GIT Department of Computer Engineering CSE 222/505 - Spring 2017 Homework 2

Due date: March 06 2016 - 23:55 PM

Q1:

Give running times of each of the algorithms in proper notations. Explain your answers.

1.

for (int i = 0; i < n - 1; i++) {
 for (int j = i + 1; j < n; j++) {
 3 simple statements
 }
}

2.

public static int length(String str) {
 if (str == null || str.equals(""))
 return 0;</pre>

return 1 + length(str.substring(1));

Q2:

```
SOME_FUNCTION (A) n \leftarrow length[A] for j \leftarrow 1 to n-1 do smallest \leftarrow j for i \leftarrow j+1 to n do if A[i] < A[smallest] then smallest \leftarrow i exchange A[j] \leftrightarrow A[smallest]
```

else

}

- 1. What does the function do?
- **2.** Give the best-case and worst-case running times of the algorithm in Θ notation. Explain your answer.

Q3:

Prove that the running time of an algorithm is $\Theta(g(n))$ if and only if its worst-case running time is O(g(n)) and its best-case running time is O(g(n)).

Q4:

1. Express insertion sort as a recursive procedure.

- 2. Write a recurrence for the worst-case running time of the procedure. Explain your answer.
- 3. Solve the recurrence. Give detailed answer.

Q5:

Indicate giving detailed explanation whether f(n) = O(g(n)), $f(n) = \Omega(g(n))$ or $f(n) = \Theta(g(n))$ for the following:

- 1. $f(n) = n^{0.1}$, $g(n) = (\log n)^{10}$
- 2. $f(n) = n!, g(n) = 2^n$
- 3. $f(n) = (\log n)^{\log n}, g(n) = 2^{(\log_2 n)^2}$

Q6:

Rearrange your Homework 1 code using **Array**, **Array List**, and **Linked List** structures. Analyze and compare their performances. Add detailed information about your test and performance analysis method. Also, write detailed analysis results in your report.

RESTRICTIONS:

- Can be only one main class in project
- Don't use any other third part library

GENERAL RULES:

- For any question firstly use course news forum in moodle, and then the contact TA.
- You can submit assignment one day late and will be evaluated over twenty percent (%20).
- Register github student pack and create private project and upload your projects into github.
- Your appeals are considered over your github project process.

TECHNICAL RULES:

- Use given CSE222-VM to develop and test your homeworks (your code must be working on CSE222-VM), CSE222-VM download link will be given on Moodle.
- Implement <u>clean code standarts</u> in your code;
 - Classes, methods and variables names must be meaningful and related with the functionality.
 - o Your functions and classes must be simple, general, reusable and focus on one topic.
 - Use standart java code name conventions.

REPORT RULES:

- Add all <u>javadoc</u> documentations for classes, methods, variables ...etc. All explanation must be meaningful and understandable.
- You should submit your homework code, javadoc and report to Moodle in a studentid_hw#.tar.gz file.
- Use the given homework format including selected parts:

_

Detailed system requirements	
The Project usecase diagrams (extra points)	
Class diagrams	
Other diagrams	
Problem solutions approach	х
Test cases	х
Running command and results	х

GRADING:

No OOP design : -100 No interface : -95 - No method overriding : -95 No error handling : -50 No inheritance : -95 No polymorphism : -95 No javadoc documentation : -50 No report : -90 Disobey restrictions : -100 : -200 Cheating

- Your solution is evaluated over 100 as your performance.

- **Important!** For Q1, Q2, Q3, Q4 and Q5, answers without rational explanations will get 0 points.

CONTACT:

Nur Banu Albayrak