King Abdulaziz University Faculty of Computing and Information
Technology Computer Science Department

CPCS204, Spring 2023
Assignment 2: Recursion
Assigned: Thursday April 13, 2023,   Due: Thursday May 04, 2023

Purpose

1. Apply recursion to solve simple problems.
2. Comprehend recursion concepts.

ABET Student Outcome (SO-2):
Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.

Read Carefully:
This assignment is worth **5%** of your final grade.

NOTE:
This is an individual assignment; you must solve it by yourself. Any form of cheating will result in receiving zero in the assignment.

The deadline for this project is **Thursday May 04, 2023 by 11:59 PM.**

LATE SUBMISSION: No assignment will be accepted after the deadline.

Blackboard Submission:

This project must be submitted online via Blackboard.

The source file(s) of your program should be zipped up. You must name the zip file using the following naming convention: **SectionNumber_StudentID_ProgramNumber**.zip.
Example: EA_1110348_P2.zip

Question:	1	2	3	Total
Points:	20	20	60	**100**

Question 1: (20 points)

Consider a **recursive** method that prints a large pyramid made of small stars ("*"). The size of the pyramid will be determined from the maximum base width of the pyramid, and is guaranteed to be odd integer. For example, the print a pyramid with maximum base width = 9 as follows:

```
        *
      *   *
     *  *  *
    *  *  *  *
   *  *  *  *  *
  *  *  *  *  *  *
 *  *  *  *  *  *  *
*  *  *  *  *  *  *  *
*  *  *  *  *  *  *  *  *
```

1- Write a pseudocode/algorithm that accepts an integer number prints a large pyramid as explained above.   **(15 points)**

   Note: write the information that you need to describe the header of the method

2-   Trace your algorithm **for length=5**.   **(5 points)**

Question 2:   (20 Points, SO-2)
Consider the following recursive method and answer the questions:

```
1 public class NewClass {
2
3    public static void main(String[] args) {
4          int[][] st =new int[5][5];
5          changeAndPrint(st,0,st.length-1);
6       }
7
8     public static void changeAndPrint(int[][] st, int i, int j) {
9         if (i < st.length) {
10             temp= st[i][i]
               st[i][i] = st[i][j];
11             st[i][j] = temp;
12             for (int k = 0; k < st.length; k++) {
13                 System.out.print(st[i][k]);
14             }
15             System.out.println("");
16             changeAndPrint(st, ++i, --j);
17          }
18       }
19 }
```

1- Trace the above code. Note: you must write your tracing using recursion trace as described in the course lectures. You must also show the final answer that is returned. **(15 points)**

2- What does the recursive method print?   **(5 points)**

Question 3: (60 Points, SO-2)
Write a Java program that reads a set of commands from the input file and executes the relevant recursive method to get the required output, as shown in the attached input/output files. Each command in the input file is followed (on the same line of input) by the respective parameters required for executing the recursive method. **[15 points for writing the main method, 15 points for writing the recursive method for each of the below problems].**

⇒ **Problem#1.** Write a recursive method **repeatDigit** which accepts a positive integer M (greater than 0), a digit K, and number of repeated times R, and returns an integer number N, such that N is a new number with the K digit that is repeated R times.
   Consider the following output examples:
   - **Example 1:**
   Command: **repeatDigit 95 1 5**
   Output: 9511111 is the number with repeated 5 times of digit 1
   - **Example 2:**
   Command: **repeatDigit 234 5 3**
   Output: 234555 is the number with repeated 3 times of digit 5
   - **Example 3:**
   Command: **repeatDigit 444 7 4**
   Output: 4447777 is the number with repeated 4 times of digit 7

⇒ **Problem#2.** Write a recursive method called **findEvenNumbers** that accepts only one parameter which is an array of integer numbers and returns the array of even numbers in that array. Consider the following output examples:
   - **Example 1:**
   Command: **findEvenNumbers 32,4,21,-87,3,49,143,47,50**
   Output: The even array list is: 32,4,50
   - **Example 2:**
   Command: **findEvenNumbers 1,2,3,4,5,6,7,8,9,10**
   Output: The even array list is: 2,4,6,8,10

⇒ **Problem 3:** You must write a recursive method **getCommonChars** that takes as inputs two strings, **str1** and **str2**, and to *return* the common characters in the two lists.
   Note: The returned common characters should be not repeated even if it is repeated in the input strings. Consider the following output examples:

   - **Example 1:**
   Command: **displayCommonChars** abadc acbdf
   Output: The common characters are abd

- **Example 2:**
  ```
  Command: displayCommonChars Ahmad ahmad
  Output: The common characters are Ahmad
  ```

- **Example 3:**
  ```
  Command: displayCommonChars ahmad Ahmad
  Output: The common characters are ahmad
  ```

Input File Specifications

You will read in input from a file, "**input.txt".** Have this AUTOMATED. Do not ask the user to the name of the input file. You should read in this automatically. The first line of the input file will have one positive integer, representing the number of commands (lines) inside the input file.

Each of the following n lines will have a command, and each command will be followed by appropriate data as described below (and this data will be on the same line as the command).

The commands (for the 3 recursive methods), and their relevant data, are described below:

⇒ **repeatDigit**: This command will be followed by three integers a, b, and c, separated by one space, where b is a one-digit number.

⇒ **findEvenNumbers:** This command will be followed by a set of integers numbers which are separated by ','.

⇒ **displayCommonChars:** This command will be followed by two strings separated by one space.

Output Format

Your program must output to a file, called **"output.txt". You must follow the program specifications exactly.** Refer to sample output file for exact formatting specifications.

Grading Details

Your program will be graded upon the following criteria:

1. Adhering to the implementation specifications listed on this write-up.
2. Your algorithmic design.
3. Correctness.
4. **Use of Recursion. If your program does not use recursion, you will get a zero.**
5. The frequency and utility of the comments in the code, as well as the use of white space for easy readability. (If your code is poorly commented and spaced and works perfectly, you could earn as low as 80-85% on it.)
6. Your program should **include a header comment with the following information: your name, ID, email, course number, section number, assignment title, and date.**
7. Your output MUST adhere to the EXACT output format shown in the sample output file.

Deliverables

You should submit a folder with **TWO** files inside:

1- You must name the java file using the following naming convention:

Example: *EA_ 1110348_ P2.**java***

2- .docx or .pdf file containing solution of Question1 and 2.