

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

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

[Матрица]

Матрицу чисел на листе электронной таблицы заполнили следующим образом:

1. Ячейки диапазона B1:CW1 заполнили последовательно по возрастанию числами от 1 до 100.
2. Ячейки диапазона A2:A101 заполнили последовательно по возрастанию числами от 1 до 100.
3. В ячейку B2 поместили формулу =ОСТАТ(B\$1;\$A2)+ОСТАТ(\$A2;B\$1)
4. Ячейку B2 скопировали во все ячейки диапазона B2:##, где ## адрес некоторой ячейки таблицы.

Определите и запишите в ответ адрес ячейки ##, если известно, что в получившейся матрице (то есть в диапазоне B2:##) ровно 86 ячеек содержат число 27, а матрица получилась квадратной (имеет равное количество строк и столбцов).

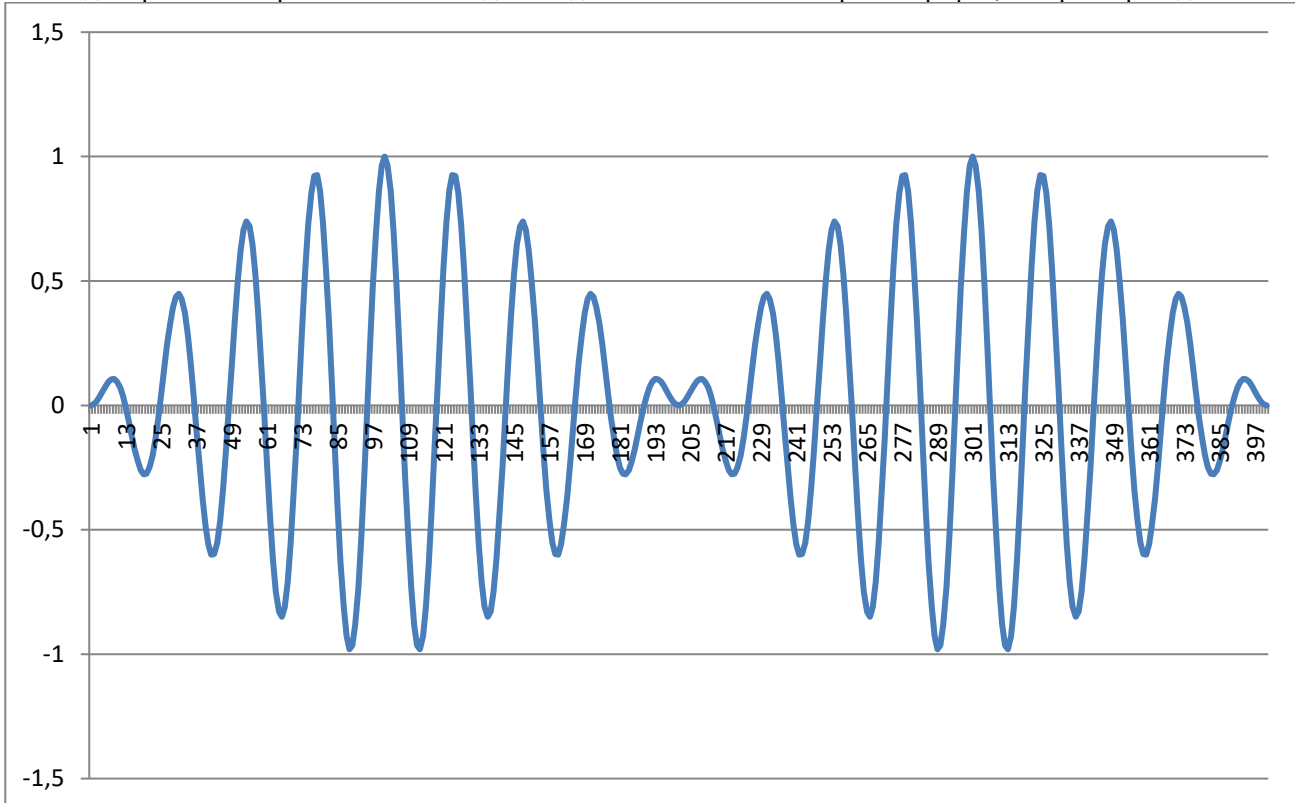
2. Электронные таблицы. Графики и диаграммы (2 балл)

[Гармоники]

Дан фрагмент электронной таблицы в режиме отображения формул:

	A	B	C	D	E	F	G	H	I	J
1	0,005				=A1*D1					
2	0	=ПИ()*A2	=SIN(B2)		0	=ПИ()*E2	=SIN(F2)		=C2*G2	
3	=A2+A\$1	=ПИ()*A3	=SIN(B3)		=E2+E\$1	=ПИ()*E3	=SIN(F3)		=C3*G3	
4										
5										

Каждую из ячеек A3, B3, C3, E3, F3, G3 и I3 скопировали вниз во все ячейки столбцов, в которых они расположены вплоть до строки с номером 402. Затем выделили диапазон I2:I402 и построили график, который приведен ниже:



Определите и запишите в ответ целое положительное число, которое записано в ячейке D1.

Примечание. функция ПИ() возвращает значение числа Пи, округленное до 15-го знака после запятой.

3. Базы данных (2 балла)

[Цепочки поставок]

База данных промышленного предприятия содержит, в том числе, две таблицы. В таблице «Сборка» хранятся все данные о составе некоторого изделия. Изделие состоит из компонентов, каждый из которых в свою очередь также может состоять из других компонентов. Каждый компонент изделия имеет уникальный буквенный идентификатор. Каждая строка таблицы хранит запись о том, что некоторый составной компонент, в том числе само изделие «R» (столбец «Создаваемый компонент») требует для своего создания определенного количества (столбец «Количество») единиц другого компонента (столбец «Требуемый компонент»).

Создаваемый компонент	Требуемый компонент	Количество
R	A	1
R	B	2
A	D	2
B	G	3
A	C	5
B	E	7
B	F	8

Компоненты, идентификаторы которых ни разу не встречаются в столбце «Создаваемый компонент» являются элементарными и поставляются в готовом виде. Остальные компоненты являются составными и требуют изготовления.

В изготовлении изделия участвуют 7 компаний. Каждая компания имеет уникальный численный идентификатор. Окончательное изделие может изготавливать только компания с идентификатором «1». Остальные компании поставляют другим компаниям элементарные компоненты или изготавливают и поставляют другим компаниям составные компоненты. Таблица «Контракты» содержит данные обо всех контрактах на поставку компонентов. Каждая строка таблицы хранит запись об одном контракте в рамках которого компания-поставщик поставляет компании-заказчику определенное количество некоторых компонентов.

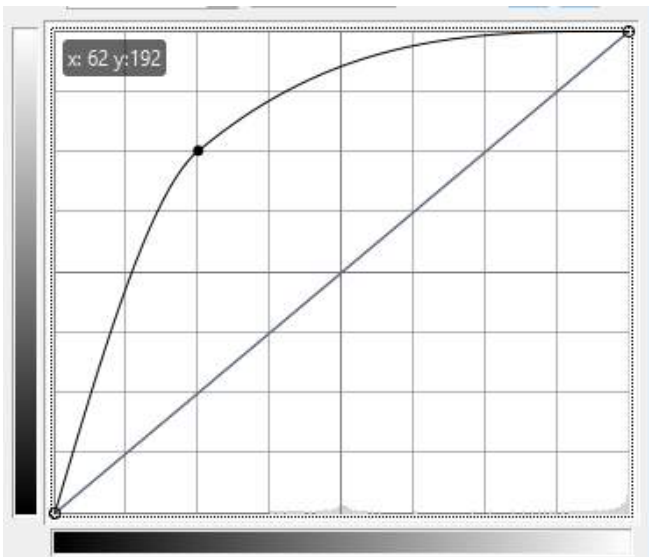
Поставщик	Заказчик	Компонент	Количество
2	1	A	5
3	1	B	6
2	1	A	8
6	3	G	11
6	2	D	13
3	1	B	14
2	1	A	15
7	2	D	15
7	2	D	19
3	1	B	22
4	2	C	31
6	3	G	31
5	2	C	32
5	3	F	34
7	3	E	42
4	2	C	54
5	3	F	61
4	3	G	87
4	3	E	115
7	3	E	137
6	3	F	226

Контракт на поставку составных компонентов может быть исполнен только, если компания-поставщик получила достаточное количество компонентов, из которых изготавливается поставляемый компонент. Контракт может быть выполнен только в полном объеме. Если компания-поставщик не может изготовить достаточное количество компонентов для выполнения контракта, контракт срывается. У каждой компании-поставщика все контракты отсортированы по убыванию количества единиц поставляемых компонентов. Компания-поставщик последовательно выполняет все контракты из своего списка. Если для очередного контракта из отсортированного списка недостаточно компонентов, то компания-поставщик пропускает его и пытается выполнить следующий в списке контракт и так пока не закончится список контрактов. Все контракты на поставку элементарных компонентов гарантированно выполнены в полном объеме до начала выполнения контрактов на составные компоненты. Если у компании остаются компоненты после выполнения некоторого контракта, она может их использовать при выполнении следующего контракта.

