CS240 Fundamentals

Complete the following problems without a calculator.

1. **Functions**
   Given this function: $T(n) = 5 + n^2$
   (a) What are the values of:

   T(0) = _____ T(1) = _____ T(2) = _____

   (b) Consider an arbitrary value c:

   T(c) = _____ T(c+1) = _____

2. **Algebra**
   (a) Simplify the following ($T(n)$ from the question above):
   $\frac{n^2 + 3n}{n^2} =$ _____ $\frac{T(n)}{n} =$ _____
   (b) Solve the inequalities:

   $x + 5 < 8$ is true when: _____

   $x^2 - 4 \geq 0$ is true when: _____

3. **Powers of 2**
   (a) Write the powers of 2 from 2^0 to 2^{10}. The first three have been completed for you.

2^0	2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8	2^9	2^{10}
1	2	4								

   (b) Solve:

   $\log_2 8 =$ _____ $\log_2 512 =$ _____ $\log_2 2^{100} =$ _____

   (c) Solve:

   $2^{\log_2 1701} =$ _____ $2^{\log_2 x} =$ _____

4. **Summations**
   (a) Consider the summation $T(n) = \sum_{i=0}^{n} 5$.

   i. Determine the value of T(6) by writing out all terms in the sum.

   ii. Give a closed form solution for $T(n)$.

(b) Consider the summation $T(n) = \sum_{i=0}^{\lfloor \log_2(n) \rfloor} 2^i + 3$.

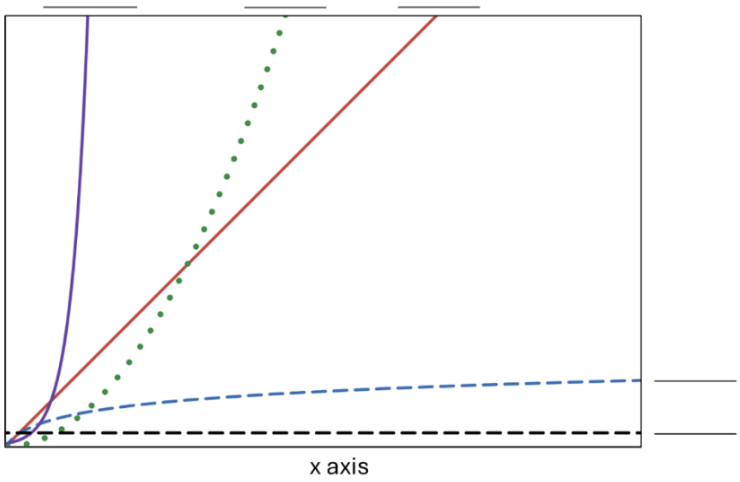
Notation reminder: The floor operator $\lfloor x \rfloor$ takes a real number and returns the largest integer that is less than or equal to that number. Essentially, it rounds down. Similarly, the ceiling operator $\lceil x \rceil$ rounds up.

i. Determine the value of T(14) by writing out all terms in the sum.

5. **Graphs**

Fill in the blanks to match each function to the line that best represents its growth (graph not to scale).

Possible functions: $A(x) = x$, $B(x) = \log_2 x$, $C(x) = x^2$, $D(x) = 4$, $E(x) = 2^x$.



6. **Code Tracing and Counting**

```
public static void printAs(int n) {
    for (int i = 0; i < n; i++) {
        System.out.print("A");
        System.out.print("A");
    }
}
```

(a) For `printAs(5)`, how many A's get printed? _____

(b) Let $A(n)$ be the number of A's printed:

A(1) = _____, A(2) = _____, A(3) = _____, A(4) = _____

(c) Generalize:

A(n) = _____

7. **More Counting**

```
public static void printBs(int n) {
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            System.out.print("B");
        }
    }
}
```

(a) For `printBs(2)`, how many B's get printed? _____

(b) Generalize:

B(n) = _____

8. **Even More Counting**

```
public static void printCs(int n) {
    System.out.print("C");

    for (int i = 0; i < n; i++) {
        System.out.print("C");
    }
}
```

Write a function representing the number of C's printed:

C(n) = _____

9. **ADT Selection**

For each scenario, choose the most appropriate Abstract Data Type / Java collection interface from `List`, `Set`, `Map`, `Queue`.

Scenario	**Answer (List / Set / Map / Queue)**
Keep a reading history in the order items were added; sometimes access by position.	
Look up a product's details given its SKU code.	
Record which users have completed onboarding; ignore repeats from the same user.	

10. **Collection Hiearchy**

Which of the following will compile? For each snippet, check one. If it does not compile, briefly explain the problem.

Check your answers using `jshell` on your laptop, or use a web-based jshell if that isn't working for you: `https://tryjshell.org/`

(a)
```
List<Integer> lst = new ArrayList<>();
```
◯ Compiles ◯ Does not compile

(b)
```
Queue<Integer> q = new LinkedList<>();
```
◯ Compiles ◯ Does not compile

(c)
```
Iterable<Integer> itbl = new ArrayList<>();
```
◯ Compiles ◯ Does not compile

(d)
```
Iterable<Integer> itbl = new ArrayList<>();
itbl.add(7);
```
◯ Compiles ◯ Does not compile

(e)
```
Collection<Integer> coll = new ArrayList<>();
coll.add(5);
coll.add(6);
coll.get(1); // item at position 2
```
◯ Compiles ◯ Does not compile

(f)
```
// Integer is a subclass of Number.
List<Number> nums = new ArrayList<Integer>();
```
◯ Compiles ◯ Does not compile

(g)
```
Collection<String> map = new HashMap<>();
```
◯ Compiles ◯ Does not compile

11. **Iterators**

For each independent snippet below, write exactly what is printed. Also, indicate any exceptions that are thrown and the line where they occur.

(a)

```
List<String> s = Arrays.asList("A", "B");
Iterator<String> it = s.iterator();

System.out.print(it.next());
System.out.print(it.next());
System.out.print(it.next());
```

(b)

```
List<Integer> nums = Arrays.asList(1, 2, 3);
Iterator<Integer> it = nums.iterator();

it.next();
it.next();
it.remove();
System.out.println(it.hasNext());
System.out.println(nums);
```

(c)

```
List<String> list = new ArrayList<>(Arrays.asList("X", "Y"));
Iterator<String> it = list.iterator();

while (it.hasNext()) {
    System.out.print(it.next());
    it.remove();
}
System.out.println(list.size());
it.remove();
```

(d)

```
List<String> list = new ArrayList<>(List.of("A", "B", "C"));
Iterator<String> it = list.iterator();

for (String s : list) {
  System.out.print(s);
  System.out.print(it.next());
}
```

12. **(IF TIME) Final Code Tracing and Counting**

```
public static void printDs(int n) {
    for (int i = 0; i < n; i++) {
        for (int j = i; j < n; j++) { // Notice loop initialization!
            System.out.print("D");
        }
    }
}
```

(a) For `printDs(2)`, how many D's get printed? _____

(b) Let $D(n)$ be the number of D's printed:

D(1) = _____, D(2) = _____, D(3) = _____, D(4) = _____

(c) Generalize:

D(n) = _____