1. (3 points) What are variables used for in a Java program?
   To store information.

2. (5 points) What are the three attributes of every variable in Java?
   i. type
   ii. name
   iii. value

3. (3 points) What does this output?  
   ```java
   System.out.println("12345" + 54321);
   ```
   1234554321

4. (3 points) What is the syntax (BNF) of a while statement?
   `<while stmt> ::= while (<boolean expression>) <stmt>`

5. (3 points) What is the semantics of an while statement?
   a) evaluate the `<expression>`
   b) if true, execute the `<stmt>`; then return to a

6. (3 points) Assume there is a class named SoccerTeam. Declare a variable of type SoccerTeam named wu.
   ```java
   SoccerTeam wu;
   ```

7. (3 points) Assume you have a variable of type SoccerTeam named wu (see above!); write a line of code to instantiate a SoccerTeam and store that new SoccerTeam in the variable wu.
   ```java
   wu = new SoccerTeam();
   ```

8. (3 points) Write a statement to send your SoccerTeam instance a scores() message.
   ```java
   wu.scores(); //!!
   ```

9. (3 points) Assuming it exists, what method in the SoccerTeam class is invoked by
   ```java
   System.out.println(wu); ? toString()
   ```

10. (4 points) Circle the legal Java identifiers:
    ```java
    foo   Foo   bar12345   baz   123123foofoofoo   2hands
    IloveJava!   youLoveJava!   weAllScreamForJava
    ```

11. (2 points) Of those, by convention, which are classes?  Foo
12. (3 points) What are parameters used for?

To send information to methods.

13. (3 points) In a Java method, what is this?

The current object (the object that was sent the message that invoked the method we are currently in)

14. (4 points) Why is it important to keep your code simple?

So you can understand and thus debug it.

15. (6 points) Ruby (!) is facing west and there are no walls except the boundary walls. Write an instruction that will make her move straight ahead and pick up all the coins she finds until she reaches a wall (make sure to pick up the coins where she starts and at the wall).

```java
void cleanLine() {
    cleanCorner();
    while (!frontIsBlocked()) {
        move();
        cleanCorner();
    }
}

void cleanCorner() {
    while (nextToACoin()) {
        takeCoin();
    }
}
```

16. (5 points) Write a Ruby instruction, assuming she is not next to a coin, to turnLeft if she has exactly 2 coins in her bag, otherwise turnRight don't leave any coins!

```ruby
void left2() {
    turnRight();
    if (coinInBag()) {  // more than 0
        putCoin();
        if (coinInBag()) {  // more than 1
            putCoin();
            if (!coinInBag()) {  // 2 exactly!
                turnAround();
            }
        }
    }
    cleanCorner();
}
```

17. (5 points) Assume you have an Circle class, with an int variable, r (the radius of the circle). Write a method, contract() which will decrease r by 3.

```java
int contract() {
    r = r - 3;
}
```
18. Write a complete class named `WalrusBeach` with one `int` variable, `number` (i.e. the number of walruses on the beach); accessors for `number`, and public `String` `toString()` (that returns the class name and the value of its instance variable using `getNumber()` -- this last so we know `getNumber()` works!).

```java
class WalrusBeach {
    int number;

    int getNumber() {  
        return number;
    }

    void setNumber(int n) {  
        number = n;
    }

    public String toString() {  
        return "WalrusBeach!! number=": + getNumber();
    }
}
```
19. (10 points) Complete this class named Test. Fill in main so it creates two WalrusBeaches, named shortSands and pointLay, sets their lengths to 0 and 35000, respectively, and then souts them each to the Output window.

class Test {
    public static void main(String [] args) {

        WalrusBeach shortSands = new WalrusBeach();
        WalrusBeach pointLay = new WalrusBeach();

        pointLay.setNumber(0);
        shortSands.setNumber(35000);

        System.out.println(shortSands + pointLay);
    }
}

20. (3 points) How do you fix a bug you can’t find? You can’t.

21. (5 points) Write a method, named double, which returns twice the value if its int parameter.

    int twice(int x) {
        return x * 2;
    }

22. (Extra credit 10 points) Write a complete LimitedWalrusBeach class which is identical with WalrusBeach, except it pretends there are never more than 1000 walruses on any beach, by overriding the getNumber() method so it returns either the actual number of walruses on the beach, or 1000 (if there are more than 1000).

    class LimitedWalrusBeach extends WalrusBeach {

        int getNumber() {
            if (number > 1000) {
                return 1000;
            }
            return number;
        }
    }