1. На ускорителе для большого числа частиц производятся замеры скорости каждой из них. Скорость частицы — это целое число (положительное, отрицательное или 0). Частиц, скорость которых измерена, может быть очень много, но не может быть меньше трёх. Скорости всех частиц различны. При обработке результатов в каждой серии эксперимента отбирается основное множество скоростей. Это такое непустое множество скоростей частиц (в него могут войти как скорость одной частицы, так и скорости всех частиц серии), для которого произведение скоростей является максимальным среди всех возможных множеств. При нахождении произведения знак числа учитывается. Если есть несколько таких множеств, то основным считается то, которое содержит наибольшее количество элементов.

Вам предлагается написать программу (укажите используемую версию языка программирования, например, *Borland Pascal 7.0*), которая будет обрабатывать результаты эксперимента, находя основное множество. Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи.

На вход программе в первой строке подаётся количество частиц *N*. В каждой из последующих *N* строк записано одно целое число, по абсолютной величине не превышающее 10^9.

Вам предлагается два задания, связанных с этой задачей: задание А и задание Б. Вы можете решать оба задания или одно из них по своему выбору. Итоговая оценка выставляется как максимальная из оценок за задания А и Б. Если решение одного из заданий не представлено, то считается, что оценка за это задание — 0 баллов.

Задание Б является усложнённым вариантом задания А, оно содержит дополнительные требования к программе.

А. Напишите на любом языке программирования программу для решения поставленной задачи, в которой входные данные будут запоминаться в массиве. Перед программой укажите версию языка программирования.
Обязательно укажите, что программа является решением **задания А**. Максимальная оценка за выполнение задания А — 2 балла.
Б. Напишите программу для решения поставленной задачи, которая будет эффективна как по времени, так и по памяти (или хотя бы по одной из этих характеристик). Программа считается эффективной по времени, если время работы программы пропорционально количеству полученных показаний прибора N, т. е. при увеличении N в k раз время работы программы должно увеличиваться не более чем в k раз. Программа считается эффективной по памяти, если размер памяти, использованной в программе для хранения данных, не зависит от числа N и не превышает 1 килобайта.

Перед программой укажите версию языка программирования и кратко опишите использованный алгоритм.
Обязательно укажите, что программа является решением **задания Б**. Максимальная оценка за правильную программу, эффективную по времени и по памяти, — 4 балла.

Максимальная оценка за правильную программу, эффективную по времени, но неэффективную по памяти, — 3 балла.

Напоминаем! Не забудьте указать, к какому заданию относится каждая из представленных Вами программ.

Пример входных данных:

```
5
123
2
-1000
0
10
```

Программа должна вывести в порядке возрастания номера частиц, скорости которых принадлежат основному множеству данной серии. Нумерация частиц ведётся с единицы.
Пример выходных данных для приведённого выше примера входных данных:

1 2 5

2. На ускорителе для большого числа частиц производятся замеры скорости каждой из них. Скорость частицы — это целое число (положительное, отрицательное или 0). Частиц, скорость которых измерена, может быть очень много, но не может быть меньше трёх. Скорости всех частиц различны. В серии обязательно присутствует хотя бы одна частица с отрицательной скоростью.

При обработке результатов в каждой серии эксперимента отбирается основное множество скоростей. Это такое непустое множество скоростей частиц (в него могут войти как скорость одной частицы, так и скорости всех частиц серии), для которого произведение скоростей является минимальным среди всех возможных подмножеств. При нахождении произведения знак числа учитывается. Если есть несколько таких множеств, то берётся то, которое содержит наибольшее количество элементов.

Вам предлагается написать эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет обрабатывать результаты эксперимента, находя основное множество.

Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи.

На вход программе в первой строке подаётся количество частиц *N*. В каждой из последующих *N* строк записано одно целое число, по абсолютной величине не превышающее 10^9. Все N чисел различны.

Пример входных данных:

5
123
2
−1000
0
10

Программа должна вывести в порядке возрастания номера частиц, скорости которых принадлежат основному множеству данной серии. Нумерация частиц ведётся с единицы.
Пример выходных данных для приведённого выше примера входных данных:

1 2 3 5

3. На ускорителе для большого числа частиц производятся замеры скорости каждой из них. Скорость частицы — это целое число (положительное, отрицательное или 0). Частиц, скорость которых измерена, может быть очень много, но не может быть меньше трёх. Скорости всех частиц различны. При обработке результатов в каждой серии эксперимента отбирается основное множество скоростей. Это такое непустое множество скоростей частиц (в него могут войти как скорость одной частицы, так и скорости всех частиц серии), для которого произведение скоростей является максимальным среди всех возможных множеств. При нахождении произведения знак числа учитывается. Если есть несколько таких множеств, то основным считается то, которое содержит наибольшее количество элементов.

