Day 7: Programmer-Defined Classes

> Please help NCC conserve resources by not printing this document on campus.

The following questions are meant to gauge your understanding of programmer-defined classes. Answer as many as you can, and make note of the ones you can't, because those are the questions you need to review most before today's exam. ***Do not ask your classmates for help*** – this is not an assignment, and it doesn't matter how many answers you get right. What matters is that you understand your strengths and weaknesses going into the first exam.

You should be able to easily define all of the **boldfaced** terms in this exercise. If not, you have a lot of catching up to do. Be sure to ask for help and/or advice.

1. Name three things that can be found in a **data-type class definition**.

2. A method that belongs to the `String` class can only be called on... what?

3. What is the difference between a **formal parameter** and an **actual parameter**?

4. What do **formal parameters** and **local variables** have in common? What's different about them?

5. True or false: A **call** to a **method that returns a value** should be part of another **statement**.
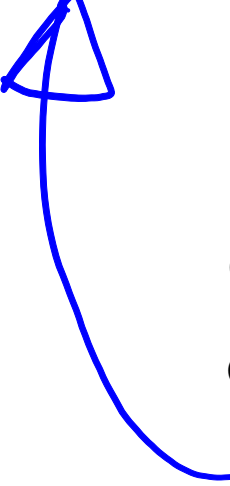
6. What happens when a **return statement** is executed?

7. Consider this method header:

```
1 public boolean equalSuit( Card otherCard )
```

Declare and instantiate all of the objects and variables that are needed before calling the `equalSuit` method. Then write a call to the method so that either "The cards have the same suit" or "The cards have different suits" is displayed.

8. Consider this method definition:

```
1 public boolean equalSuit( Card otherCard ) {
2     if( this.suit.equals( otherCard.suit ) == true )
3         return true;
4     else
5         return false;
6 }
```

(a) To what does `this` refer? ~~suit is string~~ *Card the input/current instance*

(b) What is the data type of `this.suit`? String

(c) Of what class is the `equals` method a member? String

(d) What type of data is sent as an actual parameter to the `equals` method? String

(e) What type of data is returned by the `equals` method? boolean

(f) How can this method be written more compactly?