

### III. Задания отборочного этапа олимпиады 2016-17 года Отборочный этап 11 класса. 1 тур (приведен один из вариантов заданий)

#### 1. Кодирование информации. Системы счисления (2 балла)

##### [Три основания]

Буратино загадал последовательность из трех цифр: X, Y и Z, и предложил Пьеро угадать ее, исходя из того, что равенство:

$$XYZ_K + XYZ_L + XYZ_M = 1000_{10}$$

верно для некоторых позиционных систем счисления с основаниями K, L и M при условии, что  $L=K+1$  и  $M=L+1$ .

Пьеро сказал, что такой последовательностью цифр может быть 238. При каком значении K его ответ можно считать верным? В ответе приведите целое число.

#### 2. Кодирование информации. Системы счисления (3 балла)

##### [Период]

Переведите дробь  $0,02_3$  из троичной в двоичную систему счисления. Если результатом будет периодическая дробь, запишите ее так, чтобы период имел минимальную длину, и от десятичного разделителя до начала периода было минимальное количество цифр. Пример записи ответа:  $0,0(1)$ .

#### 3. Кодирование информации. Системы счисления (1 балл)

##### [Параллельный интерфейс]

Буратино сконструировал 16-битный параллельный интерфейс для передачи данных. Интерфейс позволяет передавать четырехразрядные шестнадцатеричные числа. Каждое шестнадцатеричное число передается в виде набора из шестнадцати двоичных разрядов, являющихся результатом перевода шестнадцатеричного числа в двоичную систему счисления (при необходимости в старших разрядах передаются незначащие нули). Нумерация разрядов осуществляется слева направо с единицы, то есть старший разряд двоичной записи числа имеет номер 1, а младший – номер 16.

Для проверки работы интерфейса Буратино провел следующий эксперимент. Он взял некоторое исходное четырехразрядное шестнадцатеричное число и передал Пьеро по новому интерфейсу сначала его, а затем еще три числа, образованные побитовым циклическим сдвигом влево исходного числа на 1, 2 и 3 разряда соответственно.

Пьеро получил следующие числа:

$96DE_{16}$

$2BBD_{16}$

$517A_{16}$

$A4F4_{16}$

Посмотрев на эти числа, Пьеро понял, что интерфейс работает неверно, а именно – один разряд передается инвертированным (если передается 0, при получении разряд равен 1 и наоборот).

Определите, какое исходное число взял Буратино, и какой разряд передается неверно. В ответе укажите через пробел сначала шестнадцатеричное число, а затем десятичное число – номер разряда.

#### 4. Кодирование информации. Количество информации (1 балл)

##### [Неравномерный код]

Кот Базилио разрабатывает программный компонент для сохранения текстовых данных. Известно, что в текстах используется алфавит из 32 букв и не содержится никаких других символов. Изначально Кот Базилио хранил в памяти для каждого символа его числовой код, используя для хранения каждого кода минимально возможное *одинаковое* для всех кодов количество бит. Лиса Алиса рассказала ему о том, что различные символы встречаются в тексте с разной частотой, и привела пример для конкретного текста размером в 500 символов:

Ранг	Буква	Количество в тексте
1	о	79
2	е	55
3	а	51
4	и	46
5	н	38
6	т	37
7	с	30
8	р	23
9	в	22
10	л	21
11	к	14
12	м	12

Ранг	Буква	Количество в тексте
17	ы	4
18	ь	3
19	г	4
20	з	4
21	б	5
22	ч	5
23	й	3
24	х	2
25	ж	2
26	ш	1
27	ю	1
28	ц	1

13	д	11
14	п	10
15	у	7
16	я	4

29	щ	1
30	э	1
31	ф	2
32	ь	1

Тогда Кот Базилио решил использовать следующий алгоритм построения неравномерных кодов символов:

