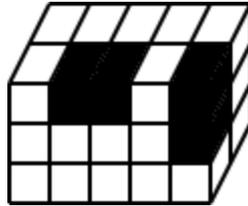


**1. КУБИКИ**

10 очков

5 секунд

Параллелепипед размерами  $N \times M \times K$  составлен из прозрачных и черных единичных кубиков. Даны три проекции параллелепипеда – спереди, сверху и справа – в каждой из которых прозрачными являются те и только те места, где целый столбец состоит только из прозрачных кубиков. Написать программу, которая проверяет, что эти три проекции находятся в соответствии друг с другом (не противоречат друг другу).



Входные данные. На первой строке текстового файла KUUBID.SIS находятся три целых числа – размеры параллелепипеда: высота  $N$  ( $1 \leq N \leq 50$ ), ширина  $M$  ( $1 \leq M \leq 50$ ) и глубина  $K$  ( $1 \leq K \leq 30$ ). На каждой из следующих  $N$  строк находится ровно  $M$  цифр – проекция спереди, в которой 0 означает прозрачное место и 1 – непрозрачное. Проекция спереди задана послойно сверху вниз и слева направо. На каждой из следующих  $K$  строк находится ровно  $M$  чисел – проекция сверху. Она задана послойно сзади вперед и слева направо. На каждой из следующих  $N$  строк находится ровно  $K$  чисел – проекция справа. Она задана послойно сверху вниз и спереди назад.

Выходные данные. Если заданные три проекции противоречат друг другу, вывести на первую строку текстового файла KUUBID.VAL слово EI. Если согласованы – слово JAN и на следующих  $K \times M$  строках описание одного возможного параллелепипеда послойно спереди назад, в каждом слое ряды сверху вниз и слева направо.

| <u>Пример.</u> | KUUBID.SIS | KUUBID.VAL |
|----------------|------------|------------|
|                | 3 5 2      | JAN        |
|                | 01101      | 01101      |
|                | 00001      | 00001      |
|                | 00000      | 00000      |
|                | 00000      | 00000      |
|                | 01101      | 00000      |
|                | 10         | 00000      |
|                | 10         |            |
|                | 00         |            |

Оценивание. В этом задании за EI-тесты очки получают только те программы, которые верно решат все JAN-тесты.

**2. ВЫРАЖЕНИЕ**

20 очков

5 секунд

Дано целое число  $X$  и выражение в виде  $A_1 + (A_2 + (A_3 + (\dots + (A_N) \dots)))$ . Написать программу, которая заменяет в этом выражении часть минусов плюсами, так чтобы значение полученного выражения было равно  $X$ .

Входные данные. В первой строке текстового файла AVALDIS.SIS дано целое число  $X$  ( $-100000 \leq X \leq 100000$ ), во второй строке – “длина” выражения  $N$  ( $1 \leq N \leq 100$ ), и на каждой из следующих  $N$  строк дано целое число  $A_i$  ( $-1000 \leq A_i \leq 1000$ ).

Выходные данные. В первую строку текстового файла AVALDIS.VAL вывести ровно  $(N-1)$  знаков ‘+’ и ‘-’ в том порядке, каком они находятся в выражении. Если такое значение получить невозможно, вывести строку EI SAA.

|                |             |             |
|----------------|-------------|-------------|
| <u>Пример.</u> | AVALDIS.SIS | AVALDIS.VAL |
|                | 0           | + -         |
|                | 3           |             |
|                | 1           |             |
|                | 2           |             |
|                | 3           |             |

**3. УРАВНЕНИЕ**

30 очков

5 секунд

Обычный вид представления арифметических выражений, то есть в котором знак операции находится между аргументами, называется инфиксным. Например,  $(A+B) * (C-D)$ . Существует и другой вид представления выражений – постфиксный, в котором знак операции находится после ее аргументов, например,  $A B + C D - *$ . Постфиксный вид более удобен для обработки на компьютере, так как при достижении любого знака действия необходимые для этого данные уже находятся в памяти. Написать программу, которая умеет решать простые уравнения, левая и правая части которого (выражения!) даны в постфиксном виде.

Входные данные. В первой строке текстового файла VORRAND.SIS дана левая часть уравнения, во второй строке – правая. Оба выражения даны в постфиксном виде, могут содержать целые числа, переменные с однобуквенными именами и четыре основных действия. Все величины отделены друг от друга пробелами. В третьей строке файла дано имя переменной, которую нужно выразить. Известно, что эта переменная встречается в уравнении ровно один раз.

Выходные данные. В первую строку текстового файла VORRAND.VAL вывести выражение для данной переменной, также в постфиксном виде. Упрощать выражение не требуется.

|                |             |               |
|----------------|-------------|---------------|
| <u>Пример.</u> | VORRAND.SIS | VORRAND.VAL   |
|                | 2 x * 3 +   | y 5 + 3 - 2 / |
|                | y 5 +       |               |
|                | x           |               |

**4. ЗАМКНУТЫЕ ПОДМНОЖЕСТВА**

40 очков

5 секунд

Рассмотрим функцию одной переменной  $f(x)$ . Множество  $H$  называется замкнутым относительно функции  $f$ , если для любого его элемента  $x$  значение  $f(x)$  также принадлежит множеству  $H$ . Написать программу, которая для заданной функции  $f$  и замкнутого относительно нее множества  $H$  перечисляет все подмножества множества  $H$ , замкнутые относительно  $f$ .

Входные данные. В первой строке текстового файла HULGAD.SIS дано число  $N$  элементов множества  $H$  ( $1 \leq N \leq 100$ ). Предполагается, что элементы множества  $H$  пронумерованы числами от 1 до  $N$ . В каждой из следующих  $N$  строк дано целое число: таблица значений функции  $f$ , то есть если в  $(i+1)$ -ой строке находится число  $k$ , это значит, что  $f(h_i) = h_k$ .

Выходные данные. В единственную строку текстового файла HULGAD.VAL вывести число замкнутых подмножеств множества  $H$ .

|                |            |            |
|----------------|------------|------------|
| <u>Пример.</u> | HULGAD.SIS | HULGAD.VAL |
|                | 3          | 3          |
|                | 2          |            |
|                | 3          |            |
|                | 2          |            |

Пояснение. В приведенном примере подмножество  $\{h_2, h_3\}$  замкнутое, так как  $f(h_2) = h_3$  и  $f(h_3) = h_2$ , то есть также принадлежат этому подмножеству. А например, подмножество  $\{h_1, h_2\}$  не является замкнутым, так как  $f(h_2) = h_3$ , то есть не принадлежит этому подмножеству. Пустое множество всегда является замкнутым относительно любой функции.