

Oddeven

Descriere

Dându-se un vector cu n elemente, numere întregi, a , să se efectueze modificările necesare, astfel încat vectorul final să aibă elementele inițiale de pe pozițiile impare sortate în ordine crescătoare, iar cele de pe pozițiile pare sortate în ordine descrescătoare.

Download sursă [aici](#).

Formatul datelor de intrare

Se va citi din terminal de pe prima linie un număr natural, n , urmat pe a doua linie de un vector a , de n elemente, numere întregi, separate prin câte un spațiu.

Observație: Orice număr poate fi stocat într-un `int`, iar indexarea vectorului a începe de la 1.

Formatul datelor de ieșire

Se va afișa pe o linie vectorul modificat a , cu elementele sortate în ordinea cerută.

Observație: Elementele afișate vor fi separate prin câte un spațiu.

Note

1. Antetul funcțiilor de sortare și pasul contorului din cadrul acestora nu poate fi modificat.
2. Conținutul instrucțiunii repetitive `do while()` cadrul funcției `main()` nu poate fi modificat.
3. Nu se pot adăuga linii de cod în program, însă se pot șterge sau modifica liniile de cod existente (cu excepția celor de la restricțiile 1 și 2).

Exemple

Exemplul 1

Intrare

```
5
6 2 3 5 10
```

Ieșire

```
3 5 6 2 10
```

Exemplul 2

Intrare

6
10 7 3 2 5 6

Ieşire

3 7 5 6 10 2

Recursivitate Reciprocă

Descriere

Ștefan, după ce a citit cartea *Gödel, Escher, Bach: an Eternal Golden Braid* de Douglas Hofstadter, și-a dorit să implementeze șirurile “Figure-Figure”, ca o provocare.

Cele două șiruri sunt definite așa:

$$R(1) = 1; S(1) = 2$$

$$R(n) = R(n - 1) + S(n - 1)$$

Unde șirul $S(n)$ este șirul de numere naturale pozitive strict crescătoare care nu apar în $R(n)$. Cele două șiruri sunt complementare.

Întrucât ultima oară când a scris un program cu recursivitate mutuală a avut greșeli, de data asta a încercat să adapteze un cod de pe wiki-ul *Rosetta Code*. Totuși, ceva nu a mers conform planului...

Download sursă [aici](#).

Formatul datelor de intare

Se va citi din terminal n , un număr natural mai mic sau egal cu 1000.

Formatul datelor de ieșire

Se va afișa în terminal șirul $R(n)$, respectiv $S(n)$, în acest format, pe 2 linii:

Note

Secvența R :

1, 3, 7, 12, 18, 26, 35, 45, 56, 69, 83, 98, 114, 131, 150, 170, 191, 213, 236, 260, ...

Șirul S :

2, 4, 5, 6, 8, 9, 10, 11, 13, 14, 15, 16, 17, 19, 20, 21, 22, 23, 24, 25, ...

Exemple

Exemplul 1

Intrare

Ieşire

R(20): 1 3 7 12 18 26 35 45 56 69 83 98 114 131 150 170 191 213 236 260
S(20): 2 4 5 6 8 9 10 11 13 14 15 16 17 19 20 21 22 23 24 25

Sum Generation

Descriere

Este ziua Mariei și aceasta vrea să aducă bomboane la școală. Aceasta, dorind să le facă colegilor de clasă ziua mai dulce, având înainte un test de mate, a luat mai multe cutii de bomboane. Însă, ea nefăcând diferențe între colegii ei vrea să aducă un număr de bomboane astfel încât toți să primească același număr de bomboane, indiferent de numărul de bomboane final. Ea a încercat să își dea seama în câte moduri poate lua din cutiile de bomboane astfel încât să îi dea un număr fix de bomboane per coleg, dar nu îi dă corect numărul total de posibilități. Ajutați-o pe Măria să afle rezultatul final corect.

Download sursă [aici](#).

Formatul datelor de intare

Se va citi din terminal de pe prima linie numărul de cutii și numărul de colegi, `nr_cutii` și `nr_colegi`, și apoi pe următoarea linie o secvență de `nr_cutii` numere întregi reprezentând bomboanele fiecărei cutii.

Formatul datelor de ieșire

Se va afișa numărul total de posibilități de aducere a cutiilor la școală.

Note

- $3 < \text{nr_cutii} < 31$
- $3 < \text{nr_colegi} < 36$

Exemple

Input 1

Exemple

Exemplul 1

Intrare

```
4 23
13 10 7 16
```

Ieșire

```
3
```

Explicație:

Maria poate sa ia fie cutiile care contin 13 si 10 bomboane,

fie cutiile cu 7 si 16 bomboane, dand astfel un numar de
1 bomboana/elev, fie toate cele 4 cutii, dand 2 bomboane/elev

Exemplul 2

Intrare

8 20
14 6 14 13 5 10 20 7

Ieșire

13

Explicație:

Combinatiile posibile sunt: 20 14, 6 x2(deoarece sunt 2 cutii diferite de 14 bomboane) 13,
7 14, 6, 20 x2 13, 7, 20 14, 13, 5 x 2 14, 14, 5, 7 14, 14, 5, 7, 20 14, 6, 13, 7, 20 x2

20, 14 6 x2, 13 7, 14 6 20 x2, 13 7 20, 14 6 13 7 20 x2, 14 13 5 x 2, 14 14 5 7, 14 14 5 7 20