

Family Name: .....

Other Names: .....

Student ID: .....

Signature .....

## Introduction to Programming: Test 2

2016, Nov 14 \*\* WITH SOLUTIONS \*\*

### Instructions

- Time allowed: **90 minutes**
- Answer **all** the questions. There are 45 marks in total.
- Write your answers in this test paper and hand in all sheets.
- If you think some question is unclear, ask for clarification.
- Brief Java documentation is provided with the test
- This test contributes 15% of your final grade  
(But your mark will be increased to your exam mark if that is higher.)
- You may use dictionaries.
- You may write notes and working on this paper, but make sure your answers are clear.

### Questions

### Marks

1. Understanding if	[8]	<input type="text"/>
2. Understanding while	[6]	<input type="text"/>
3. Writing loops	[8]	<input type="text"/>
4. Understanding Files	[6]	<input type="text"/>
5. Writing Loops with Input	[8]	<input type="text"/>
6. Writing Loops with Files	[9]	<input type="text"/>
	TOTAL:	<input type="text"/>



**Question 1. Understanding if****[8 marks]**

Complete the following printSmallest method which has three parameters, a, b, and c, which are all numbers. It should print out the minimum value of the numbers.

For example,

```
printSmallest(5, 3, 7)
```

should print out

```
Minimum = 3
```

```
public void printSmallest (double a, double b, double c){
    if (a<=b && a<=c) {
        UI.println ("Minimum = " + a);
    }
    else if (b<=a && b<=c) {
        UI.println ("Minimum = " + b);
    }
    else
        UI.println ("Minimum = " + c);
}
//another way:
double min = a;
if (b < min) { min = b; }
if (c < min) { min = b; }
UI.println ("Minimum = " + min);
}
```

## Question 2. Understanding while

[6 marks]

Consider the following doList method.

```
public void doList(int num) {  
    Ul.println("List:");  
    int n = 3;  
    while (n < num) {  
        Ul.println("n: " + n);  
        n = n + 2;  
    }  
    Ul.println("Done");  
}
```

n:

What will doList(10) print?

```
List:  
n: 3  
n: 5  
n: 7  
n: 9  
Done
```

What will doList(5) print?

```
List:  
n:3  
Done
```

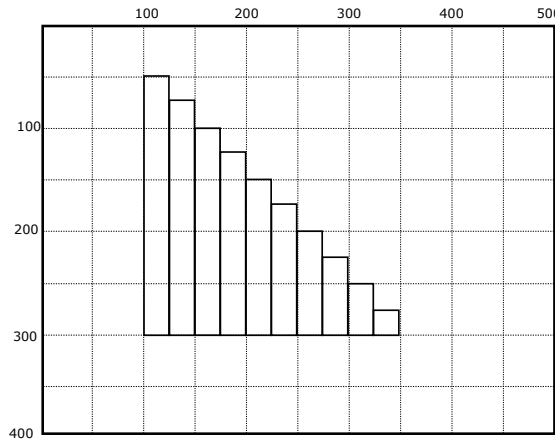
### Question 3. Writing loops

[8 marks]

Complete the following drawSteps method so that it uses a loop to draw a series of rectangles.

- The parameters specify the left top corner of the first rectangle and its height.
- The width of each rectangle is 33.
- The height of each rectangle is 25 less than the one before it
- The smallest rectangle must be at least 20 high.

For example, drawSteps(50, 100, 230) should draw rectangles like this:



```
public void drawSteps(double left , double top, double height) {  
    while (height >= 20) {  
        UI.drawRect(left , top, 33, height);  
        left = left + 33;  
        top = top + 50;  
        height = height - 50;  
    }  
  
    // another alternative :  
  
    int n = 0;  
    while (height - n*25 >= 20) {  
        UI.drawRect(left + n*33, top + n*25, 33, height - n*25);  
    }  
  
}
```

**Hint:** Use a while loop that draws each rectangle and changes the values of left, top and height.

#### Question 4. Understanding Files

[6 marks]

A company keeps summary information about its employees in files.  
The format of the files is:

- The first line contains the target number of sales per person.
- Each line after that contains a name and the number of sales by that person.

