# HomePage - Biruni UniversityDepartment of Computer Engineering

BM101 Programming Language - I

2021-2022 Fall FINAL

|  |  |
| --- | --- |
| Student Name |  |
| Student ID |  |

1) VARIABLE AND VALUES **(12 Points)**

Given the declaration int x = 9, what is the **value** and **type** of the following expressions?

1. x / 3.0 / 6 \* 2 =
2. 14 % (x / 5 + 1) =
3. --x + ++x =
4. --x == x++ =

2) CONDITIONAL STATEMENTS **(13 Points)**

Rewrite the following code segment using if-else statements. Assume that grade has been declared as a type char. (Be careful about **break** statement)

switch (grade) {

case 'A':

System.out.println("Excellent");

case 'B':

System.out.println("Good");

case 'C':

System.out.println("OK");

default:

System.out.println("Let's talk");

}

3) ARRAYS-LOOPS **(25 Points)**

a) Define an integer array   
(variable name is “list”)

b) Create this array which can  
 contain 20 elements

c) write a program part (not whole   
program) which reads 20 values   
from the keyboard  
*(use “for-loop” for this)*

d) Find the average of this array in a double-type **“average”** variable *(use “while-loop” for this)*

e) Display the numbers which are less than this average value. (use the previous calculated **average** variable*(use “do-while-loop” for this)*

4) METHODS and STRINGS **(20 Points)**

Write a **COUNTCHAR** method which gets a **String** and a **char** as parameters and counts the number of this character in the String as following use.

int x= COUNTCHAR(“abracadabra”,’a’); // assigns the value as 5

5) PROGAM WRITING **(30 Points)**

Write a Java program which do the following actions

* **Holds** a 4-digit number whose all digits are different, and first digit is not equal “0” (you should do this in a correct loop statement by using **Math.random(),** which generates a double value between 0-1).

For example

* + 0123 – 1123 - 1009 – 8458 – 6550 are ILLEGAL values
  + 1942 -9436 – 9035 – 1029 are LEGAL values for it
* **Reads** a four-digit value from the Keyboard (There is no additional control for it, the user is expected to enter a valid 4-digit value)
* **Displays** the correct located digits counts and incorrect located digits count to the screen. (For example, the hold value is **7026**

If user enter “1234” the output is

correct located = 0

incorrect located = 1

If user enter “1024” the output is

correct located = 2

incorrect located = 0

If user enter “1027” the output is

correct located = 2

incorrect located = 1)

* This reading and comparison should be executed in a do-while loop statement until the user find the hold value.