**Maximum Sum**

Attempted by: **873**

/

Accuracy: **64%**

/

Maximum Score: **20**

/

1 Vote

Tag(s):

Algorithms, Basics of Hash Tables, Data Structures, Easy, Hash Tables,Hashing, Set

**PROBLEM**

**EDITORIAL**

**MY SUBMISSIONS**

**ANALYTICS**

Given an array A of N integers. Now, you have to output the sum of unique values of the maximum subarray sum of all the possible subarrays of the given array A.  
**Note:**Subarray means contiguous elements with atleast one element in it.

**Input Format**

The first line of the input contains a single integer N, the total number of elements in array A.  
The next line of the input contains N space-separated integers representing the elements of the array.

**Output Format**

The only single line of the output should contain a single integral value representing the answer to the problem.

**Constraints**

1≤N≤2000  
0≤|Ai|≤109

**SAMPLE INPUT**

4

5 -2 7 -3

**SAMPLE OUTPUT**

17

**Explanation**

Following are the possible number of subarrays and their respective maximum subarray sums:

[5]=[5]=5  
[5,−2]=[5]=5  
[5,−2,7]=[5,−2,7]=10  
[5,−2,7,−3]=[5,−2,7]=10  
[−2]=[−2]=−2  
[−2,7]=[7]=7  
[−2,7,−3]=[7]=7  
[7]=[7]=7  
[7,−3]=[7]=7  
[−3]=[−3]=−3

5+10+(−2)+7+(−3)=17

**Time Limit:**1.0 sec(s) for each input file.

**Memory Limit:**256 MB

**Source Limit:**1024 KB

**Marking Scheme:**Marks are awarded when all the testcases pass.

**Allowed Languages:**Bash, C, C++, C++14, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), TypeScript, Julia, Kotlin, Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Swift, Swift-4.1, Visual Basic

<https://www.hackerearth.com/practice/data-structures/hash-tables/basics-of-hash-tables/practice-problems/algorithm/maximum-subarray-sum-of-subarrays-7f33aefa/>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

namespace ConsoleApp1

{

class Program

{

static long maxSum(long[] a)

{

HashSet<long> hash = new HashSet<long>();

for (int i = 0; i < a.Length; i++)

{

long max\_so\_far = a[i];

long curr\_max = a[i];

hash.Add(max\_so\_far);

for (int j = i + 1 ; j < a.Length; j++)

{

curr\_max = Math.Max(a[j], curr\_max + a[j]);

max\_so\_far = Math.Max(max\_so\_far, curr\_max);

hash.Add(max\_so\_far);

}

}

return hash.Sum();

}

static void Main(string[] args)

{

long n = long.Parse(Console.ReadLine());

long[] arr = Array.ConvertAll(Console.ReadLine().Trim().Split(' '), e => long.Parse(e));

Console.WriteLine(maxSum(arr));

//int[] arr = { 5, -2, 7, -3 };

//Console.WriteLine(maxSum(arr));

Console.ReadLine();

}

}

}