

| Points off | 1 | 2 | 3 | 4 | | Total off | Net Score |
|------------|---|---|---|---|--|-----------|-----------|
| | | | | | | | |

Your Name: _____

Your UTEID: _____

Circle your TA's Name: **Anthony Lilly** **Ashley Neal** **David K. Noah** **David T. Pranav** **Grace Skyler** **Henry Sam**

Instructions:

- There are **4** questions on this test. 100 points available. Scores will be scaled to 250 points.
- You have 2 hours to complete the test.
- Place your final answers on this test. Not on the scratch paper. **Answer in pencil.**
- You may not use a calculator or **outside resources of any kind** while taking the test.
- When answering coding questions, ensure you follow the restrictions of the question.
- Do not write code to check the preconditions.
- On coding questions, you may implement your own helper methods.
- On coding questions make your solutions as efficient as possible given the restrictions of the question.
- Test proctors will not answer any questions regarding the content of the exam. If you think a question is ambiguous or has an error, state your assumptions and answer based on those assumptions.
- When you complete the test show the proctor your UTID, give them the test and all the scratch paper, used or not, and leave the room quietly.

1. (2 points each, 50 points total) Short answer. Place your answer on the line next to or under the question. Assume all necessary imports have been made.

- If a question contains a syntax error or compile error, answer **compile error**.
- If a question would result in a runtime error or exception, answer **runtime error**.
- If a question results in an infinite loop, answer **infinite loop**.
- Recall when asked for Big O your answer shall be the most restrictive correct Big O function. For example, Selection Sort is average case $O(N^2)$, but per the formal definition of Big O it is correct to say Selection Sort is $O(N^3)$, $O(N^4)$ and so forth. Give the most restrictive, correct Big O function. (Closest without going under.)
- Assume $\log_2(1,000) = 10$ and $\log_2(1,000,000) = 20$.

A. Using the techniques and rules from lecture, what is the $T(N)$ of the following method?
 $N = \text{data.length}$.

```

private static int a(int[] data) {
    int t = 0;
    for (int i = 0; i < data.length; i++) {
        for (int j = 0; j < data.length; j++) {
            int x = data[i] - data[j];
            x = x * 2;
            t += x;
        }
    }
    return t;
}

```

- B. Using the techniques and rules from lecture, what is the $T(N)$ of the following method?
 $N =$ the parameter n . Assume the parameter n is even and $n \geq 2$. _____

```
// pre: n is an even number and n >= 2
private static int b(int n) {
    int r = 0;
    for (int i = 0; i < n; i += 2) {
        r += n % 10;
        r += n * n * n;
    }
    return r;
}
```

- C. What is the order of method c ? $N =$ the parameter n . _____

```
private static ArrayList<String> c(int n) {
    ArrayList<String> r = new ArrayList<>();
    for (int i = 0; i < n; i++) {
        r.add("X" + i % 10);
    }
    return r;
}
```

- D. What is the order of method d ? $N = \text{data.length}$. _____

```
private static ArrayList<int[]> d(int[] data) {
    ArrayList<int[]> r = new ArrayList<>();
    for (int x : data) {
        int[] t = new int[10];
        t[0] = x;
        r.add(0, t); // insert method, (position, value)
    }
    return r;
}
```

- E. What is the order of method e ? $N =$ the parameter n . _____

```
private static int e(int n) {
    int c = 0;
    for (int i = 1; i <= n; i++)
        for (int j = 1; j <= i; j++)
            c += i % j;
    return c;
}
```

F. What is the order of method f? N = the parameter n. _____

```
private static int f(int n) {
    int r = 0;
    for (int i = 1; i < n; i++) {
        final int LIMIT = i % 10;
        for (int k = 0; k < LIMIT; k++) {
            r += m1(i, k); // method m1 is O(1)
        }
    }
    return r;
}
```

G. What is the order of method g? N = data.length. _____

The methods m2, m3, and m4 are all $O(N)$ where N is the length of the array passed as an argument. Assume m2 returns true 50% of the time.

```
private static int g(int[] data) {
    int r = 0;
    for (int i = 0; i < data.length; i++)
        for (int j = 0; j < data.length; j++)
            if (m2(data, j))
                r += m3(data, i);
            else
                r += m4(data, j);
    return r;
}
```

H. What is output by the following code? _____

```
ArrayList<String> list = new ArrayList<>();
list.add("A");
list.add("B");
list.add(1, "C");
list.add(3, "D");
list.add(2, "E");
System.out.println(list);
```

I. A method is $O(N^2)$. It takes 2 seconds for the method to complete when $N = 50,000$. What is the expected time in seconds for the method to complete when $N = 200,000$?

J. What is output by the following code? _____

```
ArrayList<String> list = new ArrayList<>(5);
list.add("A");
list.add(3, "B");
list.add(1, "C");
list.add(1, "D");
System.out.println(list);
```

K. A method is $O(\log_2 N)$. It takes 1 second for the method to complete when $N = 1,000$. What is the expected time in seconds for the method to complete when $N = 1,000,000$?

