

















Greedy Algorithms Challenges

Marc's Cakewalk **■**



Problem

Submissions

Leaderboard

Discussions

Editorial

Marc loves cupcakes, but he also likes to stay fit. He eats n cupcakes in one sitting, and each cupcake i has a calorie count, c_i . After eating a cupcake with c calories, he must walk at least $2^j \times c$ (where j is the number cupcakes he has already eaten) miles to maintain his weight.

Given the individual calorie counts for each of the n cupcakes, find and print a long integer denoting the minimum number of miles Marc must walk to maintain his weight. Note that he can eat the cupcakes in any order.

Input Format

The first line contains an integer, n, denoting the number of cupcakes.

The second line contains n space-separated integers describing the respective calorie counts of each cupcake, $c_0, c_1, \ldots, c_{n-1}$.

Constraints

- $1 \le n \le 40$
- $1 \le c_i \le 1000$

Output Format

Print a long integer denoting the minimum number of miles Marc must walk to maintain his weight.

Sample Input 0

1 3 2

Sample Output 0

11

Explanation 0

Let's say the number of miles Marc must walk to maintain his weight is miles. He can minimize miles by eating the n=3 cupcakes in the following

- 1. Eat the cupcake with $c_1 = 3$ calories, so $miles = 0 + (3 \cdot 2^0) = 3$.
- 2. Eat the cupcake with $c_2 = 2$ calories, so $miles = 3 + (2 \cdot 2^1) = 7$.
- 3. Eat the cupcake with $c_0 = 1$ calories, so $miles = 7 + (1 \cdot 2^2) = 11$.

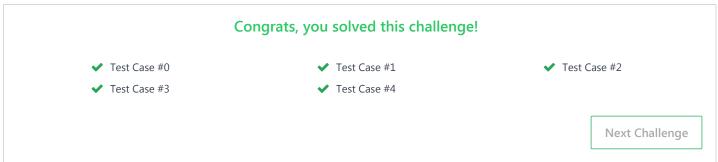
We then print the final value of *miles*, which is 11, as our answer.

⊮ in

Submissions: 1913 Max Score: 15 Difficulty: Easy



```
Current Buffer (saved locally, editable) & 49
                                                                                          C#
                                                                                                                            Ö
 1 using System;
   using System.Collections.Generic;
 3 using System.IO;
 4 using System.Ling;
 5 ▼ class Solution {
 6
 7 ▼
        static void Main(String[] args) {
             int n = int.Parse(Console.ReadLine());
 8
                 long[] a = Array.ConvertAll(Console.ReadLine().Split(' '), e => long.Parse(e));
 9
10
                 //int[] a = { 1, 3, 2 };
                 Array.Sort(a);
11
                 long miles = 0;
12
13
                 long cupcakes = 0;
14
                 for (int i = a.Length - 1; i >= 0; i--)
15 ▼
                     miles += a[i] * (long)Math.Pow(2, cupcakes++);
16
17
18
                 Console.WriteLine(miles);
19
        }
    }
20
21
                                                                                                                  Line: 18 Col: 38
                       Test against custom input
                                                                                                       Run Code
                                                                                                                     Submit Code
1 Upload Code as File
```



Copyright © 2017 HackerRank. All Rights Reserved

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Interview Prep | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature