

Отборочный этап 11 класса. 2 тур (приведен один из вариантов заданий)

1. Электронные таблицы. Адресация ячеек и вычисления (3 балла)

[Снова 100x100]

Дан фрагмент электронной таблицы в режиме отображения формул:

	A	B	C	D
1	1	=A1+1		
2	=A1+1	=A1+МИН(B\$1;\$A2)		
3				
4				
5				

Ячейку B1 скопировали во все ячейки диапазона C1:CV1. Ячейку A2 скопировали во все ячейки диапазона A3:A100. Ячейку B2 скопировали во все ячейки диапазона B2:##, где ## - адрес некоторой ячейки в диапазоне B2:CV100. Сколько существует различных адресов ячеек ##, таких, что в результате в диапазоне A1:## число 70 будет встречаться ровно 21 раз? В ответе укажите целое число.

Рекомендации к пояснению решения задачи.

В пояснении к решению задачи раскройте следующие вопросы:

- 1. Решали ли Вы задачу исключительно в электронных таблицах или использовали программирование и/или аналитические методы на каком-то этапе решения задачи?*
- 2. Для части решения, использующего электронные таблицы, охарактеризуйте операции, которые Вы в них осуществили, назначение введенных в ячейки формул, и как это позволило получить ответ.*
- 3. В случае программного решения, поясните, алгоритм, который Вы реализовали, включая назначение используемых переменных и структур данных, и как это позволило получить ответ.*
- 4. Если в какой-то части решения Вы использовали аналитические методы решения, опишите, какие зависимости, закономерности и как Вы определили из условия или результатов вычислений в электронных таблицах, какие данные и формулы использовали для вычислений, и как это позволило получить ответ.*

Рекомендуемое время пояснения решения 1-3 минуты.

Ответ:

2. Сортировка и фильтрация данных (2 балла)

[Маски файлов]

В ряде операционных систем для выделения подмножества из множества файлов в каталоге используются маски файлов. Для задания масок файлов используют следующие обозначения:

c – любой неспециальный символ, соответствует самому себе, не может быть звездочкой (*) или вопросительным знаком (?).

? – ровно один неспециальный символ.

* – любое количество (в том числе 0) неспециальных символов.

Есть каталог с файлами и известно количество файлов, которые оказались выделены при использовании некоторых масок файлов:

Маска файла	Количество файлов
??*a?	100
cb*?d	45
????	300
c?*ae	70
?b*?d	200

В этом каталоге выделили все файлы, которые соответствуют хотя бы одной из двух масок:

ag*??

a?*f?

Какое максимальное количество файлов могло оказаться выделенными? В ответе укажите целое число.

Рекомендации к пояснению решения задачи.

В пояснении к решению задачи раскройте следующие вопросы:

1. Решали ли Вы задачу исключительно аналитически или использовали средства операционной системы или программирования?
2. В аналитической части решения опишите ход Ваших рассуждений, какие формулы Вы использовали для вычислений, и как это позволило получить ответ.
3. Если Вы в какой-то части решения использовали средства операционной системы, укажите её версию и как использование её средств позволило получить ответ.

Рекомендуемое время пояснения решения 1-3 минуты.

Ответ:

3. Мультимедиа технологии (2 балла)

[Шашечки]

Цветное изображение, пиксели которого закодированы с помощью цветовой модели RGB, может быть представлено в виде яркостных представлений отдельных каналов. На яркостном представлении канала белый цвет обозначает максимальную яркость этого канала в значении цвета пикселя изображения, а черный – нулевое значение яркости. Яркостные представления каналов расположены сверху вниз как красный канал, затем зеленый канал и затем синий канал. Исходное изображение содержало пиксели четырех различных цветов.

Обработка исходного изображения заключалась в следующем:

1. Исходное изображение перевели в цветовую модель HSB (Hue, Saturation, Brightness).
2. К цветовой координате Hue каждой точки прибавили некоторое значение dH. Поскольку цветовая координата Hue не может превышать 359, то если сумма оказывалась больше, из значения вычитали 360.
3. Изображение перевели обратно в цветовую модель RGB.