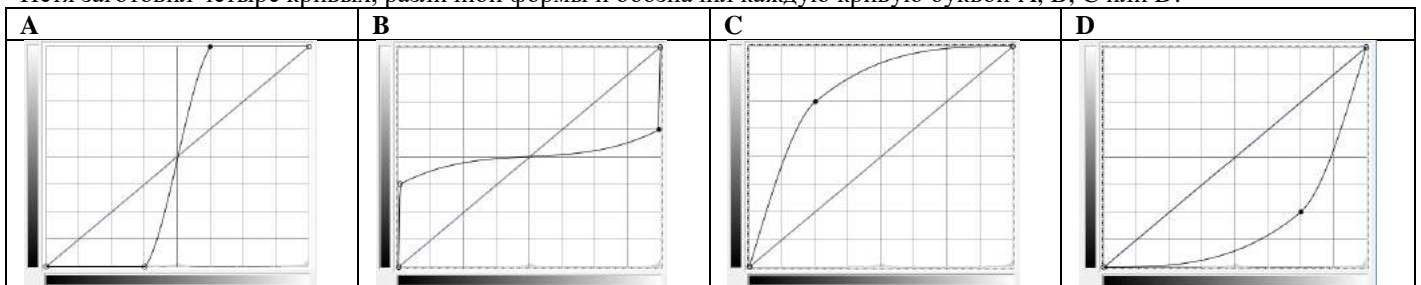
Определите, какое количество изделий «R» в результате будет произведено. В ответе укажите целое число.

4. Мультимедиа технологии (3 балла) [Кривые]

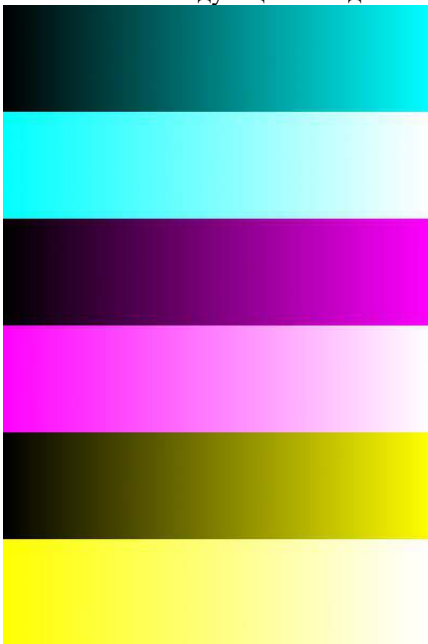
Одним из самых эффективных инструментов цветокоррекции является инструмент «Кривые» («Curves»). Инструмент отображает и позволяет изменять график, показывающей соотношение входной и выходной яркости всего изображения или отдельного цветового канала (например, в модели RGB – красного, зеленого или синего). По горизонтальной оси откладываются значения яркости исходного изображения, а по вертикальной оси – нового изображения, получаемого в результате цветокоррекции. Таким образом, в зависимости от формы кривой для всех точек с одинаковой яркостью по соответствующему каналу устанавливаются новые значения яркости. Например, в соответствии с кривой, представленной ниже, все точки, имевшие по некоторому каналу яркость 62, теперь будут иметь по этому каналу яркость 192 и т.д.



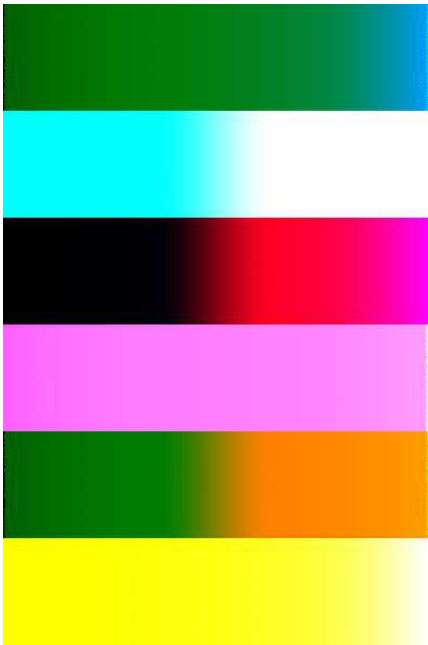
Петя заготовил четыре кривых, различной формы и обозначил каждую кривую буквой А, В, С или D:



Затем он взял следующее исходное изображение:



и выполнил его цветокоррекцию следующим образом. Он применил на каждом из цветовых каналов (красный, зеленый, синий) одну из заготовленных четырех кривых. Причем никакую кривую он не применил более одного раза. В результате он получил следующий результат:



Определите, к какому каналу Петя применил какую кривую. В ответе укажите подряд без пробелов последовательность из трех букв, обозначающих кривые: сначала букву, обозначающую кривую, примененную к красному каналу, затем букву, обозначающую кривую, примененную к зеленому каналу и затем букву, обозначающую кривую, примененную к синему каналу.

5. Телекоммуникационные технологии (1 балл).

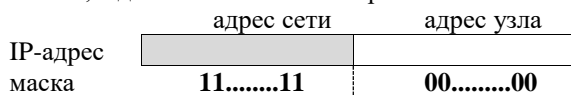
[Адрес сервера]

Адресация в Интернете:

Каждый компьютер, подключенный к сети Интернет, должен иметь собственный адрес, который называют IP-адресом (IP = *Internet Protocol*).

IP-адрес компьютера – это 32-битное число; для удобства его обычно записывают в виде четырёх чисел, разделенных точками; каждое из этих чисел находится в интервале 0...255, например: **192.168.128.210**.

IP-адрес состоит из двух частей: адреса сети и адреса узла в этой сети, причём деление адреса на части определяется маской – 32-битным числом, в двоичной записи которого сначала стоят единицы, а потом – нули:



Та часть IP-адреса, которая соответствует единичным битам маски, относится к адресу сети, а часть, соответствующая нулевым битам маски – это числовой адрес узла.

Префикс – другая форма записи маски – показывает количество единиц в маске, например, 192.168.128.0/24 показывает, что в сетевой части адреса 24 бита, а в узловой – $32-24=8$ бит.

Предположим, нам нужно разбить сеть на одинаковые по размеру подсети. Для этого надо понять, сколько бит выделить на подсети. Если у нас есть требование к размеру сети (чтобы в них вмещалось, как минимум, n узлов, например, 50), необходимо:

1. Найти минимальную степень двойки k , такую что $2^k \geq n+2$ (два адреса добавляются потому, что сеть ещё должна вместить адрес сети и адрес широковещания), и установить новый префикс $m=32-k$ (чтобы в узловой части осталось k бит для адресации устройств) ($2^6 \geq 50+2$, $m=32-6=26$).
2. В каждой такой сети будет $2^6 = 64$ адреса, а для адресации узлов будет доступно: $2^6-2=62$ адреса.
3. Таким образом, адреса сетей будут:

Порядковый номер сети	Адрес сети	Адрес первого узла	Адрес последнего узла	Адрес широковещательного адреса (последний в диапазоне)
1-я	192.168.128.0/26	192.168.128.1/26	192.168.128.62/26	192.168.128.63/26
2-я	192.168.128.64/26	192.168.128.65/26	192.168.128.126/26	192.168.128.127/26
3-я	192.168.128.128/26	192.168.128.129/26	192.168.128.190/26	192.168.128.191/26
4-я	192.168.128.192/26	192.168.128.193/26	192.168.128.254/26	192.168.128.255/26

Задача:

Вы – системный администратор компании. Вам дано адресное пространство исходной сети следующего вида – «10.10.18.x/24». Вам нужно настроить сервисы в локальной сети головного офиса, для чего необходимо разбить сеть на

подсети. Длину маски рассчитайте исходя из того, чтобы в каждой образовавшейся подсети можно было разместить ровно 14 узлов. В качестве адреса сервера необходимо взять первый возможный адрес в третьей по порядку подсети. В ответ запишите адрес сервера, например, 10.10.18.100/25

6. Операционные системы (2 балла)

[Маски файлов]

В ряде операционных систем для групповых операций с файлами используют маски имён файлов. Маска имени файла представляет собой последовательность, в которой могут встречаться символы, допустимые в именах файлов, и два специальных символа:

1. Символ "?" (вопросительный знак) означает ровно один произвольный допустимый символ.
2. Символ "*" (звездочка) означает любую последовательность допустимых символов, в том числе пустую последовательность.

