



COMSATS UNIVERSITY ISLAMABAD (CUI)
DEPARTMENT OF COMPUTER SCIENCE
TERMINAL EXAMINATION SPRING - 2024

Course: CSC103-Programming Fundamentals
Maximum Marks: 50
Instructor: Dr. Manzoor Ilahi Tamimy

Dated: 08-JUN-2024
Time Allowed: 180 Minutes
Class: BAI

Instructions:

- All questions are self-explanatory and require no further explanations during exam time.
- Make sure that you have signed the attendance sheet before leaving the examination room.
- Return the question paper along with the answer sheet.

[CLO-1: Demonstrate the fundamental concepts of programming.]

Question-1: Write the output of the following code snippets.

[Marks: 4]

i) public class A { public static void main (String[] args){ char c1 = 'D'; char c2 = 'A'; int AVSum = c1 + c2; System.out.printf("Character 1: %c, ASCII Value: %d\n", c1, (int) c1); System.out.printf("Character 2: %c, ASCII Value: %d\n", c2, (int) c2); System.out.printf("Sum of Characters as ASCII Value: %d\n", AVSum); } }	ii) public class A { public static void main(String[] args) { double a=10.5; int b=5; int c=6; System.out.println(a+b/b*c-b); } }
iii) public class A { public static void main (String[] args) { int x = 10; int y = 5; int result = x % (x - y); System.out.println("Result: " + result); } }	iv) public class J { public static void main(String[] args) { double a = 2.0; double b = 7.0; double c = ++a * b++ + ++b; System.out.println(a); System.out.println(b); System.out.println(c--); } }

[CLO-2: Employee programming constructs using a programming language.]

Question-2:

[Marks: 10]

Suppose we want to compute the amount of money in a bank account with compound interest. If the amount of money in the account is m , the amount in the account at the end of the month will be $1.005m$. Write a recursive method that will compute the amount of money in an account after t months with a starting amount of m . Also show the call stack of the execution of this function.

Question-3:

[Marks: 5]

Suppose you are required to develop an attendance system for a teacher who manages five different courses. In each course, a different number of students are registered. Write a JAVA program that uses ragged array to store the attendance('P':Present and 'A': Absent) of each course and then display the number of present and absent students in each course.

Question-4:

[Marks: 6]

Write a method that takes two parameters: an input string and a list of spam words. The method should return a new string where all instances of the spam words have been removed. The removal should be case-insensitive.

Sample Run:

String input: "This is a test string with some spam words like spam and junk."

Output: "This is a test string with some words like and."

Suppose 'spam' and 'junk' are the spam words in the above case.

Question-5:

[Marks: 10]

Write a JAVA program that reads the password from the users of a system and stores it in a 2D matrix in encrypted form. There are fifty users of the system, and the password length is exactly eight characters. For instance, if a user enters "fish175#" then it will be stored as "gjt286\$" that means each character has been replaced with its next character in ASCII series.

Question-6:

[Marks: 5]

Write a JAVA program that utilize following methods.

- fillArray(): Uses to initialize 2D array with random values.
- swapColumn(): This method will return the array such that it swaps the column having maximum sum with the column having minimum sum.
- printArray(): This method will show the output after swapping the columns.

[CLO-3: handle programs utilizing exception and file I/O.]

Question-7:

[Marks: 10]

Write a program to grade a multiple-choice exam. The exam has 20 questions, each answered with an alphabet in the range of 'a' to 'f'. The data is stored on a file (exams.txt) where each line contains the student id, and answer string consisting of 20 characters, see exams.txt. The correct answer string is abcdefaaadefefbcdefab.

What your program should do: Read the answer string from exams.txt, for each student. Calculate correct answers from each student. Create a file scores.txt and write student ID number and score to this file. Your program must use proper Exception handling where required.