Вам предлагается написать эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, *Borland Pascal 7.0*), которая будет обрабатывать результаты эксперимента, находя основное множество. Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи.

На вход программе в первой строке подаётся количество частиц *N*. В каждой из последующих *N* строк записано одно целое число, по абсолютной величине не превышающее 10^9.

Пример входных данных:

```
5
   123
   2
-1000
   0
   10
```

Программа должна вывести в порядке возрастания номера частиц, скорости которых принадлежат основному множеству данной серии. Нумерация частиц ведётся с единицы.
Пример выходных данных для приведённого выше примера входных данных:

```
1 2 5
```

4. На ускорителе для большого числа частиц производятся замеры скорости каждой из них. Скорость частицы — это целое неотрицательное число. Частиц, скорость которых измерена, может быть очень много, но не может быть меньше трёх. Скорости всех частиц различны. Скорость, по крайней мере, одной частицы нечётна.

При обработке результатов в каждой серии эксперимента отбирается множество скоростей. Это непустое подмножество скоростей частиц (в него могу войти как скорость одной частицы, так и скорости всех частиц серии), такое, что сумма всех значений скоростей у него нечётна и максимальна среди всех возможных непустых подмножеств с нечётной суммой. Если таких подмножеств несколько, то из них выбирается то подмножество, которое содержит наименьшее количество элементов.

Вам предлагается написать программу (укажите используемую версию языка программирования, например, *Borland Pascal 7.0*), которая будет обрабатывать результаты эксперимента, находя основное множество. Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи.

На вход программе в первой строке подаётся количество частиц N. В каждой из последующих *N* строк записано одно целое неотрицательное число, не превышающее 10^9. Все *N* чисел различны. Хотя бы одно из чисел нечётно.

Вам предлагается два задания, связанных с этой задачей: задание А и задание Б. Вы можете решать оба задания или одно из них по своему выбору. Итоговая оценка выставляется как максимальн

ая из оценок за задания А и Б. Если решение одного из заданий не представлено, то считается, что оценка за это задание — 0 баллов.

Задание Б является усложнённым вариантом задания А, оно содержит дополнительные требования к программе.

А. Напишите на любом языке программирования программу для решения поставленной задачи, в которой входные данные будут запоминаться в массиве. Перед программой укажите версию языка программирования.
Обязательно укажите, что программа является решением **задания А**. Максимальная оценка за выполнение задания А — 2 балла.
Б. Напишите программу для решения поставленной задачи, которая будет эффективна как по времени, так и по памяти (или хотя бы по одной из этих характеристик). Программа считается эффективной по времени, если время работы программы пропорционально количеству полученных показаний прибора N, т. е. при увеличении N в k раз время работы программы должно увеличиваться не более чем в k раз. Программа считается эффективной по памяти, если размер памяти, использованной в программе для хранения данных, не зависит от числа N и не превышает 1 килобайта.

Перед программой укажите версию языка программирования и кратко опишите использованный алгоритм.
Обязательно укажите, что программа является решением **задания Б**. Максимальная оценка за правильную программу, эффективную по времени и по памяти, — 4 балла.

Максимальная оценка за правильную программу, эффективную по времени, но неэффективную по памяти, — 3 балла.

Напоминаем! Не забудьте указать, к какому заданию относится каждая из представленных Вами программ.

Пример входных данных:

```
3

123
0
2
```

Программа должна вывести в порядке возрастания номера частиц, скорости которых принадлежат основному множеству данной серии. Нумерация частиц ведётся с единицы.