6. Шаг1. В качестве текущего набора двоичных последовательностей взять все двухразрядные двоичные последовательности («00», «01», «10» и «11»).
7. Шаг2. Из текущего набора двоичных последовательностей те, которые закончились на «00» и «01», использовать как коды очередной пары символов из таблицы в порядке возрастания рангов символов (символу с меньшим рангом присвоить код, оканчивающийся на «00», а следующему за ним символу – код, оканчивающийся на «01»).
8. Шаг3. Из оставшихся двух двоичных последовательностей получить четыре новых последовательности длиной на один разряд больше исходных, добавляя к каждой из двух исходных последовательностей справа еще один новый разряд – 0 или 1 соответственно.
9. Шаг4. Если в таблице осталось больше четырех символов, которым не присвоен код, то считать полученные на предыдущем шаге четыре последовательности новым текущим набором и перейти на шаг 2. В противном случае полученные на предыдущем шаге четыре последовательности присвоить в качестве кодов последним четырём символам в таблице, так, что символу с большим рангом будет соответствовать последовательность, являющаяся двоичной записью большего числа, и завершить работу алгоритма.

Определите выигрыш по памяти, который получит Кот Базилио, при использовании нового метода кодирования символов для текста из 500 символов с количеством повторений каждого символа как указано в таблице. В ответе укажите целое число бит – разность между количеством бит, требуемым для хранения текста при изначальном варианте кодирования, и при кодировании по новому алгоритму.

## 5. Кодирование информации. Объем данных (2 балла)

### [Квадраты]

Арлекино нужно сохранить в памяти компьютера растровое изображение размером 2048 на 1536 пикселей. Изначально он попробовал сохранять изображение, применяя простой способ (вариант А): в памяти сохранялись коды цветов каждого пикселя с использованием 24-х битной цветовой модели RGB. Сжатия не производилось, и никакой дополнительной информации не хранилось.

Поняв, что при таком способе изображение занимает слишком много памяти, Арлекино решил проанализировать цвета пикселей и обнаружил следующую закономерность: если разбить изображение на непересекающиеся квадраты, размером 256 на 256 пикселей каждый, то максимальное количество различных цветов, которое может встретиться в одном из квадратов, равно 1024. Если же разбить изображение на непересекающиеся квадраты, размером 128 на 128 пикселей каждый, то максимальное количество различных цветов, которое может встретиться в одном из квадратов, равно 512. Тогда Арлекино решил применить следующий способ записи:

1. Разбить изображение на непересекающиеся квадраты, размером  $N$  на  $N$  пикселей.
2. Сохранить для каждого квадрата независимо коды цветов его пикселей, используя для их кодирования минимально возможное одинаковое для всех пикселей всех квадратов количество бит, рассчитанное исходя из того, какое максимальное количество цветов  $K$  может встретиться в каком-либо из получившихся квадратов.
3. После каждого квадрата записать его кодovou таблицу – последовательность из  $K$  24-х битных кодов. Размеры кодовых таблиц, записываемых после каждого квадрата, одинаковые, даже если в конкретном квадрате встречается меньше чем  $K$  цветов.
4. Никакой дополнительной информации не записывается.

Арлекино применил новый способ, разбив изображение на квадраты, размером 256 на 256 пикселей (вариант Б) и размером 128 на 128 пикселей (вариант В). Определите, в каком из вариантов получится меньший объем занимаемой изображением памяти. В ответе укажите сначала букву, обозначающую вариант, а затем через пробел целое число – объем занимаемой изображением памяти в КБайт (1 КБайт=1024 байта).

## 6. Основы логики. Упрощение логического выражения (2 балла)

### [Эквивалентности]

Упростите логическое выражение или укажите его результат (при его однозначности). Результат упрощения может содержать только операции инверсии, конъюнкции и дизъюнкции.

$$(A \leftrightarrow B \text{ and } C) \rightarrow ((B \leftrightarrow A \text{ and } C) \rightarrow (C \leftrightarrow A \text{ and } B))$$

*Комментарий по вводу ответа: операнды вводятся большими латинскими буквами; логические операции обозначаются, соответственно как not, and и or.*

*Скобки используются только для изменения порядка выполнения операций. Если порядок выполнения операций очевиден из их приоритетов – дополнительное использование скобок считается ошибкой.*

