

## Заключительный этап для 9 класса (приведен один из вариантов заданий)

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

#### [Много нулей]

Дано арифметическое выражение:

$$11_{16} \cdot (20_{16}^N + 1)^2 = X_2$$

Найдите минимальное целое положительное значение  $N$ , при котором в двоичной записи числа  $X$  будет более 1000 значащих нулей. Постройте двоичную запись числа  $X$  для найденного значения  $N$  и найдите в ней последовательность из идущих подряд нулей максимальной длины. Укажите в ответе длину найденной последовательности в виде числа в десятичной системе счисления.

**Ответ: 501**

**Решение:**

Обозначим за  $\{0\}_K$  последовательность из  $K$  идущих подряд нулей. Тогда  $20_{16}^N + 1 = 100000_2^N + 1$  можно записать как  $1\{0\}_{(5N-1)}1$ . Возведение этого числа в квадрат равносильно умножению его само на себя:  $1\{0\}_{(5N-1)}1 * 1\{0\}_{(5N-1)}1 = 1\{0\}_{(5N-1)}1 * 1\{0\}_{(5N)} + 1\{0\}_{(5N-1)}1 = 1\{0\}_{(5N-1)}1\{0\}_{(5N)} + 1\{0\}_{(5N-1)}1 = 1\{0\}_{(5N-2)}1\{0\}_{(5N)}1$ . Для получения значения  $X$  нужно еще умножить на  $11_{16} = 10001_2$ .  $1\{0\}_{(5N-2)}1\{0\}_{(5N)}1 * 10001 = 1\{0\}_{(5N-2)}1\{0\}_{(5N)}10000 + 1\{0\}_{(5N-2)}1\{0\}_{(5N)}1$ . Для любого  $N > 1$  в результате получится:  $10001\{0\}_{(5N-6)}10001\{0\}_{(5N-4)}1001$ . Обратим внимание, что общее количество значащих нулей в записи будет  $3 + (5N-6) + 3 + (5N-4) + 3 = 10N-1$ , а максимальное количество идущих подряд нулей будет  $5N-4$ . Тогда минимальным  $N$ , при котором общее число нулей в записи числа  $X$  будет более 1000 является  $N=101$ . Следовательно, длина максимальной последовательности идущих подряд нулей будет равна  $5 * 101 - 4 = 501$

### 2. Кодирование информации. Объем информации (1 балл)

#### [Моноширинный шрифт]

Петя пытается переписать прошивку детского планшета для рисования, чтобы использовать его как электронную книгу. Детский планшет умеет выводить черно-белые изображения (каждый пиксель или черный, или белый), размером 640 на 480 пикселей. Петя написал программу, которая формирует из текста отдельные страницы книги так, что на одной странице 60 строк по 80 символов, преобразует такую страницу в изображение, требуемого для планшета размера и для каждой страницы сохраняет в памяти коды отдельных пикселей получившегося изображения, используя один бит на код пикселя (метод А). Петя быстро понял, что при таком подходе в память планшета помещается очень мало страниц. Тогда он задумался, что в тексте, который он хочет выводить, встречается не так уж много различных символов. Он решил использовать моноширинный шрифт. В этом случае все изображение разбивается на непересекающиеся знакоместа, размером 8 на 8 пикселей и каждый символ отображается в своём знакоместе. Петя сохранил в памяти планшета изображения знакомест для всех возможных символов и теперь для каждой страницы хранит только коды символов для всех знакомест страницы (метод Б), а уже программа на планшете формирует из изображений отдельных букв окончательную картинку и выводит её. Коды хранятся для всех символов: даже, если в некотором знакоместе не выводится символ, хранится код пробела. Для хранения каждого кода символа Петя отводит в памяти минимально возможное, одинаковое для кодов всех символов количество бит. Петя обнаружил, что в случае использования метода Б на одну страницу тратится на 34800 байт меньше памяти, чем при использовании метода А. Определите максимальное количество символов в алфавите текста, при котором это возможно.