L. A method is $O(N!)$. It takes 1 second for the method to complete when $N = 43$. What is the expected time in seconds for the method to complete when $N = 44$?

M. What is output by the following code? _____

```
ArrayList<Integer> list = new ArrayList<>();
list.add(14);
list.add(14);
list.add(13);
list.set(1, 19);
Iterator<Integer> it = new Iterator<>(list);
System.out.println(it.hasNext() + " " + it.next());
```

N. What is output by the following method if the Iterator object passed to the method is iterating over the following list:

["B", "Swift", "JS", "Java", "C", "Python", "R", "SQL"]

and the Iterator cursor is initially in between "Swift" and "JS"? _____

```
private static void n(Iterator<String> it) {
    while (it.hasNext()) {
        String s = it.next();
        if (s.length() < 3) {
            System.out.print(s + " ");
        }
    }
}
```

O. What is output by the following method if list initially contains the Integer values [5, 2, 5, 7, 3] ? _____

```
public static void o(ArrayList<Integer> list) {  
    Iterator<Integer> it = list.iterator();  
    int total = 0;  
    total += it.next() + it.next();  
    it.remove();  
    it.remove();  
    total += it.next();  
    System.out.print(total + " " + list);  
}
```

P. Which of the following are Java interfaces **not** allowed to have? _____

1. abstract methods
2. instance variables
3. constructors
4. default methods

Q. What is output by the following code? _____

```
ArrayList<String> list = new ArrayList<>();  
list.add(list.toString());  
list.add(12);  
list.add("GDC");  
System.out.println(list.toString());
```

R. What is the average case order of the following code? _____
Assume the Random constructor and nextInt method are both $O(1)$.

```
public static void methodC(ArrayList<Double> list) {  
    Random r = new Random();  
    for (int i = list.size() - 1; i >= 0; i--) {  
        int x = r.nextInt(3);  
        if (x == 1) {  
            list.remove(i); // remove based on position  
        }  
    }  
}
```

For questions S through Y on the following page, refer to the classes on page 7.

S. Consider the following lines of code:

```
Staycation s1 = new Vacation(5); // 1
Staycation s2 = new Overseas(5); // 2
```

Which of the lines causes a syntax error? _____

T. Consider the following lines of code:

```
Object o1 = new Overseas(5); // 1
Vacation v1 = new Overseas(); // 2
```

Which of the lines causes a syntax error? _____

U. What is output by the following code? _____

```
Staycation s3 = new Staycation();
System.out.println(s3);
```

V. What is output by the following code? _____

```
Vacation v3 = new Staycation();
System.out.print(v3.days());
```

W. What is output by the following code? _____

```
Vacation v4 = new Overseas(1000);
System.out.print(v4.toString());
```

X. Consider the following two methods.

```
public int diff(Vacation v) {
    return cost - v.cost;
}
public void inc(int add) {
    cost += add;
}
```

Which of the following is true? _____

1. Adding method `diff` to the `Vacation` class will result in a syntax error.
2. Adding method `inc` to the `Overseas` class will result in a syntax error.

Y. What is output by the following code? _____

```
Overseas o1 = new Overseas(100);
Overseas o2 = new Overseas(100);
Vacation v7 = new Overseas(100);
System.out.print(o1.equals(o2) + " " + o2.equals(v7));
```

For questions S through Y, refer to the following classes.

```
public class Vacation {
    private int cost;

    public Vacation(int c) {
        cost = c;
    }

    public String toString() {
        return "V" + cost;
    }
}
```

```
public class Staycation extends Vacation {

    public Staycation() {
        super(10);
    }

    public int days() {
        return 3;
    }
}
```

```
public class Overseas extends Vacation {
    public Overseas(int c) {
        super(c);
    }

    public String toString() {
        return "OS";
    }
}
```

2. (18 points) To demonstrate encapsulation and the syntax for building a class in Java, we developed a `GenericList` class that can store elements of any data type. **For this question, the elements of the `GenericList` implement the `Comparable` interface.** Recall our `GenericList` class stores the elements of the list in the first size elements of a native array. An element's position in the list is the same as the element's position in the array. The array may have extra capacity. This `GenericList` class does NOT allow elements of the list to be null.

Complete the following instance method for the `GenericList` class:

```
/* pre: other != null.
post: per the problem description. Neither this or other are
altered.*/
public GenericList<E> getMaxList(GenericList<E> other)
```

The method creates and returns a new list with each position containing the larger of the two elements at the same position from this and other. If one list is longer than the other, the remaining elements of the longer list are included in the result.