*При однозначном ответе – истинный ответ обозначается как 1, а ложный как 0.*

*Пример записи ответа: (A or not B) and C*

## 7. Основы логики. Синтез выражения по таблице истинности (2 балла)

### [Необычные столбцы]

Найдите логическую функцию  $F(A,B,C,D)$  для которой будет верна следующая таблица истинности:

$A \rightarrow B$	$\text{not } B \rightarrow C$	$C \leftrightarrow D$	$F(A,B,C,D)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

В ответе укажите формулу, которая может содержать логические переменные A, B, C и D или некоторые из них и не более чем три логические операции. В качестве логических операций могут использоваться только операции отрицания, конъюнкции или дизъюнкции. Запись формулы может содержать скобки.

*Комментарий по вводу ответа: операнды вводятся большими латинскими буквами; логические операции обозначаются, соответственно как **not**, **and** и **or**.*

*Пример записи ответа:  $\text{not } (A \text{ and } B)$ .*

## 8. Алгоритмизация и программирование. Формальный исполнитель (1 балл)

### [Бесконечный цикл]

В строке могут встречаться только символы "A" и "B". В качестве исходной строки взята последовательность "AABVVVAA".

Начальное значение  $N=0$ .

Выполняется следующий алгоритм обработки данных:

1. В имеющейся строке заменить каждый символ на пару таких же символов.
2. Пока в строке остается хотя бы одна последовательность из идущих подряд 5 или более одинаковых символов выполнить следующие действия:
  - a. Двигаясь слева направо заменять встретившиеся последовательности из идущих подряд ровно 5 одинаковых символов на один другой символ ("AAAAA" заменяется на "B", а "BBBBB" заменяется на "A"), пока не достигнут конец строки.
  - b. Вернуться к началу строки.
3. Увеличить на 1 значение  $N$  и перейти на шаг 1.

Как выглядела строка в момент времени, когда  $N$  стало равно 1000? В ответе укажите подряд без пробелов последовательность из латинских букв A и B.

## 9. Алгоритмизация и программирование. Планирование (3 балла)

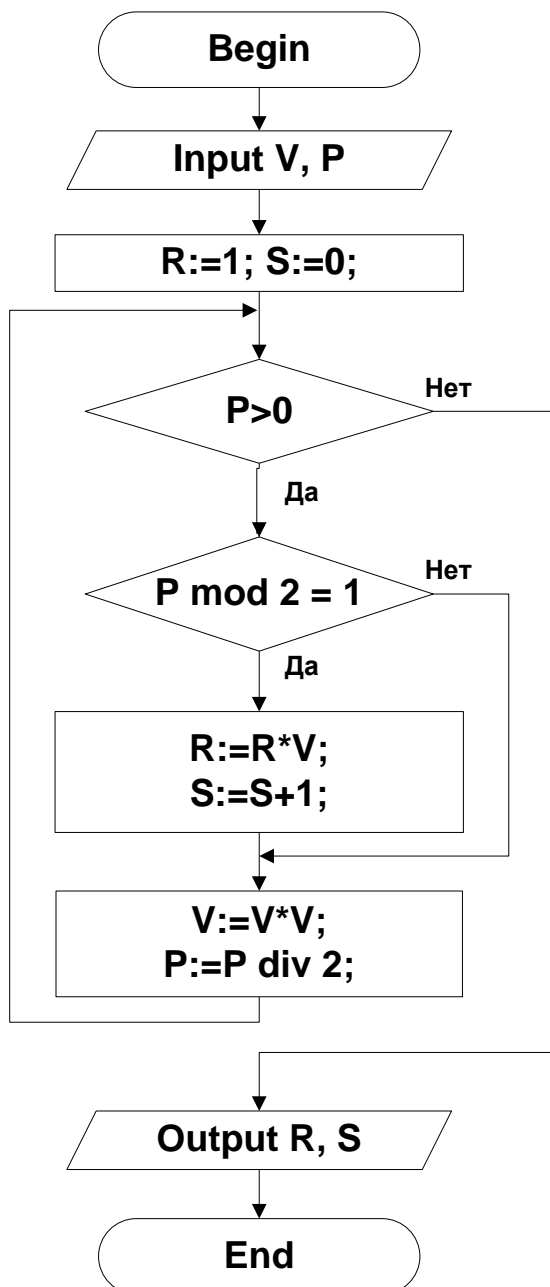
### [Разделение труда]