В результате таких обработок, отличающихся только значениями dH, получили 11 изображений, представленных в таблице:

Исходное изображение	Изображение №1	Изображение №2	Изображение №3	Изображение №4	Изображение №5
----------------------	----------------	----------------	----------------	----------------	----------------

Изображение №6 	Изображение №7 	Изображение №8 	Изображение №9 	Изображение №10 	Изображение №11

Определите и укажите в ответе через пробел два числа: сначала номер изображения, полученного в результате обработки со значением $dh=60$, а затем номер изображения, полученного в результате обработки со значением $dh=210$. Пример записи ответа: 1 2

Ответ:

4. Телекоммуникационные технологии (1 балл)

[Три узла]

В терминологии сетей TCP/IP маской сети называется двоичное 32-х разрядное число, определяющее, какая часть IP-адреса узла относится к адресу сети, а какая – к адресу самого узла в этой сети. При этом в маске сначала (в старших разрядах) стоят только единицы, а затем с некоторого разряда – только нули. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. Два узла считаются принадлежащими одной сети, если у них совпадают адреса сети.

Известно, что узлы с IP-адресами 192.168.215.91 и 192.168.151.91 принадлежат одной сети, а узел с IP-адресом 194.168.215.91 не принадлежит этой сети. Сколько существует различных масок сетей, при которых это возможно. В ответе укажите целое число.

Рекомендации к пояснению решения задачи.

В пояснении к решению задачи раскройте следующие вопросы:

1. Решали ли Вы задачу исключительно аналитически или использовали электронные таблицы и/или программирование на каком-то этапе решения задачи?

2. В аналитической части решения опишите, какие зависимости, закономерности и как Вы определили из условия, какие данные и формулы использовали для вычислений, и как это позволило получить ответ.

3. Если в какой-то части решения Вы использовали электронные таблицы, охарактеризуйте операции, которые Вы в них осуществили, назначение введенных в ячейки формул, и как это позволило получить ответ.

4. Если в какой-то части решения Вы использовали программирование, поясните, алгоритм, который Вы реализовали, включая назначение используемых переменных и структур данных, и как это позволило получить ответ.

5. Операционные системы (3 балла)

[Кэш памяти]

Большинство современных операционных систем являются мультипрограммными и используют концепцию виртуальной памяти. Это означает, что в исполняемом коде каждой программы используются виртуальные адреса, отсчитываемые от 0 для каждой программы, а при обращении к ячейке памяти по виртуальному адресу операционная система осуществляет пересчет его в физический адрес. Эта процедура может отнимать значительное время, поэтому при организации вычислительного процесса важным элементом является кэш памяти.

Кэш памяти представляет собой ассоциативный массив, хранящий пары типа «ключ-значение», где в качестве ключа выступает виртуальный адрес, а в качестве значения вычисленный физический. Без использования кэша при каждом обращении программы к ячейке памяти приходилось бы осуществлять преобразование виртуального адреса в физический, что существенно замедляло бы вычисления. В случае использования кэша, для сначала производится поиск виртуального адреса в кэше. Если виртуальный адрес там нашелся, физический адрес берется из кэша. Такая ситуация называется «кэш попадание». Если виртуального адреса в кэше не оказалось, запускается процедура вычисления физического адреса. Такая ситуация называется «кэш промах». В случае кэш промаха в кэш помещается новая пара «виртуальный адрес – физический адрес», вытесняя ту пару, которая дольше всего не использовалась.

В зависимости от архитектуры вычислительной системы, кэш памяти может использоваться как для всех операций обращения, связанных с обращением к ячейке памяти так и только для конкретных, например, только операций чтения данных из ячейки памяти в регистр процессора.

Рассмотрим следующую модель вычислительной системы с кэшем памяти.

1. Программа представляет собой набор 16-битных инструкций, для удобства записанных в виде четырехразрядных шестнадцатеричных чисел. Старший разряд является номером исполняемой команды, а младшие 3 разряда – виртуальным адресом ячейки памяти, которая используется этой командой. В записи инструкции используются арабские цифры и строчные буквы из диапазона a-f.
2. Кэш памяти используется только для команд чтения ячейки памяти в регистр процессора, которым соответствует значение старшего шестнадцатеричного разряда инструкции «1». Например, инструкция 100a означает чтение данных из ячейки с виртуальным адресом 00a. Для любых других команд (инструкций, у которых старший шестнадцатеричный разряд отличается от 1) кэш памяти не используется.
3. Кэш памяти состоит из четырех ячеек, пронумерованных от 0 до 3 и изначально пустых. Каждая ячейка кроме пары «виртуальный адрес – физический адрес» хранит также счетчик. При выполнении каждой операции чтения выполняются следующие действия:
 - a. Осуществляется попытка найти в кэше памяти виртуальный адрес, указанный в выполняемой инструкции.
 - b. Если такой адрес найден в кэше (кэш попадание), счетчик для него устанавливается в 0.
 - c. Если такой адрес не найден в кэше (кэш промах), то после вычисления физического адреса определяется пустая ячейка с минимальным номером, а если пустых ячеек нет, то определяется ячейка, значение счетчика которой максимально (если таких ячеек несколько, из них берется ячейка с минимальным номером). В эту ячейку помещается новая пара «виртуальный адрес – физический адрес», а счетчик этой ячейки устанавливается в 0.
 - d. Независимо от того, получилось ли в результате выполнения инструкции кэш попадание или кэш промах, счетчики всех четырех ячеек увеличиваются на 1.

