You have 80 minutes to complete this exam. For coding questions, you do not need to include comments, and you should assume that all necessary packages have already been imported. Good luck!

<table>
<thead>
<tr>
<th>Question 1</th>
<th>/ 30</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question 2</td>
<td>/ 40</td>
</tr>
<tr>
<td>Question 3</td>
<td>/ 30</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>/ 100</td>
</tr>
</tbody>
</table>
1. Does the following code fragment compile? YES NO

```java
int y = 10;
y = 4 * y * y + 25 / y;
System.out.println(++y);
```

If it does compile, what is printed out to the console after running the three lines?

If it does not compile, circle the error in the code fragment.

1.b You are interested in writing a method that will print out all the factored versions of a square root. For example, the factored versions of $\sqrt{36}$ are:

\[
\begin{align*}
2\sqrt{9} & \quad 3\sqrt{4} \\
\end{align*}
\]

Your method should take as an argument a number $n$ and print out the factored versions of the square root of that number. Assume that $n$ is a nonnegative integer. For example, when $n = 36$, your method should print:

- \(2\sqrt{9}\)
- \(3\sqrt{4}\)

Complete the method below:

```java
public static void rootFactors(int n) {

}
```
1.c Two classes are given below. The `Student` class is in the `studentPKG` package, while the `StudentTest` class is in the default package. Check whether the four numbered statements compile, and write down your answers on the next page.

Student.java

```java
package studentPKG;

class Student {

    public String name;
    private int studentID;

    public Student(String name, int studentID) {
        this.name = name;
        setID(studentID); // ......................... #1
    }

    private void setID(int sID) { studentID = sID; }

    int getID() { return studentID; }
}
```

StudentTest.java

```java
import studentPKG.Student;

class StudentTest {

    public static void main(String[] args) {

        Student newStudent = new Student("", 12); // #2
        int snumber = newStudent.getID(); // #3
        newStudent.name = "Jason Brine"; // #4
    }
}
```
Does statement #1 compile?  YES  NO
If no, why does it not compile?

Does statement #2 compile?  YES  NO
If no, why does it not compile?

Does statement #3 compile?  YES  NO
If no, why does it not compile?

Does statement #4 compile?  YES  NO
If no, why does it not compile?
Question 2 - Classes and Objects (40 points)

The Harry Potter series may have ended but Harry Potter themed questions still exist. You’ve decided to create a small wizard dueling game. All classes exist in the same package. You’ve already been given code for a class called Spell. The Spell class consists of three public data members:

- **NAME**: the name of the spell
- **COST**: the magical cost of casting this spell [range: 0-100]. This value is subtracted from a wizard’s magic level every time he/she casts this spell.
- **DAMAGE**: the damage this spell does to an opponent’s health [range: 0-100].

```java
public class Spell {
    public final String NAME;
    public final int COST;
    public final int DAMAGE;

    public Spell(String name, int cost, int damage){
        NAME = name;
        COST = cost;
        DAMAGE = damage;
    }
}
```
2.a You now must create a *Wizard* class. The *Wizard* class has:

- A private String data member called *name*, that holds the wizard’s name.
- A private integer data member called *health*, that holds the wizard’s current health state (100=healthy | 0=dead).
- A private integer data member called *magic*, that holds the wizard’s current magic state (100 = lots of magic | 0 = no magic).
- A public constructor that only takes in a String for the wizard’s name. Wizards always start life with health=100 and magic=100.
- A public method *hitBySpell*(Spell spell), that alters the wizard’s health by subtracting the DAMAGE value held by the spell.

Complete the *Wizard* class below:

```java
public class Wizard {
    // Answer to question 2.b (next page) is inserted here.
}
```
2.b A wizard should also have the ability to cast a spell. The method is part of the Wizard class and will take in a Spell along with a Wizard to cast that spell on. Fill in the method `castSpell(Spell spell, Wizard wizard)` of the Wizard class, so that:

- The wizard casting the spell has the spell COST subtracted from his magic total.
- The wizard who is getting the spell cast on him/her has his/her `hitBySpell(Spell spell)` method called.

// The following method is in the Wizard class.

```java
public void castSpell(Spell spell, Wizard wizard){
}
```
2.c Wizards battle things out in an Arena. The Arena class just contains a main(String[] args) method to simulate duels. Add code in the Arena class to model this chain of events:

- A Spell called wami is created, with:
  - name = “WAMI”
  - cost = 10
  - damage = 15
- A Wizard called Ron is created, with:
  - name = “Ron”
- A Wizard named Harry is created, with:
  - name = “Harry”
- Harry casts the wami spell on Ron.
- Ron tries to retaliate, but accidently casts the wami spell on himself.

```java
public class Arena {
    public static void main(String[] args) {
        // Code to simulate duel goes here
    }
}
```
Question 3 - ArrayLists (30 points)

Every semester on add/drop day, you are going to help administrative officers to update course registration lists (adding and dropping students, changing registration status ...) with a java program. The program has a Student class that is already written. Assume that the registration status of a student in a course can only be "P/D/F" or "credit".

```java
public class Student {
    private String name;
    private String regstatus;
    private static final String PDF="P/D/F";
    private static final String CREDIT="credit";

    public Student(String name, String regstatus){
        this.name=name;
        setStatus(regstatus);
    }

    public String getName(){ return name;}

    public void setStatus(String regstatus){
        if (regstatus.equals(PDF)) this.regstatus=PDF;
        else this.regstatus=CREDIT;
    }
}
```

A CourseList class is used to model the list of students registered for a particular course. You have to implement the addStudent and dropCourse methods of the CourseList class. CourseList has a data member (list) that keeps the list of the students registered for the course. The addStudent method has no return value and takes in two arguments: a student name and his/her new registration status for this course. If the student is already in the list, the method will update the students’ status; otherwise it will add the student to the list. The Drop method deletes a list of students from the course list. The CourseList and Student classes are in the same package. Complete the addStudent and dropCourse methods of the CourseList class given next page. Assume that no two students have the same name.
public class CourseList {

    ArrayList<Student> list;

    public CourseList(ArrayList<Student> courseList){
        this.list=new ArrayList<Student>();
        for (Student s: courseList){
            this.list.add(s);
        }
    }

    public void addStudent(String studentName, String regStatus){
    }

    public void dropCourse(ArrayList<Student> s){
    }
}