Арлекино и Пьеро работают программистами и вдвоем пишут одну программу. Они решили, что будут писать код по очереди блоками по  $N$  строчек кода. Когда любой программист (Арлекино или Пьеро) пишет очередной блок, со временем он устает, и скорость его работы уменьшается следующим образом: после каждых 15 строчек кода время, затрачиваемое на очередную строчку кода, увеличивается на 1 минуту (начальная производительность труда – 1 строчка кода в минуту). Таким образом, на каждую из первых 15 строчек кода очередного блока затрачивается по 1 минуте; на каждую из последующих 15 строчек – на 1 минуту больше, то есть по 2 минуты на строчку кода, на каждую из последующих 15 строчек – еще на 1 минуту больше, то есть по 3 минуты на строчку и т.д. Когда Арлекино или Пьеро заканчивает писать свой блок из  $N$  строчек кода, он уходит отдыхать и передает работу своему партнеру. Начинает работать над очередным блоком любой программист отдохнувшим (то есть начальная производительность труда опять 1 строчка кода в минуту), но перед тем как написать свою первую строчку кода в этом блоке, он должен изучить код, который перед этим написал его партнер, на что всегда затрачивает одно и то же время – 20 минут. Программист, который пишет самый первый блок программы, естественно, не тратит этого времени и сразу начинает писать первую строчку кода. Определите оптимальную длину блока  $N$ , при которой суммарное время, которое затратят программисты на написание программы из 500 строчек кода, окажется минимальным. В ответе укажите через пробел два целых числа: сначала оптимальное количество строчек кода в блоке, а затем суммарное время в минутах, которое затратят Арлекино и Пьеро на написание программы.

## 10. Алгоритмизация и программирование. Блок-схема, обратная задача (1 балл)

### [Умножитель]

Дана блок-схема алгоритма. V, P, R и S - целочисленные неотрицательные переменные.



Известно, что если после завершения алгоритма записать результаты его выполнения в девятеричной системе счисления, то получится  $R=1000000_9$  и  $S=2_9$ .

Перечислите через пробел в порядке возрастания в десятичной системе счисления целые значения, которые можно присвоить перед началом алгоритма переменной  $P$ , чтобы получить такой результат.

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

#### 1. Электронные таблицы. Адресация ячеек и вычисления (2 балла)

##### [Суммирование]

На рисунке представлен фрагмент электронной таблицы.

	A	B	C	D	E	F	G
1	1	2	3				
2	5	7	11				
3	13	17	19				
4	6						
5							
6							
7							
8							
9							
10			68				
11							
12							

В ячейке A4 находится формула следующего вида:

`=#A#1+#B#1+#C#1`

в которой некоторые символы # заменили на символ \$, а остальные удалили.

Ячейку A4 скопировали во все ячейки диапазона A4:C10.

Определите, какие символы # для формулы в ячейке A4 были заменены на \$, а какие удалены.

В ответе укажите подряд без пробелов последовательность из нулей и единиц, длиной 6 символов. Каждый символ в последовательности соответствует одному из символов # в приведенной записи формулы, считая слева направо. При этом 1 в последовательности будет означать, что соответствующий символ # был заменен на символ \$, а 0 будет означать, что соответствующий символ # был удален.