Examples of lists returned by this method. The values shown are Integer objects.

```
[ 5, -5, 7] this
[10, -5, 3, -4, -3] other
[10, -5, 7, -4, -3] GenericList returned by getMaxList
```

```
[] this
[-5, -4, 0] other
[-5, -4, 0] GenericList returned by getMaxList
```

```
[5, 10, 7, 3, -10] this
[5, 12, 0] other
[5, 12, 7, 3, -10] GenericList returned by getMaxList
```

The `GenericList` class for this question:

```
public class GenericList<E extends Comparable<E>> {
    private E[] con;
    private int size;

    public GenericList() { con = (E[]) new Object[10];}
```

You may not use any methods from the `GenericList` class other than the given constructor unless you implement them as a part of your solution.

Do not use any other Java classes or methods except native arrays and the `compareTo` method.

Recall `c1.compareTo(c2)` returns an `int < 0` if `c1 < c2`, `0` if `c1` equals `c2`, and an `int > 0` if `c1 > c2`.


```
/* pre: other != null. post: per the problem description. Neither this  
or other are altered.*/  
public GenericList<E> getMaxList(GenericList<E> other) {
```

3. (18 points) Complete an instance method for the Names class from assignment 3 that removes NameRecord objects that are **not** ranked in the last n decades from the Names object calling the method. n is a parameter to the method.

```
/* pre: 0 < n < NUM_DECADES. post: remove all NameRecord objects from
this Names object that are not ranked in the last n decades.
Return the number of NameRecords removed. */
public int removeNotRanked(int n) {
```

All NameRecord objects stored in the instance variable named data have the same number of ranks. Consider this example where NUM_DECADES = 11:

| name | ranks | | | | | | | | | | |
|---------|-------|-----|-----|-----|-----|---|---|-----|---|---|---|
| "Abe" | 248 | 328 | 532 | 764 | 733 | 0 | 0 | 0 | 0 | 0 | 0 |
| "Buffy" | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 678 | 0 | 0 | 0 |

The method would remove Abe if n were 6. The NameRecord with the name Abe is not ranked in the last 6 decades. Buffy would not be removed because it is ranked in at least one of the last 6 decades. If n equaled 1, 2, or 3 the method would remove both Abe and Buffy.

The NameRecord class for the question:

```
public class NameRecord {

    /* Return the rank of this NameRecord for the given decade.
    pre: 0 <= decade < NUM_DECADES for the Names object this belongs to.
    The returned value is between 1 and 1000 inclusive or 0 if not ranked.*/
    public int getRank(int decade)
```

The Names class for the question:

```
public class Names {
    private ArrayList<NameRecord> data;
    public final int NUM_DECADES; /* number of decades for each
NameRecord in this Names object. */
```

You may use the following methods from the ArrayList class: int size(), E remove(int pos), E get(int pos). Your answer may NOT use Iterators or the for-each loop.

Do not use any other Java classes or methods. Do not create any new data structures. Do not add any methods to the NameRecord class.

Complete the method on the next page.

```
/* pre: 0 < n < NUM_DECADES. post: remove all NameRecord objects from
this Names object that are not ranked in the last n decades.
Return the number of NameRecords removed. */
public int removeNotRanked(int n) {
```

4. Other Data Structures (14 points) Complete the removeAll instance method for the ArrayMap class. Recall a map consists of key-value pairs. Each key maps or corresponds to a value

```
/*pre: keys != null, no elements of keys == null.  
post: per the problem description. */  
public void removeAll(Object[] keys)
```

The method accepts an array of Objects that represent potential keys in the map. All keys in the map that are present in the array passed to the method are removed from the map, along with their associated values.

The ArrayMap class uses a 2d array of Objects with 2 rows to store the key-value pairs. Here is a small example of the 2d array in an ArrayMap object that stores the frequency of words. Of course the elements would actually be references to objects. The /'s indicate elements that are extra capacity. The size of the map in this example is 4 as there are 4 key-value pairs.

| | | | | | | |
|-------|-------|-------|--------|--------|---|---|
| key | "the" | "can" | "grit" | "book" | / | / |
| value | 10 | 21 | 10 | 5 | / | / |

The first row of the 2d array stores the map's keys and the second row stores the associated value for a key at the same index. **The map does not allow null keys or null values. Like the HashMap class the ArrayMap class stores its keys in no discernible order. From the client's perspective the keys can be in any order and that order may change.**

The ArrayMap class is generic based on the Java inheritance requirement and polymorphism, not Java generics.

The ArrayMap class does **not** implement the Java Map interface,

Like the GenericList class we developed in lecture, the array in the ArrayMap class may have extra capacity, and thus the ArrayMap has a size variable to track the number of key-values pairs in the map.

The ArrayMap class for this question:

```
public class ArrayMap {  
    /* kvPairs.length == 2. The key-value pairs are stored at indices  
       [0, size). There may be extra capacity. */  
    private Object[][] kvPairs;  
    private int size; // The number of key-value pairs in this Map.
```

Do not use any other Java methods or classes except the equals method.

Do not use any other ArrayMap methods unless you implement them as a part of your answer.

Do not create any new data structures. Of course you can use the native arrays length field.

Complete the method on the next page.

```
/*pre: keys != null, no elements of keys == null.  
post: per the problem description. */  
public void removeAll(Object[] keys)
```

