This was the CS 601 midterm exam given in Fall 2015. It covers topics that we have not covered and you would not be expected to know. The goal of this sample exam is to give a sense of the length of exam and general types of questions you should expect on your midterm exam.

Question	Awarded Points	Maximum Points
Question 1		10
Question 2		10
Question 3		5
Question 4		5
Question 5		5
Question 6		5
Question 7		5
Question 8		10
Question 9		5
Question 10		5
Question 11		5
Question 12		10
Question 13		5
Question 14		5
Question 15		10
Total		100

1. (10 points) The following program has several problems with coding *style*. Identify at least **five**. Each problem identified will be worth two points. Make sure to *explain* each problem you specify.

```
public class student {
	public String name;
	student(String name, boolean isGrad, double theGrade) {
		this.name = name;
		this.is_Grad = isGrad;
		this.grade = theGrade;
	}

	public boolean isPassing() {
		if(is_Grad) {
			return this.grade > 80;
		} else {
			if(this.grade > 70) {
				return true;
			} else {
				return false;
			}
		}
	}
	public Boolean is_Grad;
	public double grade;
}
```

2. (10 points) This question refers to the following class definitions:

```
abstract class A {
      public abstract void method();
}

public class B extends A {
      public void method() {
            System.out.println("B method");
      }
}

public class C extends B {
      public void method() {
            System.out.println("C method");
      }
}
```

a. Will the following statement compile? For full credit explain your answer.

```
A a = new A();
```

b. Will the following statement compile? For full credit explain your answer.

```
C c = new B();
```

c. What is the output of the following statements?

```
A a = new C();
a.method();
```

3. (5 points) For the Lab 3 `ReentrantLock` class, some students used a Map to store the IDs of threads holding read locks and some students used a List. In the case of the Map, the key is the thread ID and the value is a count of the number of read locks held by that thread. In the case of the List, a thread ID would appear possibly multiple times if the thread held many read locks. Which approach do you think is more efficient? For full credit, explain *why*.

4. (5 points) In the `ReentrantLock` class, `tryLockRead` is a non-blocking method and `lockRead` is a blocking method. Explain the difference between a blocking and non-blocking method.

5. (5 points) What is one advantage of using a read/write lock over using synchronized methods or blocks when reading and writing data?

6. (5 points) The following method decrements a number until it hits 0 and uses a lock to guarantee that no two threads will be modifying the variable at the same time. Is it possible that `num` will ever be less than 0? Why or why not?

```
public void decrementNum() {
      if(num > 0) {
            lock.writeLock().lock();
            num--;
            lock.writeLock().unlock();
      }
}
```

7. (5 points) What is the difference between the two while loops shown below? Will they always behave the same way?

```
while(!tryLockRead()) {
      wait();
}
```

```
while(!tryLockRead()) {

}
```

8. (10 points) Explain everything that happens when you type a URL (e.g., http://www.google.com) in a browser and hit return.

9. (5 points) Explain what is meant by a *web service*. How is a web service different from a basic web site.

10. (5 points) What is a port number used for?

11. (5 points) If a client opened a connection to an HTTP server using a `Socket` and then did not send a request, what would happen? See below for an example.

```
Socket s = new Socket("google.com", 80);
int nextByte = s.getInputStream().read();
//what now?
```

12. (10 points) Which of the following are valid HTTP request lines?

1. `GET / http1.0`
2. `GET /`
3. `POST /test HTTP/1.0`
4. `GET /path?param=a value HTTP/1.0`
5. `GET /path?hereis=aparam HTTP/1.0`

13. (5 points) Consider the common HTTP response codes, include 200 OK, 404 Not Found, and 400 Bad Request. Suppose an API is specified as follows:

`/reviewers`

- Required parameter: `businessID`

If a server received a request at the URL below, how should it respond? For full credit, explain your answer.

```
http://host.com/reviewers?businessID=abc123&max=15
```

14. (5 points) Consider the API to access a list of reviewers as described in question 13, with endpoint `/reviewers` and parameter `businessID`. Is this API RESTful? Why or why not?

15. (10 points) Assume that `page` contains a string representation of an HTML page. The following code will remove all tags in the HTML document.

```
String REGEX = "(?is)<.*?>";
String newPage = page.replaceAll(REGEX, "");
```

a. Modify the REGEX to replace open bold tags. Hints: the element name for a bold tag is `b`. Show the entire, modified expression below.

b. Modify the REGEX to replace only closing tags, of any type. Show the entire, modified expression below.