

**CSE3032 - Competitive Programming**

**WIN SEM (2022-2023) AMR**

**Class Number: AP2022236001007**

**Slot: L11+L12+L19+L20**

**ASSIGNMENT - 5**

Last Date for Submission: Saturday (04-02-2023)

Name: Chilamakuru Sai Venkata Yaswanth

Reg No: 20BCI7076

Write the program using (C / C++ / Java / Python) to solve the following problems.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.No** | **Problem Name** | **Statement** | | | |
| 1 | Unique sum of the subsets and their total | Given an array print all the unique sum of the subsets and their total generated.  Input: arr[]= {3, 3, 1, 1, 2, 2} Output:  0,1,2,3,4,5,6,7,8,9,10,11,12  78  Actual subsets sum is: 0 1 1 2 2 2 3 3 3 3 3 3 4 4 4 4 4 4 4 5 5 5  5 5 5 5 5 6 6 6 6 6 6 6 6 6 6 7 7 7 7 7 7 7 7 8 8 8 8 8 8 8 9 9 9 9  9 9 10 10 10 11 11 12 → Unique subsets sum is → Total → 78 Input: arr[]= {3,1,2} Output:  0,1,2,3,4,5,6  21  Actual subsets sum is: 0,1,2,3,3,4,5,6 → Unique subsets sum is → 21  Input: arr[]= {11, 1, 12, 2, 13, 3} Output:  0 1 2 3 4 5 6 11 12 13 14 15 16 17 18 19 23 24 25 26 27 28 29  30 31 36 37 38 39 40 41 42  672  Actual subsets sum is: 0 1 2 3 3 4 5 6 11 12 12 13 13 13 14 14  14 14 15 15 15 15 16 16 16 16 17 17 17 18 18 19 23 24 24 25  25 25 26 26 26 26 27 27 27 27 28 28 28 28 29 29 29 30 30 31 36 37 38 39 39 40 41 42 → Unique subsets sum is → 672 | | | |
|  | **Should not use predefined functions to remove the duplicate** | |  |
| **elements.** |  |
| 2 | Unique sum of the subsets – find how many are Fibonacci primes | Input: arr[]= {3,1,2} Output: 0,1,2,3,4,5,6  3  Actual subsets sum is: 0,1,2,3,3,4,5,6 → Unique subsets sum is  → Fibonacci series elements → {2,3,5} → primes → {2,3,5}  →3  Input: arr[]= {4,5,6}  Output: 0 4 5 6 9 10 11 15 | | | |
|  |  | 1  Actual subsets sum is: 0 4 5 6 9 10 11 15 → Unique subsets sum is → Fibonacci series elements → {5} → primes → {5} →1  Input: arr[]= {2,3,5} Output: 0 2 3 5 7 8 10  3  Actual subsets sum is: 0 2 3 5 5 7 8 10→ Unique subsets sum is  → Fibonacci series elements → {2,3,5,8} → primes → {2,3,5}  →3 | | | |

**Note:**

* If Code similarity is found, assignment will not be considered and Zero (0) Marks will be awarded.
* You have to upload a single document consisting of all the above programs and corresponding Output.
* You will be asked to explain the code, run and show the same program in the respective platforms (hacker rank / hacker earth / spoj)

1) Unique sum of the subsets and their total

Code:

import java.util.Scanner;

import java.util.\*;

public class cp5{

           public static void compute(ArrayList<Integer> arr, int n, int sum,int ind, ArrayList<Integer> ar)

   {

       if (ind == n) {

          ar.add(sum);

          return;

       }

      compute(arr, n, sum + arr.get(ind),ind + 1, ar);

      compute(arr, n, sum, ind + 1, ar);

   }

   public static void main(String[]args){

       Scanner sc=new Scanner(System.in);

                       int n=sc.nextInt();

                       ArrayList<Integer> arr=new ArrayList<>();

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

                                   arr.add(sc.nextInt());

                       }

                       Collections.sort(arr);

                       ArrayList<Integer> ar=new ArrayList<>();

                       compute(arr, arr.size(), 0, 0, ar);

                       Collections.sort(ar);

                       int a[]=new int[ar.size()];

                       a[0]=ar.get(0);

                       int c=0;

                       int size=1;

                       int sum=0;

                       for(int i=0;i<ar.size();++i){

                                   c=0;

                                   for(int j=a.length-1;j>=0;--j){

                                               if(ar.get(i)!=a[j]){

                                                           c++;

                                               }

                       }

                       if(c==a.length){

                                   a[size]=ar.get(i);

                                   sum=sum+a[size];

                                   size++;

                       }

                       }

                       for(int i=0;i<size;++i){

                                   System.out.print(a[i]+" ");

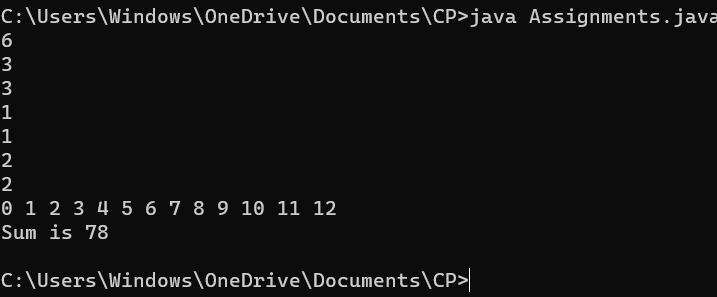
                       }

                       System.out.println();

                       System.out.println("Sum is " +sum);

   }

}



2) import java.util.ArrayList;

import java.util.HashSet;

import java.math.BigInteger;

public class UniqueSubsetSum {

public static HashSet<Integer> fibPrimes = new HashSet<>();

public static boolean isFibPrime(int n) {

if (fibPrimes.contains(n)) {

return true;

}

BigInteger f1 = BigInteger.valueOf(0);

BigInteger f2 = BigInteger.valueOf(1);

BigInteger fn = f1.add(f2);

while (fn.intValue() <= n) {

if (fn.isProbablePrime(10) && fn.intValue() == n) {

fibPrimes.add(n);

return true;

}

f1 = f2;

f2 = fn;

fn = f1.add(f2);

}

return false;

}

public static ArrayList<Integer> getSubsets(int[] set) {

ArrayList<Integer> subsets = new ArrayList<>();

int n = set.length;

int numOfSubsets = (int)Math.pow(2, n);

for (int i = 0; i < numOfSubsets; i++) {

int sum = 0;

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

if ((i & (1 << j)) > 0) {

sum += set[j];

}

}

subsets.add(sum);

}

return subsets;

}

public static void main(String[] args) {

int[] set = {1, 2, 3, 4};

ArrayList<Integer> subsets = getSubsets(set);

HashSet<Integer> uniqueSubsets = new HashSet<>(subsets);

int count = 0;

for (int num : uniqueSubsets) {

if (isFibPrime(num)) {

count++;

}

}

System.out.println("Number of Fibonacci primes: " + count);

}

}