COMP2009 2022-23 ADE Coursework ONE (6.25%)
Wed. 01-MAR-2023

Time: 30 minutes.
Do not turn over page until instructed.
Answer ALL questions for a total of 25 marks.
Calculators are not permitted.
Write your answers on these sheets within the spaces provided.
Please write clearly.
Write your name & ID in the box below CLEARLY AND IN UPPER CASE LETTERS

FAMILY NAME:	
FIRST NAME(S):	
Student ID number:	
Signature:	

(Also, write your name at the top of every sheet; just in case the sheets become separated.)

Information that might, or might not, be helpful:

Definitions (reminders): "iff" = "if and only if", "s.t." = "such that"

Big-Oh: f(n) is O(g(n)) iff there exist c, n0 s.t.
f(n) <= c g(n) forall n >= n0

Big-Omega: f(n) is Ω(g(n)) iff there exist c>0, n0 s.t.
f(n) >= c g(n) forall n >= n0

Big-Theta: f(n) is Θ(g(n)) iff there exist c', c''>0, n0 s.t.
f(n) <= c' g(n) and f(n) >= c'' g(n) forall n >= n0

little-oh: f(n) is o(g(n)) iff forall c > 0, there exists n0 s.t.
f(n) < c g(n) forall n >= n0

To help distinguish "O" and "o", a note "[[little-oh]]" is added whenever it is used, otherwise it is a Big-Oh.

For completion by markers:
Total mark (out of 25):

Question 1. "Primitive Operation counting" [6 marks]

For each of the cases of Java fragments below, give a reasonable estimate of the count of the number of primitive operations they correspond to, and give a **BRIEF** (1-2 lines) justification.

a)
```
int c = 0;
```

count=

justification:

b)
```
h = h/2;   // where h is an int
```

count=

justification:

c)
```
k = k * 3;   // where k is an int
```

count=

justification:

d)
```
int[] A = new int[n];  // where n is an int
```

count=

justification:

e)
```
int[] B = A;     // using the A from part d
```

count=

justification:

Question 2. "Big- Oh, Omega, and Theta" [7 marks]

With $f(n) = 2\,n^2 + n$

a) From the definitions (e.g. see front page), prove or disprove the following statements. Show your working. If you claim the statement is true, then be clear about the values of c and n0 that you use. If you claim it is false, then justify your claim.

i. f(n) is $O(n^2)$

ii. f(n) is $\Omega(n^2)$

b) What do you conclude, if anything, about the Big-Theta behaviour of f(n)?

Question 3. Big-Oh Family [8 marks]

Answer the following multiple-choice questions by picking ONE right answer. Incorrect answers are not penalised (no negative marking)
CLEARLY CIRCLE ONE ANSWER – or clearly write the one letter answer. There is no need (or point) to justify your answer.

a) Given $f(n) = 3\ n^3 + n$ then which ONE of the following is correct:

A. f(n) is not $O(\ n^3\ )$

B. f(n) is $\Omega(n)$ but f(n) is not $\Omega(\ n^3\ )$

C. f(n) is $\Theta(\ n\ )$

D. f(n) is $\Theta(\ n^3\ )$

b) Given $f(n) = 3\ n^2 + n \log(n)$ then which ONE of the following is correct:

A. f(n) is not $\Omega\ (\ n \log n\ )$

B. f(n) is $O\ (\ n \log n)$

C. f(n) is $o\ (\ n^3\ )$ [[little-oh]]

D. f(n) is $o\ (\ n \log n)$ [[little-oh]]

c) Given $f(n) = 2^n + n^8$ then which ONE of the following is correct:

A. f(n) is $O(n^8)$

B. f(n) is $\Omega(n^8)$ and f(n) is $o(2^n)$ [[little-oh]]

C. f(n) is $O(2^n)$

D. f(n) is $\Omega(2^n * n^8)$

d) Given $f(n) = 2\,n\,(\log n)^2 + 3\,n\log(n) + 5\,n$
then which ONE of the following is correct:

A. f(n) is $\Theta(n\,(\log n)^2)$

B. f(n) is $\Theta(n\log(n))$

C. f(n) is $\Omega(n^2)$

D. f(n) is not $o(n^2)$ [[little-oh]]

Question 4. Stability in simple sorting algorithms [4 marks]

Consider the following routine to sort numbers in an integer array into increasing, or more accurately non-decreasing, order.

```
void sort(int arr[]){
  for( int i = arr.length-1; i > 0; i--){
     int pos_greatest = 0;
     for( int j = 0; j <= i; j++){
       if( arr[j] >= arr[pos_greatest])
          pos_greatest =  j;
     }
     if ( i != pos_greatest ) {
          temp = arr[i];
          arr[i] = arr[pos_greatest];
          arr[pos_greatest] = temp;
     }
  }
}
```

a) Identify the kind of sorting by clearly circling one of the following:

BUBBLE SELECTION INSERTION

b) Given the input A = [3, 2, 2'] - where 2 and 2' are equal (as in the lectures). Give the final state of A after applying the routine '`sort`'. **Briefly**, show your working.

c) State whether the sorting algorithm is stable or unstable by circling one of the following:

STABLE UNSTABLE