Вам дан файл с программой [ссылка на файл var1.txt]. Определите количество кэш попаданий, которое произошло при выполнении этой программы. В ответе запишите целое число.

Ответ:

6. Технологии программирования (2 балла)

[CSV]

Имя входного файла	стандартный ввод
Имя выходного файла	стандартный вывод
Ограничение по времени	2 секунды
Ограничение по памяти	256 мегабайт

CSV (от англ. Comma-Separated Values — значения, разделённые запятыми) — это текстовый формат для представления таблиц. Каждая строка таблицы задается отдельной строкой текста, а различные столбцы отделены друг от друга запятыми.

В первой строке таблицы задаются заголовки столбцов, так же отделенные друг от друга запятыми. При этом значения в ячейках таблицы могут быть пустыми.

Вам необходимо вывести все заголовки полных столбцов. Столбец считается полным, если в каждой строке, кроме заголовочной, соответствующая ячейка пуста.

Формат входных данных

Формат входных данных полностью соответствует формату CSV. Первая строка содержит заголовки.

Значения в каждой ячейке задаются строками из не более чем 2020 символов, состоящих из строчных букв латинского алфавита и цифр.

Гарантируется, что в таблице хотя бы две, но не более чем 100100 строк и хотя бы один, но не более 100100 столбцов. Также гарантируется, что заголовки столбцов различны и непусты.

Формат выходных данных

В первой строке выведите число полных столбцов. Затем выведите все заголовки полных столбцов в лексикографическом порядке, каждый в отдельной строке.

Пример

Стандартный ввод	Стандартный вывод
surname,name,phone,test1,test2,test3,test4,extra,final,mark Ivanov,Pavel,,23,36,,45,,47, Kuznetsov,Ivan,,78,88,,77,,45, Sidorov,Alexander,,90,80,,100,90,46, Popov,Mike,,11,0,,4,,2 Bobov,Oleg,,,20,,30,,40,	4 name surname test2 test4

Замечание

Таблица из примера:

surname	name	phone	test1	test2	test3	test4	extra	final	mark
Ivanov	Pavel		23	36		45		47	
Kuznetsov	Ivan		78	88		77		45	
Sidorov	Alexander		90	80		100	90	46	
Popov	Mike		11	0		4			2
Bobov	Oleg			20		30		40	

7. Технологии программирования (4 балла)

[Шаблоны битовых матриц]

Имя входного файла	стандартный ввод
Имя выходного файла	стандартный вывод
Ограничение по времени	4 секунды
Ограничение по памяти	256 мегабайт

Назовем шаблоном битовой матрицы размера $a \times b$ матрицу размера $a \times b$, которая может содержать только значения «0» и «?». При этом знак «?» означает любой символ.

Таким образом, например, шаблону

0	?
?	0

соответствуют четыре битовые матрицы:

0	1
1	0

0	1
0	0

0	0
1	0

0	0
0	0

Вам даны n шаблонов размера $a \times b$ и m битовых матриц такой же размерности. Необходимо для каждого шаблона определить сколько из данных матриц ему не соответствуют.

Обратите внимание на ограничения, если просто перебрать попарно матрицы и шаблоны, вы получите превышение ограничения по времени!

Формат входных данных

В первой строке заданы четыре целых числа a, b, n и m — размерности матриц, число шаблонов и число матриц, соответственно ($1 \leq a \cdot b \leq 12, 1 \leq n, m \leq 5 \cdot 10^4$).

В следующих $a \cdot n$ строках заданы шаблоны, каждый состоит из a строк по b символов. Каждый символ, соответственно, либо «0», либо «?».

Затем в $a \cdot m$ строках заданы матрицы, каждая матрица так же состоит из a строк по b символов. Каждый символ, соответственно, либо «1», либо «0».

Формат выходных данных

Выведите n чисел, разделенных пробелом — для каждого шаблона, сколько из данных матриц ему не соответствует.

Примеры:

Стандартный ввод	Стандартный вывод
2 2 3 4 0? ?0 00 00	3 3 1

?0 ?? 10 01 11 11 00 00 10 11	
2 3 1 2 ??? ??? 101 111 111 000	0