For example, the file for one month might contain the following:

```
25
Wang 30
Li 14
Xie 40
Zhang 39
```

What will the following processFile method print if it is called on the example file shown above?

```
public void processFile (String fileName){
    try{
        File file = new File(fileName);
        Scanner scan = new Scanner (file);
        int t = scan.nextInt ();
        int c = 0;
        while (scan.hasNext()){
            String name = scan.next();
            int num = scan.nextInt();
            if (num >= t) {
                Ul. println (name);
            }
            else {
                c = c+1;
            }
        }
        Ul. println ("count: " + c);
        scan.close ();
    } catch(IOException e){Ul. println ("Fail: " + e);}
}
```

```
Wang
Xie
Zhang
count: 1
```

**Question 5. Writing loops with input**

**[8 marks]**

Complete the following countLongWords method which reads a sentence from the user. It should

- ask the user to enter a sentence,
- read the input one word at a time
- count the total number of words
- count the number of words with 7 or more letters
- stop when the user enters a word ending with a “.”,
- print the number of long words and the total number of words.

Note: The “.” should not be counted in the length of the last word.

Here is an example of what the method would do. The boldface words are what the user typed.

```
Enter sentence:
word: These
word: String
word: methods
word: might
word: be
word: helpful:
word: startsWith
word: endsWith
word: length.
4 of the 9 words were long.
```

**(Question 5 continued)**

```
public void countLongWords(){
    UI.println("Enter sentence:");
    int numWords = 0;
    int longWords = 0;
    while (true){
        String word = UI.askToken("word:");
        numWords++;
        if (word.endsWith(".")){
            if (word.length()>7){
                longWords++;
            }
            break;
        }
        else if (word.length() >= 7){
            longWords++;
        }
    }
    UI.println(longWords + " of the " + numWords + " were long");

// another alternative for the loop:

    while (true){
        String word = UI.askToken("word:");
        numWords++;
        int len = word.length();
        if (word.endsWith(".")) {
            len--;
        }
        if (len >= 7) {
            longWords++;
        }
        if (word.endsWith(".")) {
            break;
        }
    }
}
```



### Question 6. Writing Loops with Files

[9 marks]

The company in question 4 has other files with more detailed sale data. In these files, each line contains

- the full name of an employee (a first name and a second name),
- the number of their sales, and
- the total value of their sales.

For example, here is a file with four employees:

```
Wang Bing 30 6000.00
Li Xiaoying 14 14000.00
Xie Mengjie 40 5000.00
Zhang Mei 39 16000.00
```

Complete the `analyseSales` method so that it

- prints one line for each employee in the file, printing the first name of the employee and the average value of that employee's sales.
- at the end, prints the full name and profit for the employee with the highest profit.

The profit of an employee is the total value of the sales minus the total cost of the sales. Each sale costs 100.

For example, given the file above, `analyseSales` should print out:

```
Wang 200.0
Li 1000.0
Xie 125.0
Zhang 410.3
Highest profit: Li Xiaoying 12600.0
```

**Hint:** The average value of the sales is the total value of the sales divided by the number of sales.

(Question 6 continued)

```
public void analyseSales (String fileName){
    try {

        Scanner scan = new Scanner (new File(fileName));
        double maxProfit = -1000000000;
        String maxName = "";
        while ( scan.hasNext() ) {
            String name1 = scan.next();
            String name2 = scan.next();
            int count = scan.nextInt ();
            double value = scan.nextDouble();

            double av = value/count;
            Ul. printf ("%s %.1f%n", name, av);

            double profit = value - 100*count;
            if ( profit > maxProfit){
                maxProfit = profit ;
                maxName = name1 + " " + name2;
            }
        }
        Ul. printf ("Highest profit: %s %.1f%n", maxName, maxProfit);
        scan. close ();

    } catch(IOException e){Ul. println ("Fail: " + e);}
}
```

\*\*\*\*\*