**Assignment #4 (5%)**

Submission deadline: Sunday, **April 3, 2022** **(23:59)**

**NAME: Talha Abdullah Punjabi**

**QUID: 201903446**

**Chapters:** This assignment covers Chapter 5 in the textbook

**Important Notes (must read):**

1. When submitting your work, you must use Blackboard, **NO other means like email submissions, are accepted.**
2. Assignments are to be solved **individually**.
3. A mark of zero (0) will be awarded for the whole assessment in which plagiarism was found to occur. Even if a single question is plagiarized**, the whole assignment will get zero** (0).
4. Submit your work as instructed below ***before*** the deadline**. No extension will be provided.**
5. Along with the MS Word submission file, you must submit separate Java files for the programs. Put all these files in a folder named **Assignment4\_QUID**. Compress this folder and submit it.
6. In the Word document, make sure that to add screenshots for input and output of your programs.

If you have any questions or doubts about any of the above-mentioned issues, please consult Eng. Alaa Hussein [alaa.hussein@qu.edu.qa](mailto:alaa.hussein@qu.edu.qa) . There is only 1 question in this assignment.

**Q1. (Section 5.4.)**

Backtracking is used to solve problems in which a sequence of objects is selected from a specified set so that the sequence satisfies some criterion. In this assignment, you are expected to apply this Backtracking algorithm for the Sum-of-Subsets problem, which is explained as follows:

* **Problem:** Given n positive integers (weights) and a positive integer W, determine all combinations of the integers that sum to W.
* **Inputs:** Positive integer n, sorted (non-decreasing order) array of positive integers w indexed from 1 to n, and a positive integer W.
* **Outputs:** all combinations of the integers that sum to W.

Write the **Java program** of the Backtracking algorithm for the Sum-of-Subsets problem (Algorithm 4.5.) and demonstrate its correctness with some input values.

Answer: (please write your answer here, add required space if needed)

**package** Assignment4;

**import** java.util.ArrayList;

**import** java.util.Arrays;

**import** java.util.List;

**import** java.util.Scanner;

**public** **class** SumofSubsets {

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

Scanner sc=**new** Scanner(System.***in***);

System.***out***.print("Enter the Sum :" );

**int** sum=sc.nextInt();

System.***out***.print("Enter the Number of Elements in the array : ");

**int** n=sc.nextInt();

System.***out***.println("Enter the array elements : ");

**int**[] providedArray = **new** **int**[n];

**for**(**int** i=0;i<n;i++) {

System.***out***.print(i+" = ");

**int** x=sc.nextInt();

providedArray[i]=x;

}

List<**int**[]> list=*solver*(sum,providedArray);

**if**(list.isEmpty()) {

System.***out***.println("No combinations give the provided sum");

System.*exit*(0);

}

System.***out***.println("Printing List of Subsets that lead to the sum : ");

list.forEach(array -> System.***out***.print(Arrays.*toString*(array)+"\t" ));

}

**public** **static** List<**int**[]> solver(**int** sum, **int** given[]){

List<**int**[]> Answer = **new** ArrayList<**int**[]>();

**int** k,z=0;

**int** n=given.length;

**for**(**int** i=0; i<n;i++) {

**for**(**int** j=0;j<n;j++) {

**if**(given[i]+given[j]==sum) {

**int** [] array= {given[i],given[j]};

Answer.add(array);

}

}

}

**return** Answer;

}

}

Outputs:

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, email

Description automatically generated

Text

Description automatically generated