Пример выходных данных для приведённого выше примера входных данных:

```
1 3
```

5. На ускорителе для большого числа частиц производятся замеры скорости каждой из них. Скорость частицы — это вещественное неотрицательное число, записанное с точностью до одного знака после десятичной точки. Частиц, скорость которых измерена, может быть очень много, но не может быть меньше трёх. Все значения скоростей не превосходят 100000. При обработке результатов в каждой серии эксперимента отбирается основное множество частиц. Это такое непустое подмножество частиц, для которого произведение скоростей является максимально возможным. Если таких подмножеств несколько, то из них выбирается множество, которое содержит наименьшее количество элементов. В основное множество могут войти, например, как все частицы, так и ровно одна частица. Если числовое множество содержит только одно число x, то произведением элементов этого множества является число x.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, *Borland Pascal 7.0*), которая будет обрабатывать результаты эксперимента, находя количество элементов в основном множестве и значение минимального элемента в этом множестве. Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи.

На вход программе в первой строке подаётся количество частиц *N*. В каждой из последующих *N* строк записано одно неотрицательное вещественное число с точностью до 1 знака после десятичной точки. Пример входных данных:

```
5

123.4
0.2
7.2
0.0
314 .1
```

Программа должна вывести сначала размер основного множества, а затем его минимальный элемент.

Пример выходных данных для приведённого выше примера входных данных:

```
3 7.2
```

6. На ускорителе для большого числа частиц производятся замеры скорости каждой из них. Скорость частицы — это целое число (положительное, отрицательное или 0). Частиц, скорость которых измерена, может быть очень много, но не может быть меньше трёх. Скорости всех частиц различны.

При обработке результатов в каждой серии эксперимента отбирается основное множество скоростей. Это такое непустое множество скоростей частиц (в него могут войти как скорость одной частицы, так и скорости всех частиц серии), для которого произведение скоростей является максимальным среди всех возможных множеств. При нахождении произведения знак числа учитывается. Если есть несколько таких множеств, то основным считается то, которое содержит наибольшее количество элементов.

Вам предлагается написать эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, *Borland Pascal 7.0*), которая будет обрабатывать результаты эксперимента, находя основное множество. Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи.

На вход программе в первой строке подаётся количество частиц *N*. В каждой из последующих *N* строк записано одно целое число, по абсолютной величине не превышающее 10^9.

Пример входных данных:

```
5
```

123
2

-1000

0

10

Программа должна вывести в порядке возрастания номера частиц, скорости которых принадлежат основному множеству данной серии. Нумерация частиц ведётся с единицы.
Пример выходных данных для приведённого выше примера входных данных:

1 2 5

7. На ускорителе для большого числа частиц производятся замеры скорости каждой из них. Скорость частицы — это целое неотрицательное число. Частиц, скорость которых измерена, может быть очень много, но не может быть меньше трёх. Скорости всех частиц различны.

При обработке результатов в каждой серии эксперимента отбирается основное множество скоростей. Это такое непустое множество скоростей частиц (в него могут войти как скорость одной частицы, так и скорости всех частиц серии), такое, что сумма значений скоростей у него чётна и максимальна среди всех возможных непустых подмножеств с чётной суммой. Если есть несколько таких множеств, то основным считается то, которое содержит наибольшее количество элементов.

Вам предлагается написать эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, *Borland Pascal 7.0*), которая будет обрабатывать результаты эксперимента, находя основное множество. Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи.

На вход программе в первой строке подаётся количество частиц N. В каждой из последующих *N* строк записано одно целое неотрицательное число, не превышающее 10^9. Все *N* чисел различны. Хотя бы одно из чисел нечётно.

Пример входных данных:

5
123
2
1000
0
10

Программа должна вывести в порядке возрастания номера частиц, скорости которых принадлежат основному множеству данной серии. Нумерация частиц ведётся с единицы.

Пример выходных данных для приведённого выше примера входных данных: 2 3 5.

Примечание. Заметим, что прибавление нуля не изменяет чётность суммы, поэтому ноль должен бы входить в искомое множество. Но авторы задания в примере входных и выходных данных, видимо, подразумевают, что частица с нулевой скоростью не должна входить в искомое множество.

8. Радиотелескоп пытается получать и анализировать сигналы, поступающие из различных участков космоса, при этом различные шумы переводятся в последовательность вещественных неотрицательных чисел, заданных с точностью до одного знака после десятичной точки. Чисел может быть очень много, но не может быть меньше трёх. Все числа не превосходят 1000000.

В последовательности чисел, полученных из одного участка, выделяется основное подмножество элементов. Это такое непустое подмножество элементов, для которого произведение соответствующих чисел является максимально возможным. Если таких подмножеств несколько, то из них выбирается подмножество, которое содержит наименьшее количество элементов. Основное подмножество может содержать, например, как все элементы последовательности чисел, так и ровно один элемент. Если множество чисел содержит только одно число x, то произведением элементов этого множества считается число *x*.

