

Эстонская школьная олимпиада по информатике

13-14 марта 1999 года

Задачи заключительного тура для младшей группы

1. ЧАСТОТНЫЙ СЛОВАРЬ

20 очков

10 секунд

Для анализа слов, используемых писателями, филологи нуждаются в программе, которая находила бы частоты появления слов в заданном тексте.

Ввод. В текстовом файле `SONAD.SIS` находится текст, слова которого состоят из больших латинских букв, а все другие символы и переводы строки считают разделителями слов. Количество слов в файле не превышает 1000, длины слов не превышают 20 букв, а длины строк не превышают 80 символов.

Вывод. В файл `SONAD.VAL` вывести список всех разных слов, присутствовавших во входном файле, вместе с их частотами появления. Информацию о каждом слове вывести в отдельную строку. Порядок строк во выходном файле не важен.

Пример.

<code>SONAD.SIS</code>	<code>SONAD.VAL</code>
<code>AIAS SADAS</code>	<code>AIAS 2</code>
<code>SAIA. SAIAS</code>	<code>SADAS 3</code>
<code>SADAS AIA.</code>	<code>SAIA 2</code>
<code>SAIA SADAS,</code>	<code>SAIAS 1</code>
<code>AIAS.</code>	<code>AIA 1</code>

2. ПРОЦЕССОР

40 очков

10 секунд

Фирма `MegaPro` в последний момент перед запуском в производство своего нового процессора обнаружила, что на нём отсутствует одно соединение между точками `A` и `B` (на рисунке слева). Для запуска производства требуется добавить недостающее соединение так, чтобы у него небыло соприкосновений ни с одним уже существующим соединением (на рисунке справа). Процессор состоит из точек, находящихся в узлах клетчатой сетки, а также из их соединений. Соединения состоят из звеньев, причём каждое звено является стороной клетки.



Ввод. На первой строке входного файла `KIIP.SIS` находятся размеры основы процессора `x` и `y` (оба целые числа из промежутка от 1 до 100) и количество присутствующих звеньев соединений `n`. На следующих `n` строках находятся координаты начальной и конечной точек каждого из звеньев $x_{i1} y_{i1} x_{i2} y_{i2}$ ($0 \leq x_{i1} \leq x, 0 \leq y_{i1} \leq y, 0 \leq x_{i2} \leq x, 0 \leq y_{i2} \leq y$). На последней строке находятся координаты точек, которые надо соединить $x_A y_A x_B y_B$ ($0 \leq x_A \leq x, 0 \leq y_A \leq y, 0 \leq x_B \leq x, 0 \leq y_B \leq y$). Эти точки не соединены ни с одной другой точкой. Координатами нижнего левого угла является пара `0 0`, а верхнего правого – пара `x y`.

Вывод. На первую строку файла `KIIP.VAL` вывести `EI SAA` ('НЕВОЗМОЖНО' по-эстонски), если требуемого соединения невозможно создать. Если же это возможно, то вывести наименьшее количество звеньев, добавлением которых возможно создать требуемое соединение.

Пример.

<code>KIIP.SIS</code>	<code>KIIP.VAL</code>
<code>4 2 5</code>	<code>6</code>
<code>0 1 0 2</code>	
<code>0 2 1 2</code>	
<code>2 0 3 0</code>	
<code>3 0 3 1</code>	
<code>3 0 4 0</code>	

3. КРЕПОСТЬ

40 очков

10 секунд

Маленькая Кати любит вырезать из бумаги. Вчера она сделала картинку крепости. У крепости две оборонительные стены: передняя стена серого цвета, а задняя – чёрного (на рисунке слева). Обе стены имеют форму прямоугольника с прямоугольными зубцами у верхнего края, но их размеры могут отличаться. Для получения картинки Кати наклеила их одну на другую (на рисунке справа). Брат Кати по имени Мати хочет узнать площадь той части задней фигуры, что видна из-за передней.



Ввод. На первой строке файла `KINDLUS.SIS` находится количество углов передней (серой) стены n_1 , а на следующих n_1 строках на каждой находятся координаты одного угла начиная с нижнего левого угла стены и заканчивая правым нижним углом. На следующей строке находится количество углов задней (чёрной) стены n_2 и на следующих n_2 строках – координаты углов. Все координаты – целые числа из отрезка от 0 до 1000, причём y -координаты последнего и первого углов стен равны 0, а все остальные y -координаты положительны. Начало системы координат – в нижнем левом углу картинки. Все зубцы прямоугольны.

Вывод. На первую строку файла `KINDLUS.VAL` вывести искомую площадь.

Пример.

<code>KINDLUS.SIS</code>	<code>KINDLUS.VAL</code>
12	23
1 0	
1 7	
3 7	
3 4	
5 4	
5 5	
6 5	
6 3	
8 3	
8 6	
10 6	
10 0	
12	
4 0	
4 9	
5 9	
5 8	
6 8	
6 10	
7 10	
7 8	
8 8	
8 9	
9 9	
9 0	