

24. Покупка билетов

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

За билетами на премьеру нового мюзикла выстроилась очередь из N человек, каждый из которых хочет купить 1 билет. На всю очередь работала только одна касса, поэтому продажа билетов шла очень медленно, приводя «постояльцев» очереди в отчаяние. Самые сообразительные быстро заметили, что, как правило, несколько билетов в одни руки кассир продаёт быстрее, чем когда эти же билеты продаются по одному. Поэтому они предложили нескольким подряд стоящим людям отдавать деньги первому из них, чтобы он купил билеты на всех.

Однако для борьбы со спекулянтами кассир продавала не более 3-х билетов в одни руки, поэтому договориться таким образом между собой могли лишь 2 или 3 подряд стоящих человека.

Известно, что на продажу i -му человеку из очереди одного билета кассир тратит A_i секунд, на продажу двух билетов — B_i секунд, трех билетов — C_i секунд. Напишите программу, которая подсчитает минимальное время, за которое могли быть обслужены все покупатели.

Обратите внимание, что билеты на группу объединившихся людей всегда покупает первый из них. Также никто в целях ускорения не покупает лишних билетов (то есть билетов, которые никому не нужны).

Формат ввода

На вход программы поступает сначала число N — количество покупателей в очереди ($1 \leq N \leq 5000$). Далее идет N троек натуральных чисел A_i, B_i, C_i . Каждое из этих чисел не превышает 3600. Люди в очереди нумеруются, начиная от кассы.

Формат вывода

Требуется вывести одно число — минимальное время в секундах, за которое могли быть обслужены все покупатели.

Пример

Ввод	Вывод
5 5 10 15 2 10 15 5 5 5 20 20 1 20 1 1	12

Язык

GNU GCC 12.2 C++20

Набрать здесь

Отправить файл

```
1 #include <iostream>
2 #include <vector>
3
4 struct Combo {
5     int a;
6     int b;
7     int c;
8 };
9
10 int main() {
11     int n;
12     std::cin >> n;
13     std::vector<Combo> var(n);
14     for (int i = 0; i != n; ++i) {
15         std::cin >> var[i].a >> var[i].b >> var[i].c;
16     }
17     std::vector<int> dp(n);
18     dp[0] = var[0].a;
19     if (n > 1) {
20         dp[1] = std::min(dp[0] + var[1].a, var[0].b);
21     }
22     if (n > 2) {
23         dp[2] = std::min( std::min(dp[1] + var[2].a, dp[0] + var[1].b), var[0].c);
```

```
24     }
25     if (n > 3) {
26         for (int i = 3; i != n; ++i) {
27             dp[i] = std::min(std::min(dp[i - 1] + var[i].a, dp[i - 2] + var[i - 1].b), dp[i - 3] + var[i - 2].c);
28         }
29     }
30     std::cout << dp[n - 1];
31     return 0;
32 }
```

[Отправить](#)[Предыдущая](#)[Следующая](#)