Напишите программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет обрабатывать результаты, приходящие из одного участка, находя количество элементов в основном множестве и значение минимального элемента в этом множестве. Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи. На вход программе в первой строке подаётся количество сигналов N. В каждой из последующих N строк записано одно неотрицательное вещественное число с точностью до одного знака после десятичной точки.

Вам предлагается два задания, связанных с этой задачей: задание А и задание Б. Вы можете решать оба задания или одно из них по своему выбору. Итоговая оценка выставляется как максимальная из оценок за задания А и Б. Если решение одного из заданий не представлено, то считается, что оценка за это задание — 0 баллов.

Задание Б является усложнённым вариантом задания А, оно содержит дополнительные требования к программе.

А. Напишите на любом языке программирования программу для решения поставленной задачи, в которой входные данные будут запоминаться в массиве. Перед программой укажите версию языка программирования.
Обязательно укажите, что программа является решением **задания А**. Максимальная оценка за выполнение задания А — 2 балла.
Б. Напишите программу для решения поставленной задачи, которая будет эффективна как по времени, так и по памяти (или хотя бы по одной из этих характеристик). Программа считается эффективной по времени, если время работы программы пропорционально количеству полученных показаний прибора N, т. е. при увеличении N в k раз время работы программы должно увеличиваться не более чем в k раз. Программа считается эффективной по памяти, если размер памяти, использованной в программе для хранения данных, не зависит от числа N и не превышает 1 килобайта.

Перед программой укажите версию языка программирования и кратко опишите использованный алгоритм.
Обязательно укажите, что программа является решением **задания Б**. Максимальная оценка за правильную программу, эффективную по времени и по памяти, — 4 балла.

Максимальная оценка за правильную программу, эффективную по времени, но неэффективную по памяти, — 3 балла.

Напоминаем! Не забудьте указать, к какому заданию относится каждая из представленных Вами программ.

Пример входных данных:

```
6
123.4
0.2
200.2
0.0
6.7
218.0
```

Программа должна вывести в одной строке сначала количество элементов в основном множестве, а затем — его минимальный элемент.

Пример выходных данных для приведённого выше примера входных данных: 4 6.7.

9. На ускорителе для большого числа частиц производятся замеры скорости каждой из них. Скорость частицы — это неотрицательное целое число. Частиц, скорость которых измерена, может быть очень много, но не может быть меньше трёх. Скорости всех частиц различны. При обработке результатов в каждой серии эксперимента отбирается основное множество скоростей. Это такое непустое множество скоростей частиц (в него могут войти как скорость одной частицы, так и скорости всех частиц серии), такое, что сумма значений скоростей у него чётна и максимальна среди всех возможных непустых подмножеств с чётной суммой. Если есть несколько таких множеств, то основным считается то, которое содержит наименьшее количество элементов.

Вам предлагается написать эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, *Borland Pascal 7.0*), которая будет обрабатывать результаты эксперимента, находя основное множество. Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи.

На вход программе в первой строке подаётся количество частиц N. В каждой из последующих N строк записано одно целое неотрицательное число, не превышающее 10^9. Все N чисел различны. Хотя бы одно из чисел нечётно.

Пример входных данных:

```
5
123
2
1000
0
10
```

Программа должна вывести в порядке возрастания номера частиц, скорости которых принадлежат основному множеству данной серии. Нумерация частиц ведётся с единицы.

Пример выходных данных для приведённого выше примера входных данных: 2 3 5.

10. Радиотелескоп пытается получать и анализировать сигналы, поступающие из различных участков космоса, при этом различные шумы переводятся в последовательность целых неотрицательных чисел. Чисел может быть очень много, но не может быть меньше трёх. Все числа различны. Хотя бы одно из чисел нечётно.

В данных, полученных из одного участка, выделяется основное подмножество чисел. Это непустое подмножество чисел (в него могут войти как одно число, так и все поступившие числа), такое, что их сумма нечётна и максимальна среди всех возможных непустых подмножеств с нечётной суммой. Если таких подмножеств несколько, то из них выбирается то подмножество, которое содержит наименьшее количество элементов.

