

Introduction to Computer Programming, Spring Term 2018
Practice Assignment 3

Discussion: 3.3.2018 - 8.3.2018

Exercise 3-1 ATM
To be discussed in the tutorials

Write a program that asks a user to enter his/her pincode to the ATM machine. The pincode is generated randomly once and user is notified with this pincode. The ATM machine then asks the user to enter the pincode and the user has maximum 3 trials to enter. After that, a corresponding message should be shown to the user.

The output should be something like:

```
Please enter your pincode
5643
```

```
Please enter your pincode
5645
```

```
Please enter your pincode
5465
```

```
You exceeded your trials. We will lock your account!
```

Solution:

```
import java.util.*;
public class ATM {
    public static void main (String [] args)
    {
        Scanner sc = new Scanner (System.in);
        Random r = new Random();
        // get random number between 1000-10000
        int pass = r.nextInt(10000-1000)+1000;
        int n;
        int i = 0;
        do{
            System.out.println ("Please_enter_your_pincode");
            n = sc.nextInt();
            System.out.print ("");
            i++;
        } while (pass!=n && i<3);

        if (pass == n)
            System.out.println ("Correct_pincode!");
        else
            System.out.println ("You_exceeded_your_trials_We_will_lock_your_account!");
    }
}
```

Exercise 3-2 Adder
To be discussed in the tutorials

Write a program that adds up integers that the user enters. First the program asks how many numbers will be added up. Then the program prompts the user for each number. Finally, it prints the sum.

The output should be something like:

How many integers will be added:

```
5
Enter integer 1:
3
Enter integer 2:
4
Enter integer 3:
-4
Enter integer 4:
-3
Enter integer 5:
7
```

The sum is 7

Solution:

```
import java.util.*;
public class Adding {
    public static void main (String [] args)
    {
        Scanner sc = new Scanner (System.in);

        System.out.println ("How many integers will be added?");
        int n = sc.nextInt ();
        int i, x, sum = 0;

        for (i = 0; i < n ; i++)
        {
            System.out.println ("Enter integer "+(i+1));
            x = sc.nextInt ();
            sum += x;
        }
        System.out.println ("The sum=" +sum);
    }
}
```

Exercise 3-3 Euclidean Algorithm
To be discussed in the tutorials

The Euclidean algorithm determines the greatest common divisor (GCD) of two positive numbers by repeatedly replacing the larger number with the result of subtracting the smaller one from it until the two numbers are equal.

Write a Java program for Euclidean algorithm where the user has to enter the two numbers and the program should calculate their greatest common divisor. The output should be something like:

```
Please, enter a first number:
45
Please, enter a second number:
22
```

The GCD of 45 and 22 is 1

Solution:

```
import java.util.*;
public class Euclidian {
    public static void main(String [] args)
    {
        Scanner sc = new Scanner (System.in);
        System.out.println ("Please ,_enter_a_number:");
        int num1 = sc.nextInt ();
        System.out.println ("Please ,_enter_a_second_number:");
        int num2 = sc.nextInt ();

        int num1Saved = num1;
        int num2Saved = num2;

        while (num1 != num2)
        {
            if (num1 > num2)
                num1 -= num2;
            else
                num2 -= num1;
        }
        System.out.println ("The_GCD_of_" + num1Saved + "_and_" + num2Saved +
            "_is_" + num1);
    }
}
```

Exercise 3-4 Caesar Cipher
To be discussed in the tutorials

Write a Java program which takes two input variables message of data type String and key of data type int. The program should shift each character in message with a distance of key. For example: if key=3 then a will be replaced by d and b will be replaced by e and so on.

Hint: You can use the following method

- charAt(int index): Returns the character at the specified index. The first character of the sequence is at index 0, the next at index 1 and so on.

```
String s = "Hello";
char c = s.charAt(0);
```

The value of c is 'H'.

The output should be something like this:

```
Please enter the Message:
Hat
Please Enter the Key:
3
The encrypted word is:
Kdw
```

Solution:

```

import java.util.*;
public class Caesar {
    public static void main(String [] args)
    {
        Scanner sc = new Scanner (System.in);
        System.out.println("Please Enter a Word to be Encrypted:");
        String s = sc.nextLine();
        System.out.println("Please Enter a Key:");
        int key = sc.nextInt();

        char x;
        int l = s.length();

        int ascii = 0;

        for (int i = 0;i<l;i++)
        {
            ascii = s.charAt(i) + key;
            if((ascii >122 && s.charAt(i)>=97 && s.charAt(i)<=122)
                || (ascii >90 && s.charAt(i)>=65 && s.charAt(i)<=90))
                ascii -= 26;
            x = (char) ascii;
            System.out.print(x);

        }
        System.out.print("\n");
    }
}

```

Exercise 3-5 String Manipulation
To be discussed in the labs

Write a program that determines the number of consonants, vowels, punctuation characters, and spaces in an input line. Read in the line into a String (in the usual way). Now use the `charAt()` method in a loop to access the characters one by one. Use a switch statement to increment the appropriate variables based on the current character. After processing the line, print out the results.

