***CSC 3020***

***Java Programming***

**Assignment 05**

**40 points**

**Due 11/01/2023 (11:45 A.M.)**

Assignment Objectives:

* To use the Java library classes **Date** and **Random**.
* To create objects for primitive values using the wrapper classes (**Byte**, **Short**, **Integer**, **Long**, **Float**, **Double**, **Character**, and **Boolean**).
* To simplify programming using automatic conversion between primitive types and wrapper class types.
* To use the **BigInteger** and **BigDecimal** classes for computing very large numbers with arbitrary precisions.
* To use the **String** class to process immutable strings.
* Use regular expressions to validate String data entered into an application.
* To use the **StringBuilder** class to process mutable strings.

**Solution to this assignment will not be posted on Canvas; however, any question can be discussed in the class upon request of a student.**

All assignments must be submitted by the Canvas. **No email or hard copy** is accepted. You must follow the following format:

1. For non-programming questions, use a word file to type your answers. Don’t use the text box on the Canvas to answer the questions or to write comments, we will not read it.
2. State your answer clearly.
3. For programming questions, include only the source file for each problem.
4. Submit your file to the Canvas. You must submit your assignment on time; otherwise, you will receive zero. In addition, you cannot submit your file more than one time.
5. There will be several folders on the Canvas. You need to upload your file(s) using the correct folder on the Canvas.
6. Name each file: “Assignment Number(Question number(s))”.
7. To upload your file(s):

* In Course Navigation, click the Assignments link.
* Click the title of the assignment.
* Click the **Submit** Assignment button.
* Add **File**. ...
* Add Another **File**. ...
* **Submit** Assignment. ...
* View **Submission**.

**It is your responsibility to make sure that each file is uploaded correctly. If you uploaded a wrong file, you receive zero; files will not be accepted after due date even if you have a prove that the file is created before the due date.**

**Make sure you review the Cheating & Plagiarism policy on Canvas.**

**Answer questions 1 to 5 in a separate single .txt file; include only your answers with questions numbers. Write a program for each of Q.6 - Q.7; save each program in a .txt file. Submit total of 3 .txt files by the due date.**

**Q01. (10 points - 1 point each)**

1. Which packages contain the classes Date, Random, System, and Math?

Java.util Package: Date, Random  
Java.lang Package: System, Math

1. How do you convert an integer into a string? How do you convert a numeric string into an integer? How do you convert a double number into a string? How do you convert a numeric string into a double value?

Integer into a string ->Integer.toString(int)  
Numeric string to integer -> Integer.parseInt(string)  
Double to String -> Double.toString(double)

String to double -> Double.parseDouble(string)

1. What is the output of the following code?

**public** **class** Test {

**public** **static** **void** main(String[] args) {

java.math.BigInteger x = **new** java.math.BigInteger("7");

java.math.BigInteger y = **new** java.math.BigInteger("4");

java.math.BigInteger z = x.add(y);

System.out.println("x is " + x);

System.out.println("y is " + y);

System.out.println("z is " + z);

}

}

The output is:

x is 7

y is 4

z is 11

1. To create the string Welcome to Java, you may use a statement like this:

String s = "Welcome to Java";

or:

String s = **new** String("Welcome to Java");

Which one is better? Why?   
String s = “Welcome to Java”; is better because it is shorter, therefore faster to write, and faster to read.

1. Does any method in the String class change the contents of the string?

No, these methods return new string objects.

1. Suppose string s is created using new String(); what is s.length()?

0

1. What is the difference between StringBuilder and StringBuffer?

StringBuilder runs faster, but can lead to issues with more complicated programs.

1. Write three statements to reverse a string s using the reverse method in the StringBuilder class.

StringBuilder reverser =new StringBuilder(“reverseme”);

Reverser.reverse();

String reversed = reverser.toString();

1. Write three statements to delete a substring from a string s of 20 characters, starting at index 6 and ending with index 10. Use the delete method in the StringBuilder class.  
   Stringbuilder substringer = new StringBuilder(“12345678912345678900”); //you said 20 characters

Substringer.delete(6,11); //haha delete stuff

String outputtouse = substringer.toString();

1. Show the output of the following program:

**public** **class** Test {

**public** **static** **void** main(String[] args) {

String s = "Java";