**Ответ: 64**

**Решение:**

Определим, сколько памяти в байтах потребуется для хранения страницы по методу А. Это будет  $(640 * 480 * 1) / 8 = 38400$  байт. Теперь запишем формулу для вычисления объема памяти, требуемой для хранения страниц по методу Б:  $60 * 80 * \log_2(X)$ . При этом значение  $\log_2(X)$  округляется до ближайшего большего целого числа. Значит, поскольку нас спрашивают про максимальное значение, можем считать, что нас интересует такое значение  $X$ , которое будет целой степенью 2, поскольку тогда его не нужно будет округлять вверх. Также отметим, что формула для определения информационного объема для страницы, сохраненной по методу Б, даёт ответ в битах. Тогда можно построить соотношение:

$$38400 - 60 * 80 * \log_2(X) / 8 = 34800$$

Получается, что  $\log_2(X) = (38400 - 34800) * 8 / (60 * 80) = 6$ . Тогда максимальное значение  $X = 2^6 = 64$

### 3. Основы логики (3 балла)

#### [Из чего это следует?]

Две логические функции четырех переменных будут считаться не эквивалентными, если существует хотя бы одна комбинация значений переменных  $A, B, C$  и  $D$ , для которых логические функции принимают различные значения.

Сколько существует не эквивалентных друг другу логических функций от четырех переменных  $F(A,B,C,D)$  таких, что будут одновременно истинны два логических выражения:

$$F(A, B, C, D) \rightarrow ((A \wedge \bar{B}) \leftrightarrow (C \wedge \bar{D})) = \text{Истина}$$

$$F(A, B, C, D) \rightarrow ((\bar{A} \wedge B) \leftrightarrow (\bar{C} \wedge D)) = \text{Истина}$$

В ответе укажите целое число.

**Ответ: 64**

**Решение:**

Вручную или с использованием программирования построим таблицы истинности для правых частей обеих импликаций (ноль означает ложное значение, а единица – истинное):

A	B	C	D	$((A \wedge \bar{B}) \leftrightarrow (C \wedge \bar{D}))$	$((\bar{A} \wedge B) \leftrightarrow (\bar{C} \wedge D))$
0	0	0	0	1	1
0	0	0	1	1	0
0	0	1	0	0	1
0	0	1	1	1	1
0	1	0	0	1	0
0	1	0	1	1	1
0	1	1	0	0	0
0	1	1	1	1	0
1	0	0	0	0	1
1	0	0	1	0	0
1	0	1	0	1	1
1	0	1	1	0	1
1	1	0	0	1	1
1	1	0	1	1	0
1	1	1	0	0	1
1	1	1	1	1	1

Обратим внимание, что, если для некоторой комбинации значений  $A, B, C, D$  правая часть хотя бы одной импликации принимает ложное значение, функция  $F(A, B, C, D)$  также должна принимать ложное значение. В противном случае соответствующая импликация не сможет быть истинной. А вот для комбинаций значений  $A, B, C, D$ , для которых правая часть обеих импликаций истинна – функция  $F(A, B, C, D)$  может принимать любое значение. Обозначим его вопросительным знаком и построим столбец с возможными значениями функции  $F(A, B, C, D)$ :

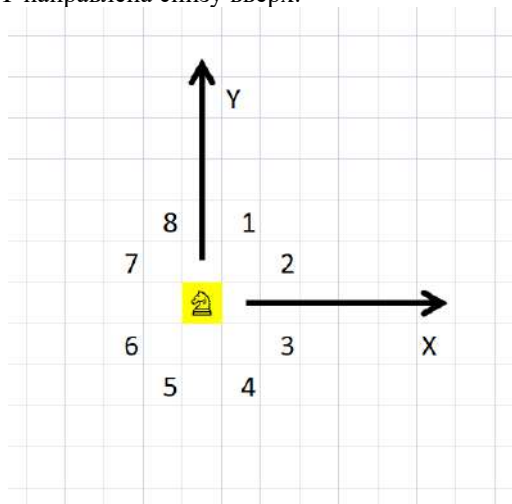
A	B	C	D	$((A \wedge \bar{B}) \leftrightarrow (C \wedge \bar{D}))$	$((\bar{A} \wedge B) \leftrightarrow (\bar{C} \wedge D))$	$F(A, B, C, D)$
0	0	0	0	1	1	?
0	0	0	1	1	0	0
0	0	1	0	0	1	0
0	0	1	1	1	1	?
0	1	0	0	1	0	0
0	1	0	1	1	1	?
0	1	1	0	0	0	0
0	1	1	1	1	0	0
1	0	0	0	0	1	0
1	0	0	1	0	0	0
1	0	1	0	1	1	?
1	0	1	1	0	1	0
1	1	0	0	1	1	?
1	1	0	1	1	0	0
1	1	1	0	0	1	0
1	1	1	1	1	1	?

