

AtCoder Beginner Contest 123

Problema B: *Five Dishes*

Prof. Edson Alves - UnB/FGA

AtCoder Beginner Contest 123B – Five Dishes

The restaurant AtCoder serves the following five dishes:

- ABC Don (rice bowl): takes A minutes to serve.
- ARC Curry: takes B minutes to serve.
- AGC Pasta: takes C minutes to serve.
- APC Ramen: takes D minutes to serve.
- ATC Hanbagu (hamburger patty): takes E minutes to serve.

Here, the time to serve a dish is the time between when an order is placed and when the dish is delivered.

AtCoder Beginner Contest 123B – Five Dishes

This restaurant has the following rules on orders:

- An order can only be placed at a time that is a multiple of 10 (time 0, 10, 20, ...).
- Only one dish can be ordered at a time.
- No new order can be placed when an order is already placed and the dish is still not delivered, but a new order can be placed at the exact time when the dish is delivered.

E869120 arrives at this restaurant at time 0. He will order all five dishes. Find the earliest possible time for the last dish to be delivered.

Here, he can order the dishes in any order he likes, and he can place an order already at time 0.

Constraints

- A, B, C, D and E are integers between 1 and 123 (inclusive).

Input

Input is given from Standard Input in the following format:

A
 B
 C
 D
 E

Output

Print the earliest possible time for the last dish to be delivered, as an integer.

Exemplos de entradas e saídas

Entrada

29

20

7

35

120

101

86

119

108

57

Saída

215

481

Solução com complexidade $O(1)$

- A solução consiste em determinar uma permutação dos pratos A, B, C, D e E que minimize o tempo para servi-los
- Se não houvesse a restrição de que os pratos só podem ser pedidos em instantes de tempo que são múltiplos de 10, qualquer permutação levaria ao mesmo resultado
- Há $5! = 120$ permutações possíveis, as quais podem ser geradas por meio da função `next_permutation()`
- O i -ésimo prato requer t_i minutos para ser servido após ser pedido e, exceto pelo último, é preciso esperar o próximo múltiplo de 10 para ser pedido
- Uma forma de tratar esta espera é substituir t_i pelo menor múltiplo de 10 maior ou igual a t_i para todos os pratos, exceto o último

Solução com complexidade $O(1)$

```
7 int solve(vector<int> xs)
8 {
9     sort(xs.begin(), xs.end());
10    int ans = 00;
11
12    do {
13        int t = xs.back();
14
15        for (int i = 0; i < 4; ++i)
16            t += 10 * ((xs[i] + 9)/10);
17
18        ans = min(ans, t);
19    } while (next_permutation(xs.begin(), xs.end()));
20
21    return ans;
22 }
```