StringBuilder builder = **new** StringBuilder(s);

change(s, builder);

System.out.println(s);

System.out.println(builder);

}

**private** **static** **void** change(String s, StringBuilder builder) {

s = s + " and HTML";

builder.append(" and HTML");

}

}  
  
The output of the above code is  
Java

Java and HTML

**Q02. (3 points)**

What are autoboxing and autounboxing? Are the following statements correct?  
 Autoboxing and Autounboxing convert data to wrapper class objects when needed, it gets rid of explicitly declaring the wrapper class – which is faster to write. Autounboxing does the opposite of boxing and convertes wrapper class objects to the primitive data type needed.

a. Integer x = 3 + **new** Integer(5); //correct

b. Integer x = 3; //correct

c. Double x = 3; //incorrect

d. Double x = 3.0; //correct

e. **int** x = **new** Integer(3); //correct

f. **int** x = **new** Integer(3) + **new** Integer(4); //correct

**Q03. (5 points)**

Suppose that s1, s2, s3, and s4 are four strings, given as follows:

String s1 = "Welcome to Java";

String s2 = s1;

String s3 = **new** String("Welcome to Java");

String s4 = "Welcome to Java";

What are the results of the following expressions? Assume that the statements are independent.

a. s1 == s2 //true

b. s1 == s3 //false

c. s1 == s4 //true

d. s1.equals(s3) //true

e. s1.equals(s4) //true

f. "Welcome to Java".replace("Java", "HTML") //Welcome to HTML

g. s1.replace('o', 'V') //WelcVme tV Java

h. s1.replaceAll("o", "VT") //WelcVTme tVT Java

i. s1.replaceFirst("o", "T") //WelcTme to Java

j. s1.toCharArray() //Welcome to Java

**Q04. (6 points)**

Suppose that s1 and s2 are given as follows:

StringBuilder s1 = **new** StringBuilder("Java");

StringBuilder s2 = **new** StringBuilder("HTML");

Show the value of s1 after each of the following statements. Assume that the statements are independent.

a. s1.append(" is fun"); //Java is fun

b. s1.append(s2); //JavaHTML

c. s1.insert(3, "is fun"); //Javisfuna

d. s1.insert(2, s2); //JaHTMLva

e. s1.charAt(2); //Java

f. s1.length(); //Java

g. s1.deleteCharAt(3); //Jav

h. s1.delete(1, 3); //Ja

i. s1.reverse(); //avaJ

j. s1.replace(1, 3, "Computer"); //JComputera

k. s1.substring(1, 3); //Java

l. s1.substring(1); //Java

**Q05. (4 points)**

Which pattern match the following regular expressions.

* ab\*c //matches strings that start with a followed by zero or more b letters, and then end with c.

* ab+c //matches strings that start with a, followed by one or more b characters and end with c.

* ab?c //matches strings that start with a, followed by zero or ONE b character, and end with c

.

* ab{1,3}c //matches strings that start with a, followed by b if b is repeated between 1 and 3 times, and ends with c.

* red|blue //matches either red or blue.

* [0-9] //matches any single digit 0 to 9.

* [a-zA-Z0-9] //matches any alphanumeric character.

* Which regex pattern should be used to match a two-digit year or a four-digit year?  
  \d{2} | \d{4} //this pattern matches either 2 or four digit numbers.

**Programming Questions**

**Q06. (5 points)**

Find the first 25 numbers greater than Long.MAX\_VALUE that are divisible by 3 and 5.

Print 5 numbers in each line.

A picture containing text

Description automatically generated

**Q07. (6 points)**

The **StringBuilder** class is provided in the Java library. Provide your own implementation (name the new class **MyStringBuilder2**)with three data fields, array of characters, size, capacity, and the following methods:

* **public** MyStringBuilder2(String s); constructor (1 points) abcxyz
* **public** MyStringBuilder2 append(MyStringBuilder2 s); a method that takes s object and add s array of characters to *this* array of characters, then returns this. If appending s array of characters exceeds this array capacity, increase the capacity of this array. (3 points)
* **public** MyStringBuilder2 substring(**int** begin, **int** end); (2 points) Welcome to Java
* Accessors methods that return the size, the capacity, and the array of characters as a string. (1 points)

Do not use any method from StringBuilder, StringBuffer, or String classes (you can use length or charAt from class String).