**CP Club 365Days Challenge**

**Date – 29/11/2022**

**Programming language – Your preferred programming language**

**Problem Statement**

**<https://www.hackerrank.com/challenges/marcs-cakewalk/problem?isFullScreen=true>**

**Your Code**:

// 0x43Day of 0x365Days challenge

// VEDANT BHARAD

// 29-11-2022

#include <assert.h>

#include <ctype.h>

#include <limits.h>

#include <math.h>

#include <stdbool.h>

#include <stddef.h>

#include <stdint.h>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

char\* readline();

char\* ltrim(char\*);

char\* rtrim(char\*);

char\*\* split\_string(char\*);

int parse\_int(char\*);

int\* myFun(int\* arr,int len)

{

    for(int loop=0;loop<len;loop++){

        for(int loop2=loop;loop2<len;loop2++){

            if(arr[loop]<arr[loop2]){

                int temp=arr[loop];

                arr[loop]=arr[loop2];

                arr[loop2]=temp;

            }

        }

    }

    return arr;

}

long marcsCakewalk(int calorie\_count, int\* calorie) {

    long sum=0,val=1;

    myFun(calorie,calorie\_count);

    for(int loop=0; loop<calorie\_count; loop++) {

        sum=sum+(calorie[loop]\*val);

        val=val\*2;

    }

    return sum;

}

int main()

{

    FILE\* fptr = fopen(getenv("OUTPUT\_PATH"), "w");

    int n = parse\_int(ltrim(rtrim(readline())));

    char\*\* calorie\_temp = split\_string(rtrim(readline()));

    int\* calorie = malloc(n \* sizeof(int));

    for (int i = 0; i < n; i++) {

        int calorie\_item = parse\_int(\*(calorie\_temp + i));

        \*(calorie + i) = calorie\_item;

    }

    long result = marcsCakewalk(n, calorie);

    // fprintf(fptr, "%ld\n", result);

    printf("%ld\n", result);

    fclose(fptr);

    return 0;

}

char\* readline() {

    size\_t alloc\_length = 1024;

    size\_t data\_length = 0;

    char\* data = malloc(alloc\_length);

    while (true) {

        char\* cursor = data + data\_length;

        char\* line = fgets(cursor, alloc\_length - data\_length, stdin);

        if (!line) {

            break;

        }

        data\_length += strlen(cursor);

        if (data\_length < alloc\_length - 1 || data[data\_length - 1] == '\n') {

            break;

        }

        alloc\_length <<= 1;

        data = realloc(data, alloc\_length);

        if (!data) {

            data = '\0';

            break;

        }

    }

    if (data[data\_length - 1] == '\n') {

        data[data\_length - 1] = '\0';

        data = realloc(data, data\_length);

        if (!data) {

            data = '\0';

        }

    } else {

        data = realloc(data, data\_length + 1);

        if (!data) {

            data = '\0';

        } else {

            data[data\_length] = '\0';

        }

    }

    return data;

}

char\* ltrim(char\* str) {

    if (!str) {

        return '\0';

    }

    if (!\*str) {

        return str;

    }

    while (\*str != '\0' && isspace(\*str)) {

        str++;

    }

    return str;

}

char\* rtrim(char\* str) {

    if (!str) {

        return '\0';

    }

    if (!\*str) {

        return str;

    }

    char\* end = str + strlen(str) - 1;

    while (end >= str && isspace(\*end)) {

        end--;

    }

    \*(end + 1) = '\0';

    return str;

}

char\*\* split\_string(char\* str) {

    char\*\* splits = NULL;

    char\* token = strtok(str, " ");

    int spaces = 0;

    while (token) {

        splits = realloc(splits, sizeof(char\*) \* ++spaces);

        if (!splits) {

            return splits;

        }

        splits[spaces - 1] = token;

        token = strtok(NULL, " ");

    }

    return splits;