Например, последовательность 011001 будет означать, что в ячейке A4 перед ее копированием находилась формула:

`=A$1+$B1+C$1`

## 2. Электронные таблицы. Графики и диаграммы (1 балл)

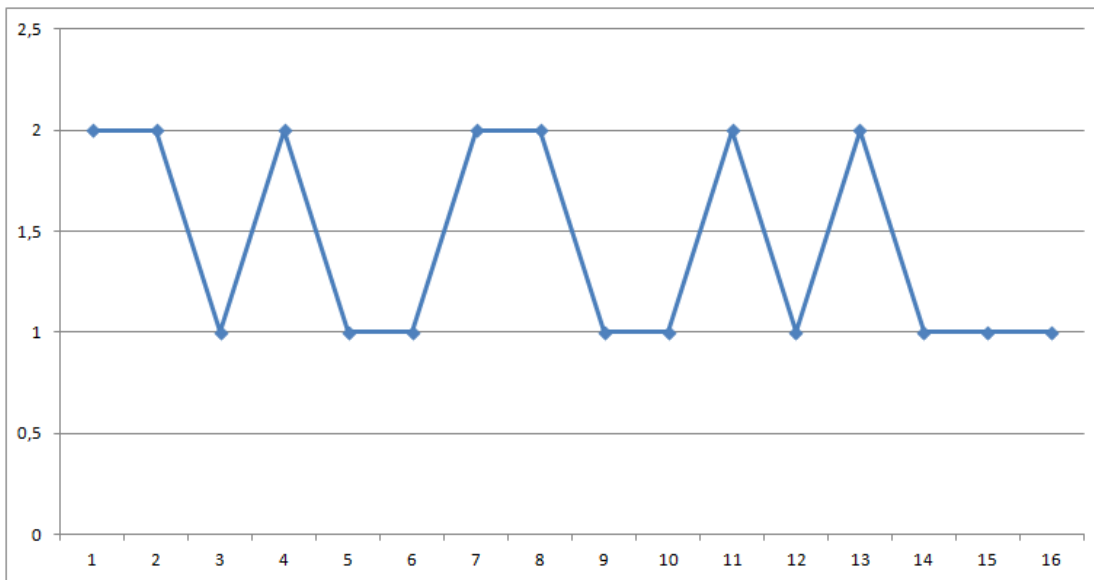
[Ломаная]

На рисунке представлен фрагмент электронной таблицы в режиме отображения формул.

	A	B	C	D	E	F	G
1	37325		36636	11722			
2	0	=ОСТАТ(ЧАСТНОЕ(B\$1;СТЕПЕНЬ(2;\$A2));2)				=ЕСЛИ(ИЛИ(И(B2=1;НЕ(C2=1));И(D2=1;НЕ(E2=1)));2;1)	
3	1						
4	2						
5	3						
6	4						
7	5						
8	6						
9	7						
10	8						
11	9						
12	10						
13	11						
14	12						
15	13						
16	14						
17	15						
18							
19							

Ячейку B2 скопировали во все ячейки диапазона B2:E17, а ячейку F2 скопировали во все ячейки диапазона F3:F17.

Какое минимальное целое положительное число необходимо поместить в ячейку E1 для того, чтобы график, построенный по диапазону F2:F17, выглядел следующим образом:



В ответе укажите целое число.

### 3. Сортировка и фильтрация данных (3 балла) [Инверсии]

Дана таблица из базы данных товаров:

Наименование	Признак 1	Признак 2	Признак 3	Признак 4
Товар 1	1	1	1	1
Товар 2	13	9	7	6
Товар 3	2	17	13	11
Товар 4	14	2	19	16
Товар 5	3	10	2	21
Товар 6	15	18	8	2
Товар 7	4	3	14	7
Товар 8	16	11	20	12
Товар 9	5	19	3	17
Товар 10	17	4	9	22
Товар 11	6	12	15	3
Товар 12	18	20	21	8
Товар 13	7	5	4	13
Товар 14	19	13	10	18
Товар 15	8	21	16	23
Товар 16	20	6	22	4
Товар 17	9	14	5	9
Товар 18	21	22	11	14
Товар 19	10	7	17	19
Товар 20	22	15	23	24
Товар 21	11	23	6	5
Товар 22	23	8	12	10
Товар 23	12	16	18	15
Товар 24	24	24	24	20

Строки этой таблицы можно отсортировать по возрастанию значений в одном из столбцов признаков (строки при сортировке переставляются целиком).