Заметим, что функция может принимать любое значения для шести комбинаций переменных  $A, B, C, D$ . Следовательно, существует  $2^6=64$  таких не эквивалентных функций.

#### 4. Алгоритмизация и программирование. Формальные исполнители (1 балл)

##### [Ход конём]

Шахматный конь находится в центре клетчатого поля в клетке с координатами (0,0). Ось X направлена слева направо. Ось Y направлена снизу вверх.



Для управления конём есть 8 команд. На рисунке показаны клетки, в которые переместится конь из желтой клетки при выполнении команды с соответствующим номером.

Конь выполнил программу из 55 команд, про которую известно следующее:

1. Команд 1 в два раза больше, чем команд 8.
2. Команд 2 на 1 больше, чем команд 3.
3. Команд 3 на 2 больше, чем команд 8.
4. Команд 4 на 3 меньше, чем команд 2
5. Команд 5 на 2 больше, чем команд 6
6. Команд 6 на 2 меньше, чем команд 2
7. Команд 7 на 2 меньше, чем команд 2

Определите координаты клетки, в которой оказался конь, выполнив программу. В ответе укажите через пробел два числа: сначала координату X, а затем координату Y.

**Ответ: 8 5**

**Решение:**

Обозначим количество некоторых команд за X, например, количество команд 4. Тогда из условия 4 получим, что команд 2 будет (X+3). По условию 2 получим, что команд 3 будет (X+2). По условию 6 получим, что команд 6 будет (X+1). По условию 7 получим, что команд 7 будет (X+2). По условию 5 получим, что команд 5 будет (X+3). По условию 3 получим, что команд 8 будет X. Наконец, по условию 1 получим, что команд 1 будет X\*2.

Эти данные позволяют нам составить уравнение:

$$X*2+(X+3)+(X+2)+X+(X+3)+(X+1)+(X+1)+X=55.$$

Упростив, получим  $9*X = 45$ ,  $X=5$ .

Составим таблицу с количеством команд и смещением по каждой координате:

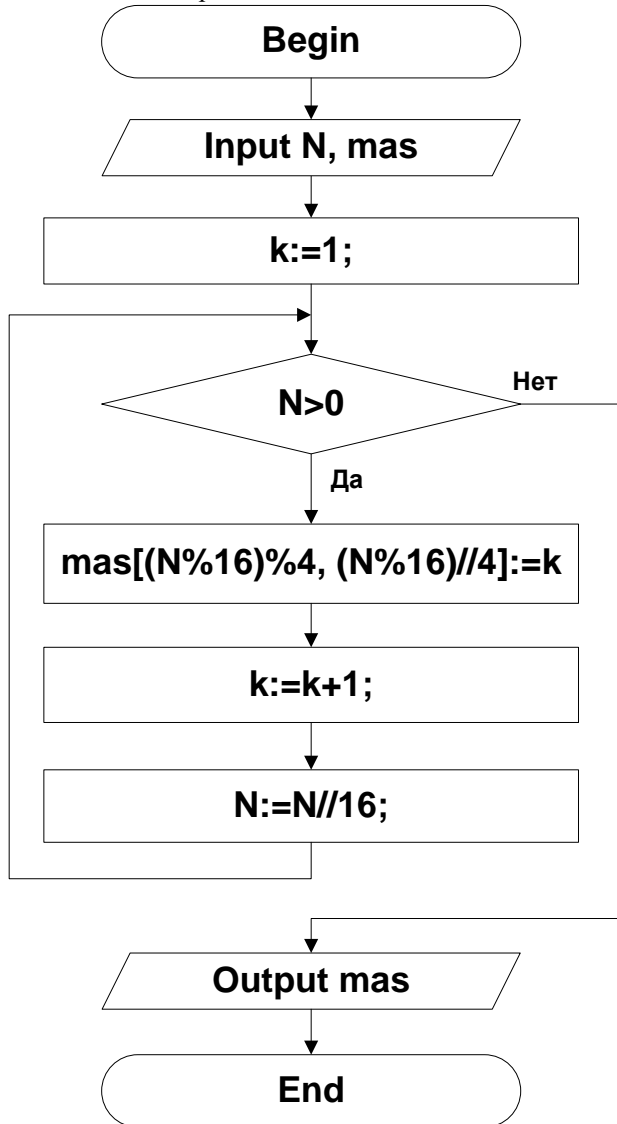
Команда	Количество команд	Смещение по X	Смещение по Y
1	X*2	10	20
2	X+3	8	8
3	X+2	7	-7
4	X	5	-10
5	X+3	8	-16
6	X+1	6	-6
7	X+1	6	6
8	X	5	10
Итого	55	8	5

