

Exam 2
ITEC 120 – Principles of Computer Science I
Spring: 2017

I will abide by the Radford University Honor Code.

Name _____

Signature _____

On this exam, you may **NOT** use already written methods such as Character class methods, or the indexOf method.

You **MAY** use the String methods **length** and **charAt**

You **MAY** use Random class methods

IMPORTANT NOTE - Any method you are asked to write on this exam **MAY** be called from any other method on this exam.
Assume the called method is written correctly.

(4 pts) What is the output of the following code?

```
String str = "abcd";  
System.out.println(str.charAt(3)); // 1pt  
System.out.println(str.length()); // 1pt
```

```
int[] numArray = { 5, 10, 15, 20, 25 };  
System.out.println(numArray[4]); // 1pt  
System.out.println(numArray.length) // 1pt
```

(12 pts) What is the output of the following code?

```
int xx = 0;
int yy = 0;
int zz = 0;

int lcv = 1;

while ( lcv <= 6 )
{
    xx = xx + 1;

    yy += lcv;

    if ( lcv%3 == 0 )
    {
        zz = zz + lcv;
    }

    lcv++;
}
System.out.println("xx: " + xx);           // 3pts each
System.out.println("yy: " + yy);
System.out.println("zz: " + zz);
System.out.println("lcv: " + lcv);
```

(15 pts) Given this method definition:

```
public String doSomething(char c, String s)
{
    String result="";
    int pos1 = 0;
    while(pos1<s.length() && s.charAt(pos1) == c)
    {
        pos1++;
    }
    int pos2 = s.length()-1;
    while(pos2>=0 && s.charAt(pos2) == c)
    {
        pos2--;
    }
    for(int i=pos1; i<=pos2; i++)
    {
        result += s.charAt(i);
    }
    return result;
}
```

What is the output each time the method is called below?

```
String result = doSomething('a', "");
System.out.println("<" + result + ">");
```

```
result = doSomething('x', "xxxABCxxx");
System.out.println("<" + result + ">");
```

```
result = doSomething('$', "$4.98");
System.out.println("<" + result + ">");
```

```
result = doSomething('*', "****");
System.out.println("<" + result + ">");
```

```
result = doSomething('-', "---7-9-2--");
System.out.println("<" + result + ">");
```

(15 pts) Given this method definition:

```
public boolean isSomething(int[] arr)
{
    boolean result = true;

    if (arr.length==0 || arr.length==1)
    {
        result = false;
    }

    int pos=1;
    while ( pos<arr.length && result )
    {
        if ( arr[pos] < arr[pos-1])
        {
            result = false;
        }
        pos++;
    }
    return result;
}
```

What is the output each time the method is called below?

```
int[] arr = {};
boolean answer = srv.isSomething(arr);
System.out.println(answer);
```

```
arr = new int[] {4};
answer = srv.isSomething(arr);
System.out.println(answer);
```

```
arr = new int[] {3,5,6,8};
answer = srv.isSomething(arr);
System.out.println(answer);
```

```
arr = new int[] {2,2,3,4,4,5,7,8,12};
answer = srv.isSomething(arr);
System.out.println(answer);
```

```
arr = new int[] {4,5,7,8,11,3,4,2};
answer = srv.isSomething(arr);
System.out.println(answer);
```

(8 pts) Write a method named `printTriangle` that takes an `int` and **prints** a triangle of asterisks. (If `printTriangle` is passed a negative number or 0, it does nothing.)

```
printTriangle(3) prints *
                    **
                    ***

printTriangle(1) prints *

printTriangle(4) prints *
                    **
                    ***
                    ****
```

(6 pts) Write a method named `leftPad` that takes a `String` and an `int` as parameters, and returns a `String` comprised of `int` number of spaces in front of the given `String`.

```
leftPad("abc",0)    --> "abc"
leftPad("abc",3)    --> "  abc"
leftPad("",2)       --> "  "
leftPad("45",2)     --> "  45"
leftPad("$",-6)     --> "$"
```

(6 pts) Write a method named `genRandomNum` that takes two `ints` and returns a randomly generated `int` between and including the two given `ints`. (i.e., the order of the parameters doesn't matter.)

```
genRandomNum(2,5) has an equal probability of returning a 2, 3, 4, or 5
genRandomNum(5,2) has an equal probability of returning a 2, 3, 4, or 5
```

- (4 pts)** Write a method `isDigit` that takes a char and answers if the char is a digit.
(No credit will be given for a solution that calls any method including `indexOf` or `Character.isDigit`.
Full credit will not be given for a solution that enumerates each digit.)

```
isDigit('5') --> true
isDigit('0') --> true
isDigit('p') --> false
isDigit('#') --> false
```

- (6 pts)** Write a method `countDigits` that returns the number of characters that are digits in the given String. You may assume your `isDigit` method above works correctly.

```
countDigits("") --> 0
countDigits("dog") --> 0
countDigits("545") --> 3
countDigits("2cool4u") --> 2
countDigits("$4,325") --> 4
```


(8 pts) Write a method `allDigits` that answers if every character in the given String is a digit.

```
allDigits("")      --> false
allDigits("abc")  --> false
allDigits("453")  --> true
allDigits("$4,567") --> false
```

(6 pts) Write code for a pass fail test case for the `allDigits` method testing this case:

```
allDigits("$4,567") --> false
```

(Don't write a main method or class. Assume that's been done and just write code for one test case.)

(8 pts) Write a method `lastDay` that takes an int representing a month and returns an int representing the last day of that month. Assume the month of February has 28 days (we are ignoring the leap year problem.) The method returns 0 for any invalid month value.

Here's a chart showing the last day of each month:

1) JAN 31	2) FEB 28	3) MAR 31	4) APR 30	5) MAY 31	6) JUN 30
7) JUL 31	8) AUG 31	9) SEP 30	10)OCT 31	11)NOV 30	12)DEC 31

```
lastDay(2) --> 28
lastDay(1) --> 31
lastDay(11) --> 30
lastDay(13) --> 0
```

(8 pts) Payday is on the 15th and last day of every month.
Write a method named `isPayday` that takes an int representing a month and an int representing a day of the month and answers if the date is a payday. The method returns false if the month number or day number is invalid. Assume February always has 28 days.

```
isPayday(2,28) --> true  
isPayday(11,30) --> true  
isPayday(4,15) --> true  
isPayday(3,14) --> false  
isPayday(3,32) --> false
```