Вам предлагается написать программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет обрабатывать результаты, приходящие из одного участка, находя основное подмножество. Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи. На вход программе в первой строке подаётся количество сигналов *N*. В каждой из последующих *N* строк записано одно целое неотрицательное число, не превышающее 10^9.

Вам предлагается два задания, связанных с этой задачей: задание А и задание Б. Вы можете решать оба задания или одно из них по своему выбору. Итоговая оценка выставляется как максимальная из оценок за задания А и Б. Если решение одного из заданий не представлено, то считается, что оценка за это задание — 0 баллов.

Задание Б является усложнённым вариантом задания А, оно содержит дополнительные требования к программе.

А. Напишите на любом языке программирования программу для решения поставленной задачи, в которой входные данные будут запоминаться в массиве, после чего будут проверены все возможные пары элементов. Перед программой укажите версию языка программирования.
Обязательно укажите, что программа является решением **задания А**. Максимальная оценка за выполнение задания А — 2 балла.
Б. Напишите программу для решения поставленной задачи, которая будет эффективна как по времени, так и по памяти (или хотя бы по одной из этих характеристик). Программа считается эффективной по времени, если время работы программы пропорционально количеству полученных показаний прибора N, т. е. при увеличении N в k раз время работы программы должно увеличиваться не более чем в k раз. Программа считается эффективной по памяти, если размер памяти, использованной в программе для хранения данных, не зависит от числа N и не превышает 1 килобайта.

Перед программой укажите версию языка программирования и кратко опишите использованный алгоритм.
Обязательно укажите, что программа является решением **задания Б**. Максимальная оценка за правильную программу, эффективную по времени и по памяти, — 4 балла.

Максимальная оценка за правильную программу, эффективную по времени, но неэффективную по памяти, — 3 балла.

Напоминаем! Не забудьте указать, к какому заданию относится каждая из представленных Вами программ.

Пример входных данных:

```
3
123
0
2
```

Программа должна вывести в порядке возрастания номера сигналов, которые принадлежат основному подмножеству данного участка. Нумерация элементов последовательности ведётся с единицы. *Пример выходных данных для приведённого выше примера входных данных:* 1 3.

11. Радиотелескоп пытается получать и анализировать сигналы, поступающие из различных участков космоса, при этом различные шумы переводятся в последовательность целых неотрицательных чисел. Чисел может быть очень много, но не может быть меньше трёх. Все числа различны. Хотя бы одно из чисел нечётно.

В данных, полученных из одного участка, выделяется основное подмножество чисел. Это непустое подмножество чисел (в него могут войти как одно число, так и все поступившие числа), такое, что их сумма чётна и максимальна среди всех возможных непустых подмножеств с чётной суммой. Если таких подмножеств несколько, то из них выбирается то подмножество, которое содержит наименьшее количество элементов.

Вам предлагается написать эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет обрабатывать результаты, приходящие из одного участка, находя основное подмножество. Перед текстом программы кратко опишите используемый Вами алгоритм решения задачи. На вход программе в первой строке подаётся количество сигналов *N*. В каждой из последующих *N* строк записано одно целое неотрицательное число, не превышающее 10^9.

Пример входных данных:

```
5
123
2
1000
0
10
```

Программа должна вывести в порядке возрастания номера сигналов, которые принадлежат основному подмножеству данного участка. Нумерация элементов последовательности ведётся с единицы. *Пример выходных данных для приведённого выше примера входных данных:* 2 3 5.

12. На плоскости задано множество точек с целочисленными координатами, никакие две из которых не совпадают и никакие три не лежат на одной прямой. Необходимо найти количество треугольников, обладающих следующими свойствами:

1) все вершины треугольника принадлежат заданному множеству;
2) ни одна вершина не лежит на осях координат;
3) треугольник не пересекается с осью Ox, но пересекается с осью Oy.

Напишите эффективную по времени и по используемой памяти программу для решения этой задачи.

Программа считается эффективной по времени, если при увеличении количества точек в k раз время работы возрастает не более чем в k раз. Программа считается эффективной по памяти, если размер памяти для хранения всех необходимых данных не зависит от количества точек и не превышает 1 килобайта.

Перед текстом программы кратко опишите алгоритм решения и укажите язык программирования и его версию.

Входные данные

В первой строке задаётся N – количество точек в заданном множестве. Каждая из следующих строк содержит два целых числа x и y – координаты очередной точки. Гарантируется, что $1 \le N \le 10000$, $-1000 \le x, y \le 1000$, никакие две точки не совпадают, никакие три не лежат на одной прямой.

