

# SE1011: Exam 2 Name:

---

This is a closed-book, closed-computer, etc. exam. You may use one 8.5"x11" sheet of notes, which you will turn in with your exam. Review all questions before you get started. Use the page at the end of the exam for extra work. The exam is printed double-sided. Show all work.

1. (15 pts.) Given a String str, write a sequence of Java statements to print out the number of occurrences of the letter "u" in the string.

```
String str = in.next();
```

2. (15 pts.) Complete the program below so that the following are true:
- The program uses the GUI for input/output.
  - The program displays only the letters “a” and “p” in the same order they occur in the String str. For example, the user enters “apple” the program displays “app”. If the user enters “**pineapples are cool**”, the program prints “pappa”
  - The program crashes if the user presses cancel. (No extra credit for fixing it.)

```
include javax.swing.JOptionPane;
```

```
class Exam {
```

```
    public static void main(String[] ignored) {
```

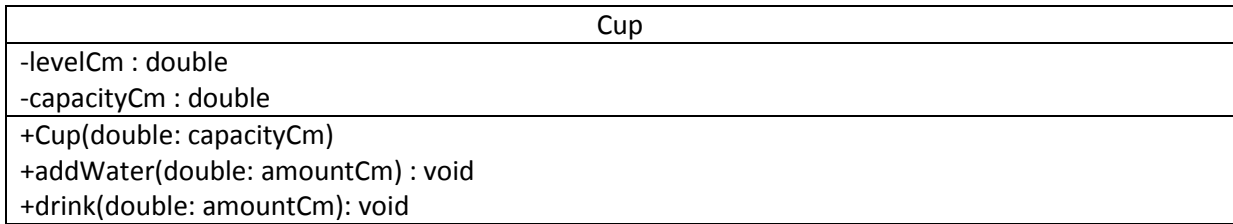
```
        String str = JOptionPane.showInputDialog("Enter some text");
```

```
    }
```

```
}
```

3. (25pts.) Short answer problems. Be clear & concise.
- a. (4 pts.) What is the difference between a “public” and “private” method?
  
  
  
  
  
  
  
  
  
  
  - b. (5 pts.) What is the role (purpose) of a constructor?
  
  
  
  
  
  
  
  
  
  
  - c. (4 pts.) Where do the values of a method’s parameters (arguments) come from? For example, where do x & y come from for the method `public void myMethod(int x, double y)`?
  
  
  
  
  
  
  
  
  
  
  - d. (4 pts.) If I have two instances (copies) of a class, how will they be different?
  
  
  
  
  
  
  
  
  
  
  - e. (4 pts.) What happens to a local variable when we reach the end of a method?
  
  
  
  
  
  
  
  
  
  
  - f. (4 pts.) Why would I want to make a method private?

4. (35 pts.) The following questions refer to this UML diagram:



```
public class Cup {
```

a. (5 pts.) Write the instance variables:

b. (10 pts.) Write the constructor for the class. The constructor should set the capacity as specified by the argument, and set the level to 0.

(3. Continued from previous page)

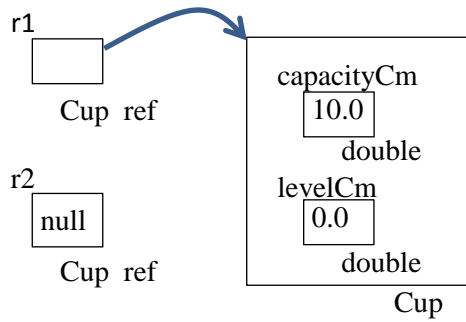
c. (10 pts.) Write the `addWater` method for the class. The `levelCm` in the cup should rise by the `amountCm`. If the `levelCm` would go over the `capacityCm`, it should be set to `capacityCm`.

d. (10 pts.) Write the `drink` method for the class. The `amountCm` should be removed from the `levelCm`. If the `levelCm` would fall below 0, it should be set to 0.

```
} // This is the end of the Cup class.
```

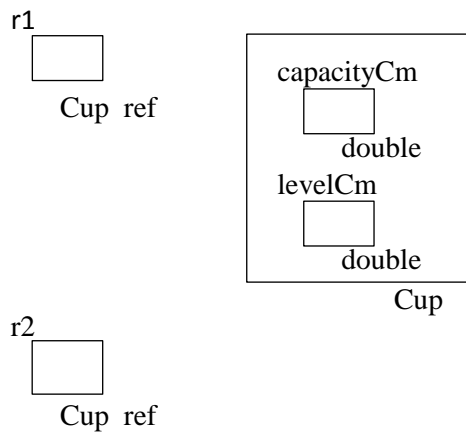
5. (20 pts.) Complete the following diagram that illustrates the state of memory at four points in a program on the right. Step a has been filled in entirely as an example. The Cup class is implemented as described in problem 4.

//a



```
Cup r1=null;
Cup r2=null;
r1 = new Cup(10.0);
// a
r1.addWater(5.0);
// b
r2 = new Cup(15.0);
r1.drink(4.0);
r2.addWater(20.0);
// c
```

//b [COMPLETE THIS]



//c [COMPLETE THIS]

You may use this page for extra work. Please write on the original problem page that you are using this space, and write here the original problem number.

# Partial Java API

---

Some of these methods should be used with an object, some should be used with the class name. Some methods show example arguments instead of the parameter definitions.

```
double Double.parseDouble(String str)
int Integer.parseInt(String str)
void JOptionPane.showInputDialog(null,String str)
void JOptionPane.showMessageDialog(null, String str)
long Long.parseLong(String str)
double Math.abs(double a)
double Math.acos(double a)
double Math.asin(double a)
double Math.atan(double a)
double Math.atan2(double a, double b)
double Math.cos(double a)
double Math.log(double a)
double Math.log10(double a)
double Math.max(double a, double b)
double Math.min(double a, double b)
double Math.pow(double a, double b)
double Math.PI
double Math.sqrt(double a)
double Math.sin(double a)
double Math.tan(double a)
void System.out.print()
void System.out.println()
Scanner.Scanner(System.in)
int Scanner.nextInt();
String Scanner.next();
String Scanner.nextLine();
char String.charAt(int pos);
int String.length();
```