BRAC University
Department of Computer Science and Engineering (CSE)

CSE111: Programming Languages II

Semester: **Fall 2024** Time: **70 minutes**
Examination: **Mock Midterm** Full marks: **20**

Name: _____ ID: _____ Section: _____

(There are 3 questions in 3 pages. You must answer all. Do not repeat any parts of the code already given here)

Q1. Carefully observe the following driver code and expected output to design the **Series** class. Here, each series object keeps track of total episodes and number of watched episodes to determine whether a series has been watched completely or not.

[8 Marks]

Driver Code

```
public class Tester2 {
 public static void main(String[] args) {
  Series s1 = new Series();
  System.out.println("========1========");
  System.out.println(s1.showDetails());
  s1.updateInfo("Wednesday", "Mystery,
Supernatural", 15);
  System.out.println(s1.showDetails());
  System.out.println("========2========");
  Series s2 = new Series();
  s2.updateInfo("Dark","Sci-fi", 10);
  System.out.println(s2.showDetails());
  System.out.println("========3========");
  s1.watchEpisodes(10);
  s2.watchEpisodes(10);
  Series s3 = new Series();
  s3.updateInfo("Suits","Comedy,
Courtroom", 20);
  System.out.println("========4========");
  System.out.println(s3.showDetails());
  s1.watchEpisodes(2);
  System.out.println("You have watched
"+s1.watched+" episodes of "+s1.title);
  s3.watchEpisodes(15);
  s1.watchEpisodes(6);
  System.out.println("========5========");
  System.out.println(s1.showDetails());
  System.out.println(s2.showDetails());
  System.out.println(s3.showDetails());
  System.out.println("========6========");
 }
}
```

Expected Output

```
========1========
Show name: [BLANK]
Total Episodes: -1
Genre: NONE
Show is watched.
Updated information for the show: Wednesday
Show name: Wednesday
Total Episodes: 15
Genre: Mystery, Supernatural
Show is not watched.
========2========
Updated information for the show: Dark
Show name: Dark
Total Episodes: 10
Genre: Sci-fi
Show is not watched.
========3========
Watching 10 episodes of Wednesday
Finished show after watching 10 episodes of
Dark
Updated information for the show: Suits
========4========
Show name: Suits
Total Episodes: 20
Genre: Comedy, Courtroom
Show is not watched.
Watching 2 episodes of Wednesday
You have watched 12 episodes of Wednesday
Watching 15 episodes of Suits
Finished show after watching 3 episodes of
Wednesday
========5========
Show name: Wednesday
Total Episodes: 15
Genre: Mystery, Supernatural
Show is watched.
Show name: Dark
Total Episodes: 10
Genre: Sci-fi
Show is watched.
Show name: Suits
Total Episodes: 20
Genre: Comedy, Courtroom
Show is not watched.
========6========
```

Q2. Trace the following main method of tracing1 class and find the outputs:
(you must show the complete tracing table on your paper)

[6 Marks]

		OUTPUTS		
1	`public class tracing1 {`			
2	`    public static void main(String[] args) {`		-1	
3	`        MidTest p = new MidTest();`			6
4	`        int[] r = {3, 0, 7};`		0	
5	`        p.method(5,1);`			
6	`        p.r = r;`			
7	`        r[0] = p.method(p.a+p.b);`			
8	`    }`			
9	`}`			
10	`class MidTest {`			
11	`    int a = 7, b = 6;`			
12	`    int[] r = {5, 2};`			
13	`    int method(int n) {`			
14	`        n += b - 1;`			
15	`        this.b = (++r[(n+b)%2]) - r[1];`			
16	`        a = r[0] + r[1];`			
17	`        System.out.printf("%d %d %d\n", a, b, n);`			
18	`        return a + this.b;`			
19	`    }`			
20	`    void method(int n, int a) {`			
21	`        r[1] = a + b;`			
22	`        r[0] += this.r[a] + this.method(r[0]);`			
23	`        b += this.a-(n++);`			
24	`        System.out.printf("%d %d %d\n", a, b, n);`			
25	`    }`			
26	`}`			

Q3. Suppose you are working with a Company for developing a DNA sequence recognizing software. DNA can be represented with a sequence consisting of 4 proteins: A, T, C and G in many combinations. Your tasks are:

- Take a String input that contains several 4-length parts, where each part is separated by a '-' character.
- Find whether the String input is a correct DNA sequence. In other words, verify that the parts only contain the valid characters for a DNA: 'A', 'T', 'C', 'G'.

Finally, your task is to **find the least frequently appearing protein** among all the parts in the input sequence.

Carefully observe the following sample inputs and corresponding outputs and write a java program to complete the tasks described above.

[6 Marks]

Sample Input 1	**Sample Output 1**
ATTA-CGGA-ATCG-AATA-ATTA-AATA-ATCG-AATA	Least Occurring protein: C (3 out of 32)

Sample Input 2	**Sample Output 2**
CGTA-ATCM-ACTA-ATGA-CATA-ATCG	Error in input

(END of Questions. You can start writing after the following line)