Пример входных данных:

```
4
6 6
−8 8
−9 −9
7 5
```

Выходные данные

Необходимо вывести единственное число: количество удовлетворяющих требованиям треугольников.

Пример выходных данных для приведённого выше примера входных данных:

1

13. На плоскости задано множество точек с целочисленными координатами, никакие две из которых не совпадают и никакие три не лежат на одной прямой. Необходимо найти количество треугольников, обладающих следующими свойствами:

1) все вершины треугольника принадлежат заданному множеству;
2) ни одна вершина не лежит на осях координат;
3) треугольник не пересекается с осью Oy, но пересекается с осью Ox.

Напишите эффективную по времени и по используемой памяти программу для решения этой задачи.

Программа считается эффективной по времени, если при увеличении количества точек в k раз время работы возрастает не более чем в k раз. Программа считается эффективной по памяти, если размер памяти для хранения всех необходимых данных не зависит от количества точек и не превышает 1 килобайта.

Перед текстом программы кратко опишите алгоритм решения и укажите язык программирования и его версию.

Входные данные

В первой строке задаётся N – количество точек в заданном множестве. Каждая из следующих строк содержит два целых числа x и y – координаты очередной точки. Гарантируется, что $1 \le N \le 10000$, $-1000 \le x, y \le 1000$, никакие две точки не совпадают, никакие три не лежат на одной прямой.

Пример входных данных:

```
4
6 6
−8 8
−9 −9
−7 5
```

Выходные данные

Необходимо вывести единственное число: количество удовлетворяющих требованиям треугольников.

Пример выходных данных для приведённого выше примера входных данных:

1

14. На вход даны пары чисел. Нужно выбрать из каждой пары по одному числу так, чтобы сумма всех выбранных чисел не была кратна 6 и при этом была минимально возможной. Напишите программу, выводящую такую сумму на экран. Если же ее невозможно получить, выведите 0.

Вам предлагается два задания, связанных с этой задачей: задание А и задание Б. Вы можете решать оба задания или одно из них по своему выбору. Итоговая оценка выставляется как максимальная из оценок за задания А и Б. Если решение одного из заданий не представлено, то считается, что оценка за это задание — 0 баллов.

Задание Б является усложнённым вариантом задания А, оно содержит дополнительные требования к программе.

А. Напишите на любом языке программирования программу для решения поставленной задачи, в которой входные данные будут запоминаться в массиве. Перед программой укажите версию языка программирования.
Обязательно укажите, что программа является решением **задания А**. Максимальная оценка за выполнение задания А — 2 балла.
Б. Напишите программу для решения поставленной задачи, которая будет эффективна как по времени, так и по памяти (или хотя бы по одной из этих характеристик). Программа считается эффективной по времени, если время работы программы пропорционально количеству полученных показаний прибора N, т. е. при увеличении N в k раз время работы программы должно увеличиваться не более чем в k раз. Программа считается эффективной по памяти, если размер памяти, использованной в программе для хранения данных, не зависит от числа N и не превышает 1 килобайта.

Перед программой укажите версию языка программирования и кратко опишите использованный алгоритм.
Обязательно укажите, что программа является решением **задания Б**. Максимальная оценка за правильную программу, эффективную по времени и по памяти, — 4 балла.

Максимальная оценка за правильную программу, эффективную по времени, но неэффективную по памяти, — 3 балла.

Напоминаем! Не забудьте указать, к какому заданию относится каждая из представленных Вами программ.

Задача А. Количество пар известно заранее и равно 6. Числа не превышают 30 000.
Пример входных данных:
```
5 4
3 2
1 1
18 3
11 12
2 5
```
Пример выходных данных:
```
23
```

Задача Б. Количество пар N не известно заранее и может принимать значения 2 <= N <= 200 000. На вход подается сначала количество пар, затем сами пары. Числа по модулю не превышают 30 000.
Пример входных данных:
```
6
5 4
3 2
1 1
18 3
11 12
2 5
```
Пример выходных данных:
```
23
```

15. На вход даны пары чисел. Нужно выбрать из каждой пары по одному числу так, чтобы сумма всех выбранных чисел не была кратна 4 и при этом была максимально возможной. Напишите программу, выводящую такую сумму на экран. Если же ее невозможно получить, выведите 0. Баллы начисляются за ту из подзадач, что решена на большее количество баллов. Задача А дает 2 балла, задача Б - 4 балла. В задаче А приведите неэффективный алгоритм. При решении указывайте, какую подзадачу делаете. За алгоритм, неэффективный по времени ИЛИ памяти, дается 3 балла, по времени И памяти - 2 балла.