Как видно, произошло смещение на 8 клеток по оси X и на 5 клеток по оси Y, то есть, начав в точке с координатами (0,0) конь, выполнив программу, оказался в клетке с координатами (8,5), то есть ответ «8 5».

5. Алгоритмизация и программирование. Анализ алгоритма, заданного в виде блок-схемы (3 балла)

[4\*4=16]

Дана блок схема алгоритма:



На вход алгоритма подали целое положительное число  $N$  и целочисленную матрицу  $mas$ , размером 4 на 4 элемента, заполненную нулями. На выходе получили следующую матрицу  $mas$ :

$$\begin{bmatrix} 14 & 9 & 4 & 7 \\ 13 & 5 & 2 & 1 \\ 6 & 10 & 8 & 16 \\ 15 & 12 & 3 & 11 \end{bmatrix}$$

Определите минимальное число,  $N$  для которого это возможно. Представьте запись числа  $N$  в двоичной системе счисления, сложите те цифры этой записи, номер которых (нумеруя цифры с 0 от младшего разряда к старшему) делится нацело на 3 и запишите результат в десятичной системе счисления. Например, если  $N=10101001_2$ , ответом будет 2.

Примечания. Операция  $A\%B$  вычисляет остаток от целочисленного деления  $A$  на  $B$ ; операция  $A//B$  вычисляет частное от целочисленного деления  $A$  на  $B$ . При обращении к элементам матрицы первый индекс – номер строки, а второй – номер столбца, нумерация с (0,0).

**Ответ: 12**

**Решение:**

Заметим, что на каждом шаге цикла происходит обращение к значению  $N\%16$ , а потом деление  $N//16$ . Это не что иное как разбор на отдельные цифры шестнадцатеричной записи числа  $N$ . То есть, на каждом шаге цикла берется цифра с номером  $k$  (считая справа налево от 1) в записи числа в шестнадцатеричной системе счисления. Вспомним, что любая шестнадцатеричная цифра может быть представлена как пара цифр для записи в четверичной системе счисления, например  $D_{16}=31_4$ . Тогда  $(N\%16)\%4$  – это младшая (вторая) цифра в такой паре, а  $(N\%16)//4$  – старшая (первая) цифра в такой паре. Тогда операция  $mas[(N\%16)\%4, (N\%16)//4]:=k$  означает, что в элемент матрицы, номер строки которого соответствует младшей цифре такой пары, а номер столбца – старшей цифры записывается номер соответствующей шестнадцатеричной цифры. Значит, мы можем брать последовательно элементы матрицы со значениями от 1 до 16, формировать для них пары (номер столбца, номер строки) и записывать такие пары справа налево (заметим, что нумерация идет от нуля):

32030001133312102230021120232131

Пользуясь тем, что любая четверичная цифра может быть представлена независимо как две двоичных, переведем в двоичную систему счисления и выделим разряды, номера которых (при нумерации с 0 от младшего к старшему разряду) делятся нацело на 3:

11100011000000101111101100100101000010010100010110011101

Посчитаем количество выделенных единиц – их будет ровно 12, что и является ответом на задание.

## 6. Телекоммуникационные технологии (2 балла)

### [Три порта]

Петя моделирует работу прототипа коммутатора. У этого прототипа есть три порта. Два порта принимают данные из двух сетей со скоростью 5 КБайт в секунду каждый. Третий порт отдает данные в третью сеть со скоростью 10 КБайт в секунду. Данные из первой сети приходят пакетами по 40 КБайт, причем первый пакет начинает приниматься в начальный момент времени и далее новый пакет начинает приниматься по истечении каждых 10 секунд. Данные из второй сети приходят пакетами по 30 КБайт, причем первый пакет начинает приниматься по истечении 4 секунд от начального момента времени и далее новый пакет начинает приниматься по истечении каждых 8 секунд.