Ниже представлен нумерованный список масок файлов:

1. *?*c?*?d?*.e?*
2. ?*c*d*?.?*
3. ?*c?*?d?*.??*
4. ?*c*d?*.?**
5. ?c?*?d?*.e?*
6. ?*c?*?d*?.*e*?

Расположите номера этих масок файлов в порядке невозрастающей последовательности количества имен файлов, которые будут выбираться каждой маской. Это означает, что для любого произвольного набора имен файлов количество имен файлов, которые будут выбраны первой маской из последовательности, будет больше или равно количеству имен файлов, которые будут выбраны второй маской из последовательности. Аналогично количество имен файлов, которые будут выбраны второй маской из последовательности, будет больше или равно количеству имен файлов, которые будут выбраны третьей маской из последовательности и т.д. В ответе укажите подряд без пробелов последовательность из шести чисел.

7. Технологии программирования (2 балла)

[Необычная сортировка]

Имя входного файла	input.txt
Имя выходного файла	output.txt
Ограничение по времени	1 секунда
Ограничение по памяти	256 мегабайт

Маленький мальчик Вася не умеет сортировать массивы. Поэтому, когда ему нужно отсортировать массив по неубыванию, он по очереди берет элементы из начала, а затем из конца массива, ставит их в новый.

На массиве {1, 2} этот алгоритм безупречно работает. Однако, на деле оказалось, что алгоритм работает далеко не всегда.

Вася не видит смысла пытаться сортировать массив, если заведомо известно, что ничего не выйдет. Ваша задача написать программу, которая по заданному массиву определит, можно ли его отсортировать Васиным алгоритмом или нет.

Формат входных данных

В первой строке задано одно число n - длина массива ($1 \leq n \leq 1000$). Во второй строке находятся n целых чисел a_1, a_2, \dots, a_n - элементы массива ($1 \leq a_i \leq 10^4$).

Формат выходных данных

Выведите YES, если массив можно отсортировать Васиным алгоритмом, и NO в противном случае.

Примеры

input.txt	output.txt
3 1 3 2	YES
2 1 1	YES
3 3 1 2	NO

8. Технологии программирования (2 балла)

[Музыкальный плейлист]

Имя входного файла	input.txt
Имя выходного файла	output.txt

Ограничение по времени	1 секунда
Ограничение по памяти	256 мегабайт

У Пети выдалось несколько свободных часов, которые он решил потратить на наведение порядка в своем музыкальном плейлисте. Плейлист Пети состоит из n песен, некоторые из которых могут повторяться. Так как слушать одну и ту же песню несколько раз за короткий промежуток времени не очень интересно, Петя решил принять меры по избежанию этого.

Он создал новый плейлист, в котором выделил слоты для n песен. После этого он разбил эти n слотов на группы размером k , начиная с начала: у него получились группы $[1..k]$, $[k+1..2*k]$, $[2*k+1..3*k]$, ... (последняя группа может иметь размер меньше k). После этого Петя решил, что в каждой из этих групп не должно быть повторяющихся песен, то есть все песни внутри одной группы должны быть попарно различны. Однако, после этого перед ним возник вопрос - как расположить песни из плейлиста так, чтобы они удовлетворяли его условию? С этим вопросом он обратился к вам, помогите ему!

Формат входных данных

В первой строке содержатся два числа n, k - количество песен в плейлисте и размер группы, на которые Петя разбивает свой плейлист ($1 \leq k \leq n \leq 1000$).

В следующей строке содержатся n чисел a_i - идентификаторы песен в плейлисте Пети ($1 \leq a_i \leq 1000$). Одинаковые идентификаторы означают одинаковые песни, разные идентификаторы - разные песни.

Формат выходных данных

Если расположить песни из плейлиста способом, удовлетворяющим требованию Пети невозможно, в единственной строке выведите «-1». В противном случае, в единственной строке через пробел выведите n чисел - новое расположение песен в плейлисте Пети.

Если существует несколько вариантов ответа, выведите любой.

Примеры

input.txt	output.txt
5 3 2 2 4 3 3	3 4 2 2 3
5 3 2 2 3 3 3	-1

Замечание

В первом тестовом примере Петя разбивает плейлист на две группы: первая состоит из первых трех песен, вторая из последних двух. Несложно заметить, что если расположить песни в порядке 3, 4, 2, 2, 3, в обеих группах все песни будут попарно различны.