Задача А. Количество пар известно заранее и равно 6. Числа не превышают 30 000.

Задача Б. Количество пар N не известно заранее и может принимать значения 2 <= N <= 200 000. На вход подается сначала количество пар, затем сами пары. Числа по модулю не превышают 30 000.

16. Дан набор из N целых положительных чисел. Необходимо выбрать из набора произвольное количество чисел так, чтобы их сумма была как можно больше и при этом не делилась на 6. В ответе нужно указать количество выбранных чисел и их сумму, сами числа выводить не надо. Если получить нужную сумму невозможно, считается, что выбрано 0 чисел и их сумма равна 0.

Напишите эффективную по времени и по памяти программу для решения этой задачи.

Программа считается эффективной по времени, если при увеличении количества исходных чисел N в k раз время работы программы увеличивается не более чем в k раз.

Программа считается эффективной по памяти, если память, необходимая для хранения всех переменных программы, не превышает 1 килобайт и не увеличивается с ростом N.

Максимальная оценка за правильную (не содержащую синтаксических ошибок и дающую правильный ответ при любых допустимых входных данных) программу, эффективную по времени и по памяти, — 4 балла.

Максимальная оценка за правильную программу, эффективную только по времени или только по памяти, — 3 балла.

Максимальная оценка за правильную программу, не удовлетворяющую требованиям эффективности, — 2 балла.

Вы можете сдать одну или две программы решения задачи. Если Вы сдадите две программы, каждая из них будет оцениваться независимо от другой, итоговой станет бóльшая из двух оценок.

Перед текстом программы кратко опишите алгоритм решения. Укажите использованный язык программирования и его версию.

Описание входных и выходных данных

В первой строке входных данных задаётся количество чисел N ($1 \le N \le 1000$).

В каждой из последующих N строк записано одно натуральное число, не превышающее 10 000.

Пример входных данных:

```
3
1
2
3
```

В результате работы программа должна вывести два числа: сначала количество выбранных чисел, затем их сумму.

Пример выходных данных для приведённого выше примера входных данных:

```
2 5
```

В данном случае из предложенного набора нужно выбрать два числа (2 и 3), их сумма равна 5.

17. Дан набор из N целых положительных чисел. Необходимо выбрать из набора произвольное количество чисел так, чтобы их сумма была как можно больше и при этом не делилась на 8. В ответе нужно указать количество выбранных чисел и их сумму, сами числа выводить не надо. Если получить нужную сумму невозможно, считается, что выбрано 0 чисел и их сумма равна 0.

Напишите эффективную по времени и по памяти программу для решения этой задачи.

Программа считается эффективной по времени, если при увеличении количества исходных чисел N в k раз время работы программы увеличивается не более чем в k раз.

Программа считается эффективной по памяти, если память, необходимая для хранения всех переменных программы, не превышает 1 килобайт и не увеличивается с ростом N.

Максимальная оценка за правильную (не содержащую синтаксических ошибок и дающую правильный ответ при любых допустимых входных данных) программу, эффективную по времени и по памяти, — 4 балла.

Максимальная оценка за правильную программу, эффективную только по времени или только по памяти, — 3 балла.

Максимальная оценка за правильную программу, не удовлетворяющую требованиям эффективности, — 2 балла.

Вы можете сдать одну или две программы решения задачи. Если Вы сдадите две программы, каждая из них будет оцениваться независимо от другой, итоговой станет бóльшая из двух оценок.

Перед текстом программы кратко опишите алгоритм решения. Укажите использованный язык программирования и его версию.

Описание входных и выходных данных

В первой строке входных данных задаётся количество чисел N ($1 \le N \le 1000$).

В каждой из последующих N строк записано одно натуральное число, не превышающее 10 000.

Пример входных данных:

```
3
1
2
5
```

В результате работы программа должна вывести два числа: сначала количество выбранных чисел, затем их сумму.

Пример выходных данных для приведённого выше примера входных данных:

```
2 7
```

В данном случае из предложенного набора нужно выбрать два числа (2 и 5), их сумма равна 7.