CS 220 Software Design II — Spring 2022

# In-class Exercises 01

*University of Wisconsin - La Crosse* — *February 4*

**1**. The formula below represents an alternating series (i.e., a series where terms alternate signs):

$$\sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n} = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \frac{1}{5} + \ldots$$

Write a `private static`, non-void method named `altSeries` as described in the JavaDoc comment below. Remember that the `Math.pow(a, b)` method calculates $a^b$ (returning a `double`).

```
/**
 * Display the alternating series, and return the accumulated value.
 * For example, if the limit is equal to 4 then this method will
 * display the following to the console:
 *    1 - 1/2 + 1/3 - 1/4
 * Then it will return the value: 0.5833
 *
 * @param limit Limit of the number of iterations (int)
 * @return Accumulated value (double)
 */
```

**Solution:**

```java
    double sum = 0;

    for (int i = 1; i <= limit; ++i) {
        if (i == 1) {
            System.out.print("1");
        } else if (i \% 2 == 0) {
            System.out.print(" - 1/" + i);
        } else {
            System.out.print(" + 1/" + i);
        }

        sum += Math.pow(-1.0, i-1) / i;
    }

    return sum;
}
```

Consider the following classes for the next two problems.

```java
public class Airplane {
    protected int numPassengers;
    public Airplane(int n) {
        numPassengers = n;
    }
    public String fly() {
        return "whoosh";
    }
    public void board(int n) {
        numPassengers += n;
    }
    public int getPassengers() {
        return numPassengers;
    }
}
```

```java
public class BuddyJet extends Airplane {
    protected int engines;
    public BuddyJet(int n, int e) {
        super(n*2);
        engines = e;
    }
    public String fly() {
        String str = "WHOOSH";
        for (int i = 0; i < engines; ++i) {
            str += "!";
        }
        return str;
    }
    public int getPairs() {
        return numPassengers/2;
    }
}
```

**2**. What is printed to the console as a result of this code snippet?

```java
Airplane a = new BuddyJet(1, 2);
System.out.println(a.getPassengers() + ": " + a.fly());
a.board(1);
System.out.println(a.getPassengers() + ": " + a.fly());
```

**Solution:**

```
2: WHOOSH!!
3: WHOOSH!!
```

**3**. Cross out the lines of code that are syntactically invalid. In the space below, for each line you cross out, note the line number and **why** that line is invalid.

```java
1 BuddyJet b = new Airplane(2);
2 Airplane a = new BuddyJet(1, 2);
3 System.out.println(a.getPassgengers());
4 System.out.println(a.getPairs());
5 BuddyJet c = new BuddyJet(1, 2);
6 System.out.println(c.board(3));
```

**Solution:**

```
1: invalid type conformance
4: method will not be available for a
```