Назовем инверсией в столбце «Наименование» ситуацию, когда товар с большим номером стоит выше товара с меньшим номером.

Количество инверсий в столбце «Наименование» – это количество пар строк (не обязательно соседних), в которых строка с товаром с большим номером стоит выше строки с товаром с меньшим номером.

Например, в приведенной ниже таблице количество инверсий в столбце «Наименование» равно 5:

Наименование	Признак 1
Товар 3	1
Товар 4	2
Товар 2	3
Товар 1	4

Определите, по какому одному столбцу с признаком необходимо провести сортировку в порядке возрастания значений в этом столбце, чтобы количество инверсий в столбце «Наименование» было **максимальным**. В ответе укажите через пробел два числа: сначала номер признака, а затем количество инверсий в столбце «Наименование», которое получится при сортировке по столбцу с этим признаком.

#### 4. Телекоммуникационные технологии (1 балл)

##### [Балансировка канала]

Три клиентских приложения, K1, K2 и K3, совместно используют канал передачи данных для загрузки файлов. Система балансировки нагрузки в каждый момент времени обеспечивает равную скорость передачи данных каждому клиентскому приложению так, что сумма скоростей передачи данных всем клиентским приложениям всегда равна 6 МБайт в секунду. Например, если в данный момент времени два клиентских приложения используют канал передачи данных, а третье неактивно, то скорость передачи данных каждому из двух активных клиентских приложений будет равна 3 МБайт в секунду.

В начальный момент времени приложение K1 начало загружать файл размером 240 МБайт. В этот момент оба других клиентских приложения были неактивны.

Через 16 секунд от начального момента времени приложения K2 и K3 одновременно начали загружать по одному файлу. Размер файла, который загружает приложение K2 составляет 192 МБайт.

Известно, что приложение K1 загрузило файл за 80 секунд, а приложение K2 за 72 секунды.

Определите размер файла в МБайт, который загрузило приложение K3.

Кроме упомянутых загружаемых файлов никакой другой нагрузки на канал передачи данных нет.

В ответе укажите целое число.

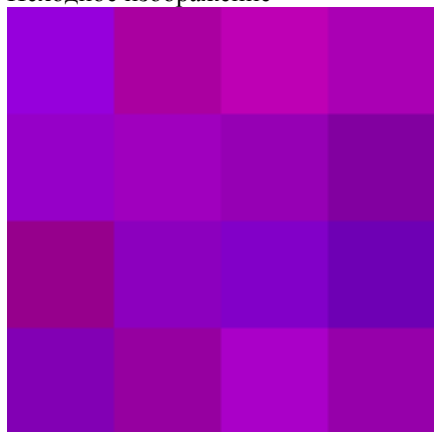
#### 5. Мультимедиа технологии (2 балла)

##### Внимание на вводе изображения вставлять приложенными файлами

##### [Волшебная палочка]

Дано растровое изображение в 24-х битной цветовой модели RGB, разбитое на 16 квадратов, каждый из которых залит одним цветом. Каждый квадрат обозначен латинской буквой, как показано на рисунке справа.

Исходное изображение



Цвета квадратов приведены в таблице:

	R	G	B
A	150	0	220
B	170	0	160
C	190	0	180
D	170	0	180
E	150	0	200
F	160	0	190
G	150	0	180
H	130	0	160

I	150	0	140
J	140	0	190
K	130	0	200
L	110	0	180
M	130	0	180
N	150	0	160
O	170	0	200
P	150	0	170

Цветовая модель RGB может быть представлена в виде трехмерного пространства с прямоугольной системой координат и осями R, G и B соответственно. Тогда цвет любого пикселя может быть определен как точка в этом трехмерном пространстве.

