CS 3305: Data Structures

Assignment 06 – Big O



Note: Never hard-code test data in the test program, unless explicitly stated otherwise in the assignment. Always allow the user to enter the test data using a menu option.

GENERAL SUBMISSION REQUIREMENTS

Upload all files individually as specified, not as zip files, to Assignments in D2L. Do not email files.

Make sure your program compiles, runs and produces the correct output.

Ensure you have the correct file name(s), and author header, as specified in the Assignment.

Always use meaningful labels for prompts, inputs, and outputs.

Always use comments, indentation and whitespace as shown in examples.



**Objectives** The purpose of this lab is to reinforce concepts of runtime analysis ( Big O )

**Assignment 06 PART 1 Runtime Analysis (50 points) - Note Part 1 is a separate deliverable:**

1. Use Big O notation to estimate the time complexity of each of the following code fragments. Clearly show all steps of your work. Review examples in textbook pp. 822-827

1.1 sum = 0;

for (int i = 0; i < n; i++){

sum++;

}

1.2 sum = 0;

for (int i = 0; i < n; i++)

{

for (int j = 0; j < n; j++){

sum++;

}

}

1.3 sum = 0;

for (int i = 0; i < n; i++){

for (int j = 0; j < n \* n; j++) {

sum++;

}

}

1.4 sum = 0;

for (int i = 0; i < n; i++) {

for (int j = 0; j < i; j++) {

sum++;

}

}

1.5 sum 0;

for (int i = 0; i < n; i++) {

for (int j = 0; j < i \* i; j++) {

for (int k = 0; k < j; k++) {

sum++;

}

}

}

You do not need screenshots for A6-Part 1 and you do not need to copy and paste any code. If you work your problems out on paper, you may scan the work into a word doc or image. Phones have scan apps that do a good job. Name your file LastName-A6-Part-1-Runtime.docx or .pdf. Last step is to upload word/pdf file to D2L. You do not need to submit a .java file for A6-Part-1.

**Assignment 06 – PART 2** **Program Analysis (50 points) - Note Part 2 is a separate deliverable:**

1. Write a Java program that displays the current time and determine the time compexity (i.e. what is the big O)
2. Write a java program that will convert distance measurements from miles to kilometers for input of n numbers, and determine the worst case efficiency.

Specifically, the program needs to convert 0, 10, 20, 30, 40 , 50, 60, 70, 80, and 90 miles to kilometers displaying both the number of miles and the number of kilometers on the same line. Tip: 1 mile = 1.6 x km

// Name:

// Class:

// Term:

// Instructor:

// Assignment:

<your name>

CS 3305/ put your section number after the /

Fall 2023

Carla McManus

06-Part-2.1-Time or 06-Part-2.2-Miles

Capture a **READABLE** screenshot(s) of your program output and paste into a word/pdf document. Readable means readable! Screenshots ***should not be an entire desktop*** – use a snipping tool. After your output screenshots, copy and paste the source code for your exercise into the word/pdf doc.

Make sure that the source you’ve copied and pasted into your word/pdf is the same as that in your .java source file that you will upload to D2L.

Save doc for each part of Part 2 as files named LastName-A6-Part-2.1-Time.docx or .pdf and LastName-A6-Part-2.2-Miles.docx or .pdf.

Upload everything to D2L including the word/pdf doc ***and*** source code file.(.java file)

**MAKE SURE YOUR CODE HAS COMMENTS !** We are getting submissions without comments in the code. No comments = ( -20 ) points *per Part of the assignment.*

Do not submit zip files. **Late penalties of 10 % per day are in effect for this assignment.**