Прототип коммутатора работает следующим образом. Поступающий пакет по мере передачи данных записывается в буферную память. Если одновременно принимаются пакеты в оба порта их запись в буферную память происходит одновременно и независимо. Как только некоторый пакет целиком принят, он становится готов к передаче через выходной порт. Передача данных через выходной порт происходит исходно полученными пакетами по 40 и по 30 КБайт соответственно, причем, если в некоторый момент времени есть возможность начать передавать пакеты обоих размеров, сначала передается пакет большего размера. В один момент времени через выходной порт может передаваться только один пакет и пакет всегда передается целиком. Передача пакета через выходной порт начинается сразу, как только появляется готовый пакет, если в этот момент не передается другой пакет или сразу после завершения передачи предыдущего пакета. Как только очередной пакет передан, он удаляется из буфера. Если в один момент времени нужно удалить переданный пакет и записать очередную порцию данных принимаемого пакета, сначала происходит удаление. Если вся память в буфере занята, но происходит попытка принять данные, происходит переполнение буфера и аварийная остановка коммутатора. Определите минимальный размер буфера в КБайтах, при котором в процессе работы данной модели никогда не произойдет аварийная остановка. В ответе укажите целое число.

**Ответ: 90**

**Решение:**

Вручную или с помощью электронной таблицы построим модель работы прототипа коммутатора. Одна строка означает состояние модели по истечении очередной секунды. В столбцах «Порт 1» и «Порт 2» отображается объем данных в КБайт, полученных через этот порт к очередному моменту времени. В столбце «Выход» показаны в виде отрицательных чисел данные, удаленные из буфера к очередному моменту времени. В столбце «Буфер» считается объем данных, хранимых в буфере в указанный момент. Желтые клетки означают секунду, на которой заканчивается приём очередного пакета. Со следующей секунды возможна его передача. Зеленые клетки означают секунды, на которых передаётся в третью сеть пакет, размером, 40 КБайт, а голубые – размером 30 КБайт.

Секунда	Порт 1	Порт 2	Выход	Буфер
1	5			5
2	10			10
3	15			15
4	20			20
5	25	5		30
6	30	10		40
7	35	15		50
8	40	20		60
9	40	25		65
10	40	30		70
11	45	30		75
12	50	30		80
13	55	35	-40	50
14	60	40	-40	60
15	65	45	-40	70
16	70	50	-70	50
17	75	55	-70	60
18	80	60	-70	70
19	80	60	-70	70
20	80	60	-70	70
21	85	65	-70	80

22	90	70	-70	90
23	95	75	-110	60
24	100	80	-110	70
25	105	85	-110	80
26	110	90	-140	60
27	115	90	-140	65
28	120	90	-140	70
29	120	95	-140	75

Обратим внимание, что по истечении 22-ой секунды от начального момента времени в буфере занято 90 КБайт. При этом, уже в начале следующей секунды из буфера будет удалено 40 КБайт, а поступит 10, причем по условию удаление будет предшествовать заполнению. Легко заметить, что достижение занятости в буфере 90 КБайт происходит в худшем случае, когда одновременно завершен на 18-й секунде прием сразу двух пакетов. В этом случае будет происходить накопление данных в буфере во время передачи большего пакета при наличии в памяти полного пакета меньшего размера. В принципе, можно продолжить таблицу и убедиться, что вторая такая же ситуация с одновременным завершением приёма двух пакетов произойдет по истечению 56-й секунды и, следовательно, занятость буфера в 90 КБайт следующий раз будет достигнута на 60-й секунде. Следовательно, процесс изменения памяти буфера будет цикличен и значение 90 КБайт никогда не будет превышена. Значит это и есть максимальный объем буфера, при котором будет неограниченная во времени безаварийная работа.

## 7. Технологии обработки информации в электронных таблицах, технологии сортировки и фильтрации данных (1 балл)

[Если...]

