

---

## Разбор задачи «Бюджет»

Решим задачу с помощью динамического программирования. Будем поддерживать массив  $dp[i][j]$ , в ячейке которого будем хранить значение «true», если на  $i$ -ом префиксе можно получить сумму  $j$ , «false» — если нельзя. База динамики:  $dp[0][f[i]] = true$  (если  $f[i]$  попадает в отрезок  $[a, b]$ ),  $dp[0][-f[i]] = true$  (аналогично, если  $-f[i]$  попадает в отрезок  $[a, b]$ ). Соответственно, во внешнем цикле будем перебирать все префиксы (по  $i$ ), во внутреннем — суммы от  $a$  до  $b$  (по  $j$ ). Если на предыдущем префиксе набралась сумма  $j$ , то есть  $dp[i-1][j] == true$ , значит на текущем префиксе набирается сумма  $j + f[i]$ , а также  $j - f[i]$ .

Если на последнем префиксе (то есть на всем массиве) набралась хотя бы одна сумма, лежащая на отрезке  $[a, b]$ , значит ответ на задачу существует. Чтобы восстановить ответ, необходимо знать, прибавляли ли мы или вычитали текущее число в массиве, будем также хранить эти данные в ячейке массива  $dp$ . Тогда, зная, какая сумма могла быть получена на последнем префиксе и какой знак был у последнего числа в массиве при получении этой суммы, мы можем переходить по суммам на меньших префиксах и получать знаки остальных чисел.

Асимптотика:  $O(n \cdot (b - a))$ .