**1.** По каналу связи передаются положительные целые числа, не превышающие 1000 – результаты измерений, полученных в ходе эксперимента (количество измерений $N$ известно заранее, гарантируется, что $N > 2$). После окончания эксперимента передаётся контрольное значение – наибольшее число $R$, удовлетворяющее следующим условиям.

1. $R$ – сумма двух различных переданных элементов последовательности («различные» означает, что нельзя просто удваивать переданные числа, суммы различных, но равных по величине элементов допускаются).
2. $R$ кратно 3.
3. Если в последовательности нет двух чисел, сумма которых кратна 3, контрольное значение считается равным 1.

В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Free Pascal 2.6.4), которая будет проверять правильность контрольного значения.

Программа должна напечатать отчёт по следующей форме:

```
Вычисленное контрольное значение: …
Контроль пройден (или Контроль не пройден)
```

Перед текстом программы кратко опишите используемый вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел $N$ ($N > 2$). В каждой из последующих N строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

*Пример входных данных*:

```
6
100
8
33
145
19
84
153
```

*Пример выходных данных для приведённого выше примера входных данных*:

```
Вычисленное контрольное значение: 153
Контроль пройден
```

**2.** По каналу связи передаются положительные целые числа, не превышающие 1000 – результаты измерений, полученных в ходе эксперимента (количество измерений $N$ известно заранее, гарантируется, что $N > 2$). После окончания эксперимента передаётся контрольное значение – наименьшее число $R$, удовлетворяющее следующим условиям.

1. $R$ – сумма двух различных переданных элементов последовательности («различные» означает, что нельзя просто удваивать переданные числа, суммы различных, но равных по величине элементов допускаются).
2. $R$ кратно 3.
3. Если в последовательности нет двух чисел, сумма которых кратна 3, контрольное значение считается равным 1.

В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Free Pascal 2.6.4), которая будет проверять правильность контрольного значения.

Программа должна напечатать отчёт по следующей форме:

```
Вычисленное контрольное значение: …
Контроль пройден (или Контроль не пройден)
```

Перед текстом программы кратко опишите используемый вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел $N$ ($N > 2$). В каждой из последующих N строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

*Пример входных данных*:

```
6
100
8
33
145
19
84
27
```

*Пример выходных данных для приведённого выше примера входных данных*:

```
Вычисленное контрольное значение: 27
Контроль пройден
```

**3.** Последовательность натуральных чисел характеризуется числом Y – наибольшим числом, кратным 26 и являющимся произведением двух элементов последовательности с различными номерами.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), находящую число Y для последовательности натуральных чисел, значение каждого элемента которой не превосходит 1000. Программа должна напечатать найденное число, если оно существует для заданной последовательности, или ноль в противном случае.

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел N. В каждой из последующих N строк записано одно натуральное число, не превышающее 1000.

Пример входных данных:

```
5
40
100
130
28
51
```

Пример выходных данных для приведённого выше примера входных данных: 13000

**4.** Последовательность натуральных чисел характеризуется числом X — наибольшим числом, кратным 14 и являющимся произведением двух элементов последовательности с различными номерами. Гарантируется, что хотя бы одно такое произведение в последовательности есть.

*Вам предлагаются два задания, связанные с этой задачей: задание А и задание Б. Вы можете решать оба задания А и Б 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 составляет 0 баллов.*

*Задание Б является усложненным 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 элементов последовательности N, т. е. при увеличении N в k раз время работы программы должно увеличиваться не более чем в k раз.

**Обязательно** укажите, что программа является решением **задания Б**.

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел N. В каждой из последующих N строк записано одно натуральное число, не превышающее 1000.

Пример входных данных:

5
40
1000
7
28
55

Пример выходных данных для приведённого выше примера входных данных:

28000

**5.** По каналу связи передаётся последовательность положительных целых чисел, все числа не превышают 1000. Количество чисел известно, но может быть очень велико. Затем передаётся контрольное значение последовательности — наибольшее число *R*, удовлетворяющее следующим условиям:

1) *R* — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных элементов последовательности, равных по величине, допускаются);

2) *R* делится на 14.

Если такого числа *R* нет, то контрольное значение полагается равным 0. В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

*Вычисленное контрольное значение: ...*

*Контроль пройден (*или *— Контроль не пройден)*

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел N. В каждой из последующих N строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

*Вам предлагаются два задания, связанные с этой задачей: задание А и задание Б. Вы можете решать оба задания А и Б 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 составляет 0 баллов.*

*Задание Б является усложненным вариантом задания А, оно содержит дополнительные требования к программе*. Перед программой укажите версию языка программирования.

**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.
**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 элементов последовательности N, т. е. при увеличении N в k раз время работы программы должно увеличиваться не более чем в k раз. **Обязательно** укажите, что программа является решением **задания Б**.

*Пример входных данных:*

```
6
77
14
7
9
499
100
7700
```

*Пример выходных данных для приведённого выше примера входных данных:*

```
Вычисленное контрольное значение: 7700
Контроль пройден
```

**6.** По каналу связи передаётся последовательность положительных целых чисел, все числа не превышают 1000. Количество чисел известно, но может быть очень велико. Затем передаётся контрольное значение последовательности — наибольшее число *R*, удовлетворяющее следующим условиям:

1) *R* — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных элементов последовательности, равных по величине, допускаются);

2) *R* делится на 10.