Дан фрагмент электронной таблицы в режиме отображения формул:

	A	B	C	D	E
1		28	63	18	
2		=ЕСЛИ(ИЛИ(ОСТАТ(A2;B1)<>0;НЕ(И(ОСТАТ(A2;C1)=0;ОСТАТ(A2;D1)=0)));ОСТАТ(A2;A1)=0);1;0)			
3					
4					

Какое максимальное целое число можно поместить в ячейку A1 так, чтобы при любом целом положительном числе в ячейке A2, в ячейке B2 выводилось значение 1. В ответе укажите целое число.

**Ответ: 252**

**Решение:**

Обозначим утверждение, что число в ячейке A2 делится нацело на некоторое целое число X как  $DIV(X)$ .

Тогда можно утверждать, что 1 в ячейке B2 будет выведено тогда и только тогда, когда будет истинным выражение:

$\overline{DIV(28)} \vee \overline{DIV(63)} \wedge \overline{DIV(18)} \vee DIV(A1)$ , где A1 – значение в ячейке A1.

Преобразуем это выражение и получим:

$$\overline{DIV(28)} \vee \overline{DIV(63)} \vee \overline{DIV(18)} \vee DIV(A1)$$

Тогда, чтобы это выражение было истинно для любых значений в ячейке A2, утверждение  $DIV(A1)$  должно быть эквивалентно утверждению  $\overline{DIV(28)} \wedge \overline{DIV(63)} \wedge \overline{DIV(18)}$ . То есть A1 должно принимать такое значение, чтобы для тех чисел, которые одновременно делятся на 28, 63 и 18, они точно делились бы на A1. Найдем наименьшее общее кратное трех чисел:  $28=2*2*7$ ,  $63=3*3*7$ ,  $18=2*3*3$ . Получим  $2*2*3*3*7=252$ . Тогда, в качестве значения A1 нам подходит любой делитель числа 252, но в условии просят найти максимальное значение. Значит это само число 252.

## 8. Технологии программирования (2 балла)

[Рассадка в самолете]

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

Вам дана схема самолета, со следующими обозначениями:

Символ «#» обозначает занятое место

«.» - свободное место

«|» - проход

«\_» - перегородки между отсеками

Пример схемы самолета на шестнадцать мест (четыре свободных и двенадцать занятых) с одним проходом и двумя отсеками:

```
##|#.
.#|##
_ | _
.#|##
####.
```

Координатой кресла будем считать номер строки и номер столбца, в которых это кресло содержится на схеме. Расстоянием между двумя креслами определим по формуле:  $d=|x_1-x_0|+|y_1-y_0|+p+2w$ , где  $(x_0, y_0)$  — координаты первого кресла,  $(x_1, y_1)$  — координаты второго кресла,  $p$  — число проходов между  $y_0$ -м и  $y_1$ -м столбцом, а  $w$  — число перегородок между  $x_0$ -й и  $x_1$ -й строкой.

Например, на схеме выше расстояние между левым верхним свободным креслом (2-я строка, 1-й столбец) и правым нижним креслом (5-я строка, 5-й столбец) будет равно  $|5-2|+|5-1|+1+2\cdot 1=10$ .

Вам необходимо найти два свободных места с наибольшим расстоянием между ними и рассчитать это расстояние.

#### Формат входных данных

В первой строке входных данных задано число  $t$  — число схем самолетов ( $1 \leq t \leq 100$ ).

Затем следует  $t$  описаний схем самолетов. В первой строке описания схемы содержатся два целых числа  $n$  и  $m$  — число строк и столбцов ( $1 \leq n, m \leq 20$ ). В следующих  $n$  строках содержатся по  $m$  символов каждый из которых либо «#», либо «.», либо «|», либо «\_».

Гарантируется, что есть хотя бы два свободных места, и если хотя бы один символ в строке «\_», то все символы в строке задают перегородку либо проход. Также гарантируется, что если хотя бы один символ в столбце «|», то все символы в этом столбце задают проход.

#### Формат выходных данных

Для каждой схемы выведите расстояние между двумя искомыми местами в отдельной строке.

#### Пример

Стандартный ввод	Стандартный вывод
4	10
5 5	3
## #.	14
.# ##	5
_   _	
.# ##	
## #.	
1 5	
.	
5 5	
.# ##	
_   _	
## #.	
_   _	
## #.	
3 4	
_   _	
_   _	
. .	

