate name. (3 points)
- The main method should only have 2 lines. A call to the method and the print statement. (6 points)
- In a method called `checkPalin`, prompt the user to enter a number N and then accept N strings from the user. These strings will consist only of letters (both uppercase and lowercase), numbers and spaces. (10 points)
- Check if the strings are palindromes (ignoring case and whitespace). If a string happens to be a palindrome, add it to a result string. (18 points)
- After processing all the input, return the result string to main, where it is printed. (5 points)
- Prompts are the only print statements allowed outside the main method.
- The output should be printed in a single line, and the case of the letters in the input should be preserved. (5 points)
- You need not check for any errors.
- Please include comments wherever appropriate. (3 points)

Sample Run

Regular text is what's printed by your program. Underlined text is user input, shown here as a sample. You will not be printing the underlined parts in your program.

```
Enter the number of strings: 8
Enter the strings:
Race Car
Mountain Dew
BATMAN
Taco Cat
Stressed Desserts
Is Mayonnaise an instrument
swap paws
A Toyotas a Toyota
```

The palindromes are: Race Car Taco Cat Stressed Desserts swap paws A Toyotas a Toyota

Problem 2 - Insertion Sort

Dr. Insano is conducting an experiment. He is approaching random people on the street and ranking their competence based on the number of superheroes they can name. He has gathered his data meticulously. However, he has trouble getting the numbers in order. So, he has enlisted your help in writing a Java console application to sort the numbers in ascending order. He is very insistent that you use his preferred algorithm, given below, to do it.

```
loop i from 1 to Length(A)
  x = A[i]
  j = i - 1
  loop as long as j >= 0 and A[j] > x
    A[j+1] = A[j]
    j = j - 1
  end loop
  A[j+1] = x
end loop
```

Specifications

- This program is worth 50 points and should be called Sort.java
- Create a class with the appropriate name. (4 points)
- In the main method, read in the number of data points 'N' from the user. Check if the number is positive. If not, print an error message and exit. (6 points)
- Create an array of doubles of size 'N', and read in the data into the array. (8 points)
- Write a method called `sortArray`, that accepts a double array as a parameter and returns nothing. (6 points)
- Use the above algorithm to sort the array in ascending order. I am aware Java has a sort method. Please don't use that. It is a different algorithm. (10 points)
- Print the sorted array in the main method. The values should be printed in a same line, with a tab between values. (6 points)
- Please include comments wherever appropriate. (4 points)
- Include an explanation of the Insertion Sort algorithm as a part of your comments. (6 points)

Sample Runs

Regular text is what's printed by your program. Underlined text is user input, shown here as a sample. You will not be printing the underlined parts in your program.

Sample Run 1

Please enter the number of data points: 7

Enter the data:

10

3

5

9

15

2

6

The sorted data is:

2 3 5 6 9 10 15

Sample Run 2

Please enter the number of data points: -5

The number should be positive. Exiting.

Generic Guidelines

- Please add your name and FSUID as comments on the top of your program.
- Please make sure you're only using the concepts already discussed in class. These assignments are used to determine if you passed a certain learning milestone. So, please follow the specifications. Using concepts that are not in the specifications will result in a score of 0 points for that particular specification.
- Please make sure that you're conforming to output specifications (program name, expected inputs and outputs etc.). Your output must match the sample output exactly (especially the literal text on print statements).
- Please make sure your code is readable and well documented.
- Make sure to compile and run your program before you turn it in. Compilation errors can be costly.
- You can use an IDE for this assignment, but make sure your program compiles and runs on a terminal, since the program will be tested on a terminal.