Show all of your work clearly in the space provided or on the additional page at the end of the exam. **Read each problem carefully**. Note that the exam is double sided.

1. (4 points) Describe the difference between compilation and execution errors.

2. (8 points) Graphically illustrate the difference between primitives and reference types with memory diagrams like those drawn in class.

3. (4 points) Demonstrate at least two forms of syntax for adding comments to source code.

4. (8 points) Suppose `String number = in.next();` contains a sequence of digits, e.g., `86`. Write code that converts the value stored in `number` into an `int` and store it in a variable called `value`.

5. (8 points) Write code to display the message "Happy exam I" in a graphical window.

6. (8 points) Use the substring method from the String class to print to the console all but the first two characters of a String called word.

7. (10 points) Rewrite the following `for` loop as a `while` loop that is functionally equivalent.

```
for(int i=0; i<50000; i++) {
    System.out.println(i*i);
}
```

8. (3 points) In the previous problem, how many times will i be compared with 50000?

9. (15 points) For each of the following, indicate what will be displayed.

(a)

```
System.out.println(3.5 + 1 / 4);
```

(b)

```
System.out.println(3.5 + (1 / 4));
```

(c)

```
System.out.println((5>2 && 4<2) || (1!=1));
```

(d)

```
double x = 2;
System.out.println(x);
```

(e)

```
for(int i=0; i<2; i++) {
  System.out.println(i);
```

10. (12 points) Complete the following program that pops up a window asking the user to enter a word. If the user clicks OK without typing anything, the program should reprompt them to enter a word (over and over until they user actually types something before clicking OK). Once the user has entered a word, display the number of characters that the user typed in. Display the number of characters in a separate window.

E.g., If the user enters "five", a window should appear with the value "4" displayed in it.

```
import javax.swing.JOptionPane;

public ExamA {
    public static void main(String[] args) {









    }
}
```

11. (20 points) Complete the following program that asks the user to enter three integers:

- smallBars – the total number of 1 Kg bars available
- bigBars.– the total number of 5 Kg bars available
- desired – the total number of Kg of chocolate desired

Based on the values entered by the user, display the number of small bars and big bars needed to make a package with desired Kg of chocolate. If it is not possible to make a package with the exact amount of chocolate (breaking chocolate bars is prohibited), display an error message instead. Note: when possible use the big bars first to minimize the number of chocolate bars in the package.

Examples:

- smallBars==4, bigBars==1, desired==9 should result in: "Include 1 big bar(s) and 4 small bar(s) in the package"
- smallBars==4, bigBars==1, desired==10 should result in: "It's not possible to create the desired package"
- smallBars==4, bigBars==2, desired==11 should result in: "Include 2 big bar(s) and 1 small bar(s) in the package"

```
import java.util.Scanner

public class ExamB {

    public static void main(String[] args) {
        System.out.println("Enter the desired number of Kgs of chocolate");
```

Additional space — indentify which problem your work is associated with.