

## 2. Палитра (pal)

3 секунды

100 очков

Компания Gold & Silver Soft разрабатывает новый графический редактор. Один из его модулей — это управление цветовой палитрой. При запуске программы палитра пуста. Затем пользователь может либо добавить новые цвета, либо спросить, который из имеющихся в палитре цветов наиболее похож на данный цвет.

Цвета представлены  $K$ -битными числами (со значениями от 0 до  $2^K - 1$ ), и схожестью двух цветов считается число совпадающих битов в их двоичных представлениях. Например, при  $K = 5$  схожесть цветов 00110 и 10101 равна 2, так как совпадают значения только второго и третьего бита слева.

Написать для редактора модуль управления палитрой, в котором будут следующие функции:

Функция	Описание
<code>void init(int k, int n)</code>	Инициализация палитры для использования $k$ -битных цветов. Эта функция вызывается один раз в начале работы программы, затем следуют в общей сложности $n$ вызовов <code>add</code> и <code>find</code> .
<code>void add(int c)</code>	Добавление в палитру цвета $c$ .
<code>int find(int c)</code>	Поиск в палитре наилучшего соответствия цвету $c$ . Функция должна возвращать цвет из палитры наиболее схожий с цветом $c$ . Если максимально схожих цветов несколько, возвращается любой из них. Эта функция вызывается только тогда, когда в палитре уже есть хотя бы один цвет.
<code>void done()</code>	Завершение работы. Эта функция вызывается один раз в конце работы программы.

В тестовой среде есть файлы примеров, где необходимые функции уже описаны и в виде решения требуется написать только их имплементации. В файл решения можно записывать и свои функции. Для тестирования своего решения на своём компьютере есть также и пример оценивающей программы, описание входных и выходных данных которой приведён ниже (на сервере используется другая оценивающая программа, которая проверяет верность ответов, возвращаемых решением). Для компиляции и тестирования своего решения на своём компьютере:

Язык	Решение	Командная строка
C++	<code>pal.cpp</code>	<code>g++ -o main main.cpp pal.cpp ./main</code>
Java	<code>pal.java</code>	<code>javac -cp . main.java pal.java java -cp . main</code>
Python	<code>pal.py</code>	<code>python main.py</code>

**Входные данные.** Первая строка текстового файла `palsis.txt` содержит количество битов  $K$  ( $1 \leq K \leq 20$ ), используемых для представления цветов, и количество операций  $N$  ( $1 \leq N \leq 10^6$ ). Каждая из следующих  $N$  строк содержит целые числа  $T_i$  и  $C_i$  ( $0 \leq C_i < 2^K$ ), где  $T_i = 1$  означает добавление в палитру цвета  $C_i$ , а  $T_i = 2$  — поиск в палитре наилучшего соответствия цвету  $C_i$ .

**Выходные данные.** В текстовый файл `palval.txt` выводится лог общения оценивающей программы и модуля.

**Пример.**

palsis.txt	palval.txt
2 3	init(2, 3)
1 1	add(1)
2 0	find(0) = 1
2 1	find(1) = 1
	done()

**Пример.**

palsis.txt	palval.txt
3 5	init(3, 5)
1 0	add(0)
1 7	add(7)
2 1	find(1) = 0
1 1	add(1)
2 1	find(1) = 1
	done()

**Оценивание.** В тестах суммарной стоимостью 30 очков выполняется дополнительное условие  $2^K \cdot N \leq 50\,000\,000$ .