Инструмент «волшебная палочка» в большинстве графических редакторов работает следующим образом. У инструмента есть один параметр T – чувствительность. Пользователь применяет инструмент к одному из пикселей изображения – исходному пикселю. Считываются цветовые координаты исходного пикселя (значения R, G и B для цвета этого пикселя) и тем самым определяется точка в пространстве RGB. Строится шар радиусом T с центром в этой точке. Затем выделяются все пиксели изображения, для которых выполняется следующая пара условий:

1. Точка в пространстве RGB, соответствующая цвету этого пикселя, находится внутри или на границе построенного шара.
2. Между этим пикселем и исходным пикселем можно построить путь, проходящий через смежные (имеющие общую границу) пиксели, для которых точки пространства RGB также находятся внутри или на границе построенного шара.

Определите минимальное значение параметра T, такое, что существует хотя бы один пиксель на исходном изображении, применение к которому инструмента «волшебная палочка» приведёт к выделению всех пикселей этого изображения. В ответе укажите через пробел сначала латинскую букву, обозначающую квадрат, содержащий этот пиксель к которому нужно применить инструмент, а затем целое число - найденное минимальное значение T.

Примечание. В задании описано стандартное поведение инструмента «Волшебная палочка» без опций сглаживания и других дополнительных опций.

## 6. Операционные системы (2 балла)

### [Читатели и писатели]

Хранилище данных может обрабатывать запросы на чтение и запись данных в соответствии со следующими принципами:

3. Запросы на чтение данных могут выполняться параллельно. Если в момент появления запроса на чтение данных хранилище не занято обработкой какого-либо запроса, поступивший запрос начинает немедленно выполняться. Если в момент появления запроса на чтение данных хранилище обрабатывает один или несколько других запросов на чтение данных, вновь поступивший запрос также начинает немедленно выполняться. При этом количество одновременно выполняющихся запросов на чтение данных не влияет на время выполнения отдельного запроса.
4. Запросы на запись данных могут выполняться только отдельно. Если в момент появления запроса на запись данных хранилище не занято обработкой какого-либо запроса, поступивший запрос начинает немедленно выполняться. Если в момент появления запроса на запись данных хранилище обрабатывает другой запрос на запись данных или запрос(ы) на чтение данных, поступивший запрос на запись данных помещается в конец очереди запросов на запись данных.
5. Если запрос на чтение данных появляется в момент времени, когда хранилище обрабатывает запрос на запись данных, этот запрос помещается в очередь запросов на чтение данных.
6. Если хранилище закончило обрабатывать запрос на запись данных, но в очереди остались еще запросы на запись данных или в этот момент времени появился новый запрос на запись данных, будет обрабатываться запрос на запись данных, независимо от наличия очереди из запросов на чтение данных.
7. Если хранилище закончило обрабатывать запрос на запись данных, в очереди нет больше запросов на запись данных и в этот момент времени не появился новый запрос на запись данных, начнут параллельно обрабатываться все имеющиеся в очереди и/или появившиеся в этот момент запросы на чтение данных.
8. Если в один и тот же момент появились запросы на чтение и на запись данных, и нет обрабатываемых в этот момент времени других запросов, будет выполняться запрос на запись данных, а запрос на чтение данных будет помещен в очередь.
9. Если запрос на чтение данных появляется в момент времени, когда хранилище не обрабатывает никаких запросов, но есть невыполненные запросы в очереди запросов на запись данных, то начнет обрабатываться очередной запрос из очереди запросов на запись данных, а запрос на чтение данных будет помещен в очередь запросов на чтение данных.

Существует 3 источника требований к хранилищу данных:

Источник 1. Отправляет запросы на чтение данных. Первый запрос он отправляет в начальный момент времени. По истечении 6 секунд после этого он отправляет второй запрос, и затем по истечении каждых 6 секунд этот источник отправляет очередной запрос. Обработка каждого запроса от этого источника занимает 4 секунды.



Источник 2. Отправляет запросы на чтение данных. Первый запрос он отправляет по истечении 3 секунд от начального момента времени. По истечении 5 секунд после этого он отправляет второй запрос, и затем по истечении каждые 5 секунд этот источник отправляет очередной запрос. Обработка каждого запроса от этого источника занимает 4 секунды.

