

**1. ОПЕРАЦИИ НАД МНОЖЕСТВАМИ**

30 очков

10 секунд

Написать программу, которая проводит действия над множествами, элементами которых являются непустые последовательности строчных латинских букв (от а до з).

Входные данные: На единственной строке (длиной до 100 символов) текстового файла HULK.SIS находится выражение вида  $\text{множ}_1 + \text{множ}_2$ ,  $\text{множ}_1 - \text{множ}_2$  или  $\text{множ}_1 * \text{множ}_2$ , где знаки +, - и \* обозначают соответственно объединение, разность и пересечение множеств. Множества представлены как заключённые в фигурные скобки списки элементов. Элементы разделены друг от друга запятыми и никакой элемент не встречается в одном множестве более одного раза. Список элементов может быть и пустым.

Выходные данные: На единственную строку текстового файла HULK.VAL вывести значение входного выражения в формате указанном в описании входных данных. Порядок элементов в списке произвольный.

Примечание: Объединение двух множеств состоит из элементов, которые принадлежат хотя бы одному из этих двух множеств. Разность двух множеств состоит из элементов, которые принадлежат первому, но не принадлежат второму множеству. Пересечение двух множеств состоит из элементов, которые принадлежат обоим множествам.

Пример1: HULK.SIS {a, abc, ca, aad} + {a, bb, ca} HULK.VAL {a, abc, ca, aad, bb}

Пример2: HULK.SIS {a, b} - {} HULK.VAL {a, b}

**2. ЛОЗУНГИ**

30 очков

10 секунд

Малое предприятие, занимающееся производством лозунгов, получает свои материалы от завода в форме блоков вида ху, ух, хху, хуу, ухх или уух, где х обозначает любого согласного и у любого гласного. Для блоков каждого вида завод установил фиксированную цену. Чтобы заработать максимальную прибыль, предприятие заказывает материалы для каждого лозунга по самому дешёвому разложению текста лозунга на блоки, а своему клиенту представляет счет согласно самому дорогому разложению.

Входные данные: На первой строке текстового файла LOOSUNG.SIS 6 разделённых пробелами целых чисел от 1 до 100 – цены, установленные заводом на блоки вида ху, ух, хху, хуу, ухх и уух. Во второй строке (длиной до 1000 символов) – текст лозунга, состоящий из строчных латинских букв и пробелов.

Выходные данные: На единственную строку текстового файла LOOSUNG.VAL вывести одно целое число – разница между ценами самого дорогого и самого дешёвого разложения. Если текст невозможно разложить на блоки данных видов, вывести -1.

Примечание: В латинском алфавите гласными считают а, е, и, о и у (но не y).

Пример: LOOSUNG.SIS 1 1 1 1 1 1 LOOSUNG.VAL 2  
baabbaabbaab

**3. ТАБЛИЦА ПРОЦЕССОВ**

40 очков

10 секунд

В многопользовательских операционных системах (например УНИХ) могут одновременно выполняться программы многих разных пользователей. Каждую программу запускает некий пользователь, полномочия которого определяют права пользования разными ресурсами для процессов, запущенных им. Любой процесс может породить новый процесс и тогда порождающего процесса называют родителем нового процесса.

При запуске компьютера автоматически порождается процесс с идентификатором пользователя, равным 0 (идентификатор суперпользователя), и идентификатором процесса, равным 1. Все последующие процессы порождаются неким имеющимся процессом. Обычно порожденный процесс наследует идентификатор пользователя от своего родителя (те новый процесс получает такие же права пользования ресурсами компьютера). В качестве исключения процессам суперпользователя разрешено порождать процессы других пользователей. Если процесс обыкновенного пользователя порождает процесс, который принимает полномочия другого пользователя, это угрожает сохранности данных другого пользователя, поэтому администратору нужна программа для выявления таких нарушений.

Входные данные: На первой строке текстового файла PROTS.SIS общее число процессов в компьютере ( $N$ ,  $1 \leq N \leq 1000$ ) и на каждой из последующих  $N$  строк информация об одном процессе в виде PID PPID UID, где PID идентификатор процесса и PPID идентификатор его родителя, оба целые числа ( $1 \leq PPID \leq PID \leq 10000$ ), а UID идентификатор пользователя-владельца процесса ( $0 \leq UID \leq 10000$ ). Процессы представлены в порядке возрастания идентификаторов (PID). Родителем корневого процесса (PID=1) считают его самого.

Выходные данные: На первую строку текстового файла PROTS.VAL вывести число пользователей, которые нарушают вышеуказанные правила (X) и на каждую из следующих X строк информацию об одном нарушителе в формате UID UID<sub>1</sub> UID<sub>2</sub> ..., где UID – идентификатор пользователя-нарушителя, а UID<sub>1</sub>, UID<sub>2</sub> итд идентификаторы тех пользователей, процессы которых происходят (непосредственно или косвенно) от процессов UID. Нарушителем считают только инициатора цепочки нарушения (те если процесс одного пользователя порождает процесс другого пользователя, а тот в свою очередь процесс третьего, то второй пользователь является жертвой первого и его не следует считать нарушителем). Строки выходного файла вывести в порядке возрастания UID, а UID<sub>1</sub>, UID<sub>2</sub> итд на каждой строке расположить в возрастающем порядке.

Пример:

PROTS.SIS	PROTS.VAL
10	2
1 1 0	1 2 3
2 1 1	4 0 5
4 1 4	
5 4 0	
6 1 7	
8 2 2	
9 6 7	
10 8 3	
11 5 5	
12 6 7	