

CS 304 Homework Assignment 4

Due: 11:59pm, Thursday, October 27th

This assignment is scored out of 60. It consists of 5 questions. When you submit, you are required to create a folder with your name (Last name first, then First name), CS304, HW4, e.g., LastName_FirstName_CS304_HW4. Type your answers into a text file (**only .txt, .doc, and .pdf file formats are accepted**) and save it in this folder. Put all your Java programs (***.java**) as well as output files in the same folder. Zip this folder, and submit it as one file to Desire2Learn. Do not hand in any printouts. Triple check your assignment before you submit. **If you submit multiple times, only your latest version will be graded and its timestamp will be used to determine whether a late penalty should be applied.**

Short Answers

P1. (8pts, 1pt each) Q1 on page 372

P2. (5pts) Q11 on page 375

P3. (4pts) Q27-a on page 464

P4. (5pts) Q46 on page 468

Programming Questions

P5. (38pts)

a. Completing the SortedPersonList class

You are provided with the files "**PersonNode.java**" and "**SortedPersonList.java**". You are required complete the methods in the latter file to implement a *sorted* linked list. The list is sorted according to the person's **m_ID** field. You need to write these methods:

contains(int ID) – This method takes an ID and checks whether a **PersonNode** associated with the given ID is in the list. If so, return **true**, otherwise, return **false**.

get(int ID) – This method takes an ID and checks whether a **PersonNode** associated with the given ID is in the list. If so, return the reference to the node, otherwise, return **null**.

add(int ID, String name) – This method takes an ID and a name and adds a **PersonNode** with these values into the list. The list must still be sorted in *ascending order* by ID numbers. If **add** is successful, return **true**. If the node already exists in the list, return **false**.

remove(int ID) – This method takes an ID and remove a **PersonNode** associated with the given ID from the list. If **remove** is successful, return **true**. If no such node exists, return **false**.

Note that you are only supposed to touch the above four methods. You are NOT allowed to create any other methods, instance variables, or make any changes to methods other than these four methods or files other than "SortedPersonList.java". Points will be taken off if you fail to follow this rule.

b. Code Testing

You are provided with a test driver implemented by "**TestPersonList.java**" (**Do not make any changes to this file!**) so there is no need to write your own.

Once you have completed the methods, you can run the test. You should create a plain text file named "**output.txt**", copy and paste the output (if your code crashes or does not compile, copy and paste the error messages) to this file and save it.

Grading Rubrics:

Code does not compile: -10

Code compiles but crashes when executed: -5

Changes were made to things other than the required methods: -5

Has output file: 5

Methods were implemented and code passes 33 test cases: 33

Sample output:

```
Test 1: size() ==> [Passed]
Expected: 0
Yours: 0
```

```
Test 2: add(1003, "Xiwei Wang") ==> [Passed]
Expected: true
Yours: true
```

```
Test 3: isEmpty() ==> [Passed]
Expected: false
Yours: false
```

```
Test 4: add(1005, "Neil Moore") ==> [Passed]
Expected: true
Yours: true
```

```
Test 5: add(1001, "Peter Kimmel") ==> [Passed]
Expected: true
Yours: true
```

```
Test 6: size() ==> [Passed]
Expected: 3
Yours: 3
```

```
Test 7: toString() ==> [Passed]
Expected: [1001 | Peter Kimmel] [1003 | Xiwei Wang] [1005 | Neil Moore]
Yours: [1001 | Peter Kimmel] [1003 | Xiwei Wang] [1005 | Neil Moore]

Test 8: contains(1001) ==> [Passed]
Expected: true
Yours: true

...

Test 28: toString() ==> [Passed]
Expected: [1003 | Damon Caskey] [1004 | Jeremy Wells] [1006 | Erich
Smith]
Yours: [1003 | Damon Caskey] [1004 | Jeremy Wells] [1006 | Erich Smith]

Test 29: contains(1004) ==> [Passed]
Expected: true
Yours: true

Test 30: contains(1005) ==> [Passed]
Expected: false
Yours: false

Test 31: remove(1002) ==> [Passed]
Expected: false
Yours: false

Test 32: remove(1003) ==> [Passed]
Expected: true
Yours: true

Test 33: size() ==> [Passed]
Expected: 2
Yours: 2

Total test cases: 33
Correct: 33
Wrong: 0
```