Источник 3. Отправляет запросы на запись данных. Первый запрос он отправляет по истечении 6 секунд от начального момента времени. По истечении 10 секунд после этого он отправляет второй запрос, и затем по истечении каждые 10 секунд этот источник отправляет очередной запрос. Обработка каждого запроса этого источника занимает 6 секунд.

Через сколько секунд от начального момента будет завершена обработка 5-го запроса на запись данных?

В ответе укажите целое число.

**Ответ: 54**

## 7. Технологии программирования (3 балла)

Сегодня на уроке класс Васи проходил различные алгоритмы кодирования данных. Однако, уже придуманные алгоритмы кодирования его не заинтересовали, и он решил придумать свой собственный. Первый метод, который пришел ему в голову, выглядел так: каждый символ строки, состоящей из латинских строчных символов, кодируется числом от 1 до 26 в алфавитном порядке (символ 'a' кодируется числом 1, символ 'b' — числом 2, и т.д.), а затем все эти числа записываются в одну строку подряд без пробелов. Например, строка **abza** будет закодирована следующим образом: **12261**.

Все бы ничего, но Васи метод оказался не очень эффективен — полученное закодированное сообщение не всегда можно единственным образом декодировать. Однако, Вася решил, что это не такая большая проблема — вместо этого он решил по полученному коду восстанавливать строку минимальной возможной длины. Если таких строк несколько, ему все равно, какую из них он найдет. Помогите ему с этой задачей.

### Формат входного файла

В первой строке входного файла `input.txt` записана строка, состоящая из цифр. Ее длина не превосходит 100.

Гарантируется, что строка получена в результате применения Васиного алгоритма кодирования к некоторой строке, состоящей только из строчных латинских букв.

### Формат выходного файла

В выходной файл `output.txt` требуется вывести раскодированную строку — строку, после применения к которой алгоритма Васи, получается строка, данная во входном файле. Из всех возможных вариантов таких строк, строка в ответе должна иметь минимальную возможную длину. Если строк минимальной длины несколько, разрешается вывести любую их них.

### Пример входных и выходных данных

input.txt	output.txt	Комментарий
219	bs	Символ 'b' кодируется в число 2, а символ 's' в число 19. Также правильным ответом является строка "ui".
271	bga	Других вариантов декодирования нет.

## 8. Технологии программирования (4 балла)

Вася с его сестрой Леной увидели на окне **n** горшков с цветами, которые стоят на **n** позициях, пронумерованных от 1 до **n**. В каждом горшке растет либо роза, либо фиалка.

Ребята захотели расставить горшки с цветами таким образом, чтобы каждому из них расстановка показалась красивой. Но у них разное представление о красоте: Васе понравится расстановка, если на нечетных позициях будут стоять розы, а Ане — если на четных позициях будут стоять фиалки.

Ребята решили действовать сообща, а именно: они поочередно будут менять местами горшки на позициях **i** и **j**, причем **i** и **j** ребята каждый раз выбирают на свое усмотрение.

Помогите ребятам: вычислите, какие действия им нужно совершать.

### Формат входного файла

В первой строке входного файла `input.txt` записано натуральное число **n**, не превосходящее 1000. В следующей строке находятся **n** чисел, **i**-е равно либо 0 — это означает, что на данной позиции стоит роза, либо 1 — это означает, что на данной позиции стоит фиалка.

### Формат выходного файла

Первая строка выходного файла `output.txt` должна содержать число **-1** если ребятам не удастся достичь расстановки, которая понравится им обоим, либо число **k** — неотрицательное и не превосходящее **n**, равное количеству действий, которое необходимо совершить ребятам, чтобы достичь расстановки, которая им обоим понравится. В следующих **k** строках нужно вывести пары **i** и **j** ( $1 \leq i \leq n, 1 \leq j \leq n$ ) — действия, которые необходимо совершить ребятам. Первое действие делает Вася, затем Лена, и так поочередно.

### Пример входных и выходных данных

input.txt	output.txt
4 0 1 0 1	0
5 1 0 0 0 1	2 5 1 2
3 1 1 1	-1