

Разбор некоторых задач контеста  
«Задачи-2» из темы  
«3. Динамическое программирование»

Егор Подлесов

02.02.2022

## Задача А. Минимум предметов

Как набрать вес в точности  $M$ , используя как можно меньше предметов из множества  $\{m_1, \dots, m_N\}$ ?

► Решать будем при помощи двумерной динамики.

Пусть  $dp[i][j]$  равно минимальному количеству предметов, которое нужно взять из множества  $\{m_1, \dots, m_i\}$ , чтобы собрать вес  $j$ . Если вес  $j$  собрать нельзя, то  $dp[i][j] = \text{inf}$ , где  $\text{inf}$  — достаточно большое число, например,  $\text{inf} = 10^9$ .

Переход будет выглядеть следующим образом:

$dp[i][j] = \min\{dp[i-1][j], dp[i-1][j-m_i]\}$ , если конечно  $j \geq m_i$ , иначе у нас нет выбора, и мы берём значение  $dp[i-1][j]$ .

Таким образом осталось определиться с граничными значениями. Предлагается сделать следующее: так как чтобы собрать вес равный 0 из любого подмножества предметов нам потребуется 0 предметов, то

$\forall i \in [1, N] \Rightarrow dp[i][0] = 0$ , а также  $dp[1][m_1] = 1$ , все остальные элементы двумерного массива  $dp$  сделаем равными  $\text{inf}$ . ◀

Временная сложность решения:  $O(NM)$

## Задача В. Гирьки: кучки одного размера

Разделите набор  $\{m_1, \dots, m_N\}$  на две кучки равной массы, содержащие равное число гирек.

►

Сперва нужна парочка костылей, а именно: сразу отмечаем варианты когда общая сумма или общее количество на 2 не делится. Если всё же обе величины чётные, то сразу делаем так  $m = m/2$ , ведь нас интересует лишь возможность формирования из гирек в сумме веса  $m/2$  и более того лишь из  $n/2$  гирек. Данное решение подразумевает динамику по трём аргументам.

Пусть  $able[i][k][j]$  равно **true**, если используя  $k$  гирек из множества  $\{m_1, \dots, m_i\}$ , можно собрать вес  $j$  и **false** в противном случае.

Переход:

```
1 able[i][k][j] = able[i-1][k][j];
2 if (k > 0 && j >= w[i]) able[i][k][j] |= able[i-1][k-1][j-w[i]];
```

где  $w[i]$  означает вес  $i$ -й гирьки.

Теперь к граничным значениям. Тут всё просто, всё, что нужно, это установить  $able[0][0][0] = \text{true}$ , вполне логично, что из 0 гирек можно собрать вес 0, притом, как ни странно, этого достаточно, все остальные значения массива  $able$  устанавливаем в **false**.

**Где хранится ответ и как восстанавливать путь?**

Ответ хранится в  $able[n][n/2][m]$ , при условии, что мы поделили  $m$  на 2.

Восстанавливаем его мы следующим образом:

```
1 vector<int> b, c;
2 int cnt = n/2;
3 for (int i = n; i >= 1; --i) {
4     if (cnt > 0 && m >= w[i] && able[i-1][cnt-1][m-w[i]]) {
5         b.PB(i);
6         --cnt;
7         m -= w[i];
8     } else c.PB(i);
9 }
```

Где  $b$  и  $c$  контейнеры для первой второй кучки ответа. ◀

Временная сложность решения:  $O(N^2M)$