}

int parse\_int(char\* str) {

    char\* endptr;

    int value = strtol(str, &endptr, 10);

    if (endptr == str || \*endptr != '\0') {

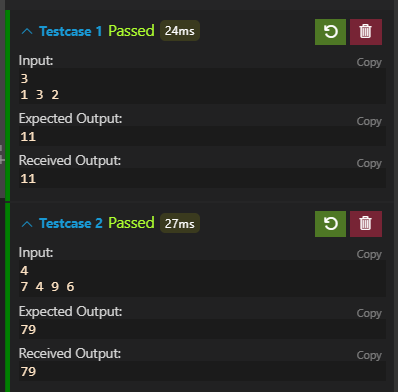
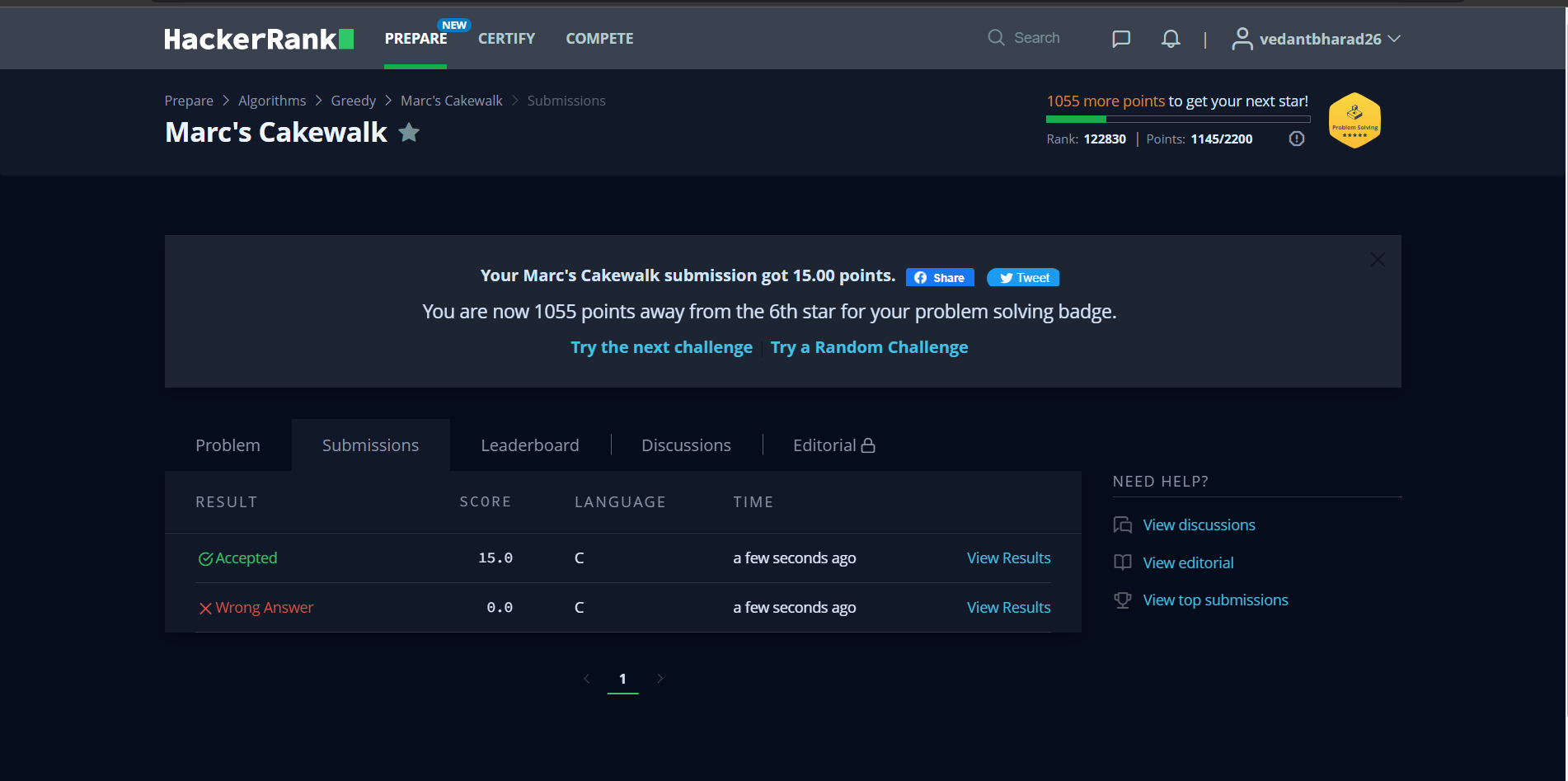
        exit(EXIT\_FAILURE);

    }

    return value;

}

**Output (Screen Shot)**:



**Understanding about problem:**

* In this task there two input first is size of array and second is array it self.
* first I need to sort array in descending order.
* In this task I need to return long int value which will be sum of array element with power of 2.

Note: If you can't understand the problem, feel free to contact us and we'll help you. Please don't copy and paste from anywhere.

ALL THE BEST

Team CP Club