#### Решение

Данная задача была представлена в двух вариантах. В первом нужно было найти два кресла с наибольшим расстоянием между ними, во втором с наименьшим.

Оба варианта решаются абсолютно аналогично. Более того, от участников не требовалось написать самое оптимальное решение. Ограничения позволяют перебрать два свободных кресла и линейным проходом найти расстояние между ними.

Например, для фиксированных свободных кресел  $(x_0, y_0)$  и  $(x_1, y_1)$ , можно было запустить два цикла:

- Первый от  $\min(x_0, x_1)$  до  $\max(x_0, x_1)$  внутри которого будем считать число встреченных символов «\_» в столбце  $y_0$ , например. Символов «|» в столбце  $y_0$  быть не может, нам это гарантировано условием.

- Второй от  $\min(y_0, y_1)$  до  $\max(y_0, y_1)$ , внутри которого будем считать число встреченных символов «|» в строке  $x_0$ , например. Символов «\_» в строке  $x_0$  быть не может, нам это также гарантировано условием.

Зная число перегородок и проходов, можно вычислить расстояние подставив значения в формулу из условия. Далее остается всего лишь найти минимум/максимум среди всех посчитанных расстояний.

С примерами авторских решений можно ознакомиться в папке solutions в архиве задачи.

## 9. Технологии программирования (2 балла)

### [Ближайшая интересная пара]

Имя входного файла	стандартный ввод
Имя выходного файла	стандартный вывод
Ограничение по времени	3 секунды
Ограничение по памяти	256 мегабайт

Назовём пару элементов массива  $a_i$  и  $a_j$  интересной, если  $i < j$ ,  $a_i = -a_j$  и  $a_i \leq 0$ .

Вам дано несколько массивов, необходимо в каждом из них найти интересную пару с минимальной абсолютной разностью между индексами.

#### Формат входных данных

В первой строке дано число  $t$  — число тестовых наборов ( $1 \leq t \leq 100$ ).

Затем следуют  $t$  тестовых наборов. Каждый тестовый набор задается двумя строками. В первой из них дано число  $n_i$  — количество элементов в  $i$ -м массиве ( $1 \leq n_i \leq 10^5$ ). Во второй строке заданы  $n_i$  чисел  $a_{i,j}$  — элементы  $i$ -го массива ( $-10^9 \leq a_{i,j} \leq 10^9$ ).

Гарантируется, что сумма  $n_i$  по всем тестовым наборам не превосходит  $5 \cdot 10^5$ .

#### Формат выходных данных

Для каждого тестового набора выведите одно число — минимальную разность индексов в интересных парах. Если интересных пар в массиве нет, выведите  $-1$ .

#### Пример

Стандартный ввод	Стандартный вывод
5	1
3	3
1 -1 1	1
4	-1
-1 2 -2 1	1
4	
1 -2 2 -1	
3	
1 2 3	
2	
0 0	

#### Решение

Данная задача была представлена в двух вариантах. В первом нужно было найти пары, где  $i < j$ ,  $a_i = -a_j$  и  $a_i \leq 0$ , а во втором  $i < j$ ,  $a_i = -a_j$  и  $a_i \geq 0$ . Оба варианта решаются абсолютно аналогично. Разберем первый вариант. Один из способов решения заключается в том, чтобы составить два массива  $b$  и  $c$ , таким образом, чтобы в  $b$  лежали пары  $(a_i, i)$  для  $a_i \leq 0$ , а в  $c$  пары  $(a_i, i)$  для  $a_i \geq 0$ . Затем можно отсортировать массив  $b$  по убыванию  $a_i$ , а  $c$  по возрастанию  $a_i$ . При этом порядок индексов для равных  $a_i$  должен быть возрастающим. Например, если  $a = [-1, 2, -2, 1, 0, 0, -2]$ , то  $b$  после сортировки будет  $[(0, 4), (0, 5), (-1, 0), (-2, 2), (-2, 6)]$ , а  $c$  —  $[(0, 4), (0, 5), (1, 3), (2, 1)]$ . Далее с помощью метода двух указателей можно пройти по этим двум массивам и найти нужную пару. Подробнее с примером такого прохода в авторском решении можно ознакомиться в папке solutions в архиве задачи.