

Баланс настроения

Автор задачи и разработчик: Константин Бац

Рассмотрим величину $y_i = x_i + 2i$, где x_i — настроение Кости в i -ю минуту, а i — собственно, минута прогулки. Проследим, как меняется эта величина при переходе от минуты i к минуте $i + 1$.

- Если Косте пришла в голову негативная мысль, x_i превращается в $x_{i+1} = x_i - 1$, поэтому $y_{i+1} = (x_i - 1) + 2(i + 1) = y_i + 1$;
- Если же Косте пришла в голову неоднозначная мысль, x_i превращается в $x_{i+1} = 2(x_i + (i + 1) - 2) = 2(x_i + i - 1)$, то есть $y_{i+1} = 2(x_i + i - 1) + 2(i + 1) = 2y_i$.

Таким образом, с каждой следующей минутой прогулки величина y либо увеличивается на один, или увеличивается в два раза. Заметим так же, что изначально, до первой минуты прогулки, $y_0 = 0$, а в конце прогулки мы хотим получить $y_n = 0 + 2n$. Получается, что мы свели задачу к получению числа $2n$ из числа 0 операциями прибавления единицы и умножения на два. Более того, требуется как можно меньше раз использовать операцию, которая дает $y_{i+1} = y_i + 1$, так как требуется минимизировать количество негативных мыслей.

Посмотрим на первый раз, когда будет применена операция увеличения на 1. Заметим, что после этого y_i будет только расти, и при этом

- мы не можем сделать больше $\lfloor \log_2(2n) \rfloor$ умножений на два, потому что тогда конечный y_n будет больше $2n$;
- мы обязаны сделать хотя бы $\text{bitcount}(2n)$ операций прибавления единицы, потому что операция умножения на два не увеличивает количество единичных бит в числе.

Из этих двух наблюдений получается алгоритм: будем «собирать» число $2n$ по битам от старших к младшим: каждый раз умножаем текущий результат на два (неоднозначная мысль), и, если соответствующий бит в числе $2n$ равен единице, прибавляем к результату 1 (негативная мысль). Время работы алгоритма — $\mathcal{O}(\log n)$.