Solution:

```

import java.util.*;
public class Chars
{
    public static void main (String [] args)
    {
        Scanner sc = new Scanner (System.in);
        int consonant = 0, vowel = 0, punctuation = 0, space = 0, digit = 0;
        String str, lowered;
        System.out.print("Enter a string:");
        str = sc.nextLine();

        //Convert the letter of the String 'str' to lower case.
        lowered = str.toLowerCase();

        for (int i = 0; i<lowered.length(); i++){

            switch(lowered.charAt(i)){
                case ' ': space++; break;

```

```

        case 'a':
        case 'e':
        case 'i':
        case 'o':
        case 'u': vowel++; break;
        case 'b':
        case 'c':
        case 'd':
        case 'f':
        case 'g':
        case 'h':
        case 'j':
        case 'k':
        case 'l':
        case 'm':
        case 'n':
        case 'p':
        case 'q':
        case 'r':
        case 's':
        case 't':
        case 'v':
        case 'w':
        case 'x':
        case 'y':
        case 'z': consonant++; break;
        case '0':
        case '1':
        case '2':
        case '3':
        case '4':
        case '5':
        case '6':
        case '7':
        case '8':
        case '9': digit++; break;
        default: punctuation++;
    }
}
System.out.println ("Number_of_□consonants:□" + consonant);
System.out.println ("Number_of_□vowels:□" + vowel);
System.out.println ("Number_of_□digits:□" + digit);
System.out.println ("Number_of_□punctuations:□" + punctuation);
System.out.println ("Number_of_□spaces:□" + space);
}
}

```

Exercise 3-6 Fixed Length

Write a program that asks the user to enter two words. The program then prints out both words on one line. The words will be separated by enough dots so that the total line length is 30. We can use it to make an index for a book. The user enters the name of the chapters/sections and the page number and the program generate the index. You can only print one dot at a time.

```

Enter first word:
Chapter 5

```

Enter second word:
153

Chapter 5.....153

Solution:

```
import java.util.*;
public class Word {
    public static void main (String [] args)
    {
        Scanner sc = new Scanner (System.in);
        String word1, word2, line = "";
        int dots;

        System.out.print ("Enter first word:");
        word1 = sc.nextLine();
        System.out.print ("Enter second word:");
        word2 = sc.nextLine();

        dots = 30 - (word1.length() + word2.length());
        line = line.concat(word1);

        for (int i = 0; i<dots; i++)
            line = line.concat(".");

        line = line.concat(word2);
        System.out.println (line);
    }
}
```

Exercise 3-7 Stream of Numbers

Write a Java program to read a list of nonnegative integers and outputs the maximum integer, the minimum integer, and the average of all the integers. The end of the input is indicated by the user entering a negative number. Note that the negative number is not used in finding the maximum, minimum, or average. The output should be something like this:

```
Please enter a sequence of positive numbers
2
3
5
4
-1
The maximum number is : 5
The minimum number is: 2
The average is: 3.5
```

Use in one program a while loop and in another program a do while loop.

Solution:

- Using While

```
import java.util.*;
public class NumbersWhile {
    public static void main(String [] args)
```

```

{
Scanner sc = new Scanner (System.in);
System.out.println("Please_enter_the_number");
int num = sc.nextInt();
if(num<0)
System.out.println("No_positive_Numbers_entered");
else {
int small, large;
small = num;
large = num;
double sum = 0;
double avg;
int count = 0;

while (num>=0)
{
if(num<small)
small = num;
else if (num>large)
large = num;

sum += num;
count++;
System.out.println("Please_enter_another_number:");
num = sc.nextInt();
}
avg = sum/count;
System.out.println("The_average_of_the_numbers_is_" + avg);
System.out.println("The_smallest_integer_you_entered_is_" + small);
System.out.println("The_largest_integer_you_entered_is_" + large);
}
}
}
}

```

- Using Do While

```

import java.util.*;
class NumbersDoWhile {
public static void main (String [] args)
{
Scanner sc = new Scanner (System.in);
System.out.println("Please_Enter_a_sequence_of_positive_numbers:");
int i;
//Getting the first number and assigning it to min and max
i = sc.nextInt();
int max = i, min = i, sum = 0, count = 0;
if(i<0)
System.out.println("No_positive_numbers_entered");
else
{
do{
if(max<i)
max = i;
if(min>i)
min = i;
sum += i;
count++;
}
}
}
}

```

```

        i = sc.nextInt();
    } while(i>0);

    double avg = (double) sum / count;
    System.out.println("The_maximum_number_is_" + max);
    System.out.println("The_minimum_number_is_" + min);
    System.out.println("The_average_is_" + avg);
}
}
}

```

Exercise 3-8 Triangle N
To be discussed in the labs

Write a Java program to construct a triangle shape of numbers given that n is an input from the user. For example if n=6, the shape should look like the following:

```

1
12
123
1234
12345
123456

```

Solve using a single loop only.

Solution:

```

import java.util.*;
public class Triangle {
    public static void main(String [] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Please_enter_the_number");
        int n = sc.nextInt();

        int i,j;
        String s = "";

        for(i = 1; i <= n; i++){
            s += i;
            System.out.println(s);
        }
    }
}

```