CS2030 Programming Methodology
Semester 2 2019/2020

20 February 2020
Problem Set #5

1. The following static generic method `max3` that takes in an array of generic type `T` that such that `T` implements the `Comparable` interface.

```
static <T extends Comparable<T>> T max3(T[] arr) {
    T max = arr[0];
    if (arr[1].compareTo(max) > 0) {
        max = arr[1];
    }
    if (arr[2].compareTo(max) > 0) {
        max = arr[2];
    }
    return max;
}
```

What happens if we replace the method header with each of the following:

(a) `static <T> Comparable<T> max3(Comparable<T>[] arr)`

(b) `static <T> T max3 (Comparable<T>[] arr)`

(c) `static Comparable max3(Comparable[] arr)`

2. Suppose a `Fruit` class implements the `Comparable` interface, and `Orange` is a sub-class of `Fruit`, how would you change the `max3` method header in question 1 such that the parameter type is `max3` is `List<T>` instead? You should aim to make the method as flexible as you can.

3. Compile and run the following program fragments and explain your observations.

(a) `import java.util.List;`

```
class A {
    void foo(List<Integer> integerList) {}
    void foo(List<String> StringList) {}
}
```

(b)
```
class B<T> {
    T x;
    static T y;
}
```

(c)
```java
class C<T> {
    static int b = 0;

    C() {
        this.b++;
    }

    public static void main(String[] args) {
        C<Integer> x = new C<>();
        C<String> y = new C<>();

        System.out.println(x.b);
        System.out.println(y.b);
    }
}
```

4. Which of the following code fragments will compile? If so, what is printed?

(a)
```java
List<Integer> list = new ArrayList<>();
int one = 1;
Integer two = 2;

list.add(one);
list.add(two);
list.add(3);

for (Integer num : list) {
    System.out.println(num);
}
```

(b)
```java
List<Integer> list = new ArrayList<>();
int one = 1;
Integer two = 2;

list.add(one);
list.add(two);
list.add(3);

for (int num : list) {
    System.out.println(num);
}
```

(c)
```java
List<Integer> list = Arrays.asList(1, 2, 3);

for (Double num : list) {
    System.out.println(num);
}
```

(d)
```java
List<Integer> list = Arrays.asList(1, 2, 3);

for (double num : list) {
    System.out.println(num);
}
```

(e)
```java
List<Integer> list = new LinkedList<>();
list.add(5);
list.add(4);
list.add(3);
list.add(2);
list.add(1);

Iterator<Integer> it = list.iterator();
while (it.hasNext()) {
    System.out.println(it.next());
}
```