Если такого числа *R* нет, то контрольное значение полагается равным 0. В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

*Вычисленное контрольное значение: ...*

*Контроль пройден (*или *— Контроль не пройден)*

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел N. В каждой из последующих N строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

*Пример входных данных:*

*6*
*95*
*17*
*10*
*102*
*957*
*95*
*9690*

*Пример выходных данных для приведённого выше примера входных данных:*

*Вычисленное контрольное значение: 9690*
*Контроль пройден*

**7.** По каналу связи передаётся последовательность положительных целых чисел, все числа не превышают 1000. Количество чисел известно, но может быть очень велико. Затем передаётся контрольное значение последовательности — наибольшее число *R*, удовлетворяющее следующим условиям:

1) *R* — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных элементов последовательности, равных по величине, допускаются);

2) *R* делится на 6.

Если такого числа *R* нет, то контрольное значение полагается равным 0. В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

*Вычисленное контрольное значение: ...*

*Контроль пройден (*или *— Контроль не пройден)*

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел N. В каждой из последующих N строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

*Пример входных данных:*

*6*
*70*
*17*
*6*
*99*
*997*
*70*
*6930*

*Пример выходных данных для приведённого выше примера входных данных:*

*Вычисленное контрольное значение: 6930*
*Контроль пройден*

**8.** По каналу связи передаётся последовательность положительных целых чисел, все числа не превышают 1000. Количество чисел известно, но может быть очень велико. Затем передаётся контрольное значение последовательности — наибольшее число *R*, удовлетворяющее следующим условиям:

1) *R* — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных элементов последовательности, равных по величине, допускаются);

2) *R* делится на 33.

Если такого числа *R* нет, то контрольное значение полагается равным 0. В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

*Вычисленное контрольное значение: ...*

*Контроль пройден (*или *— Контроль не пройден)*

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел N. В каждой из последующих N строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

*Пример входных данных:*

*6*
*550*
*33*
*7*
*997*
*9*
*60*
*33000*

*Пример выходных данных для приведённого выше примера входных данных:*

*Вычисленное контрольное значение: 33000*
*Контроль пройден*

**9.** По каналу связи передаётся последовательность положительных целых чисел, все числа не превышают 1000. Количество чисел известно, но может быть очень велико. Затем передаётся контрольное значение последовательности — наибольшее число *R*, удовлетворяющее следующим условиям:

1) *R* — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных элементов последовательности, равных по величине, допускаются);

2) *R* делится на 35.

Если такого числа *R* нет, то контрольное значение полагается равным 0. В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

*Вычисленное контрольное значение: ...*

*Контроль пройден (*или *— Контроль не пройден)*

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел N. В каждой из последующих N строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

*Пример входных данных:*

*6*
*77*
*35*
*7*
*9*
*997*
*500*
*38500*

*Пример выходных данных для приведённого выше примера входных данных:*

*Вычисленное контрольное значение: 38500*
*Контроль пройден*

**10.** По каналу связи передаётся последовательность положительных целых чисел, все числа не превышают 1000. Количество чисел известно, но может быть очень велико. Затем передаётся контрольное значение последовательности — наибольшее число *R*, удовлетворяющее следующим условиям:

1) *R* — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных элементов последовательности, равных по величине, допускаются);

2) *R* делится на 21.

Если такого числа *R* нет, то контрольное значение полагается равным 0. В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

*Вычисленное контрольное значение: ...*

*Контроль пройден (*или *— Контроль не пройден)*

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел *N*. В каждой из последующих *N* строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

*Пример входных данных:*

*6*
*70*
*21*
*997*
*7*
*9*
*300*
*21000*

*Пример выходных данных для приведённого выше примера входных данных:*

*Вычисленное контрольное значение: 21000*
*Контроль пройден*

**11.** По каналу связи передаётся последовательность положительных целых чисел, все числа не превышают 1000. Количество чисел известно, но может быть очень велико. Затем передаётся контрольное значение последовательности — наибольшее число *R*, удовлетворяющее следующим условиям:

1) *R* — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных элементов последовательности, равных по величине, допускаются);

2) *R* делится на 22.

Если такого числа *R* нет, то контрольное значение полагается равным 0. В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

*Вычисленное контрольное значение: ...*

*Контроль пройден (*или *— Контроль не пройден)*

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел *N*. В каждой из последующих *N* строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

*Пример входных данных:*

*6*
*55*
*997*
*22*
*7*
*9*
*400*
*22000*

*Пример выходных данных для приведённого выше примера входных данных:*

*Вычисленное контрольное значение: 22000*
*Контроль пройден*

**12.** По каналу связи передаётся последовательность положительных целых чисел, все числа не превышают 1000. Количество чисел известно, но может быть очень велико. Затем передаётся контрольное значение последовательности — наибольшее число *R*, удовлетворяющее следующим условиям:

1) *R* — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел; допускаются произведения различных элементов последовательности, равных по величине);

2) *R* делится на 21.

Если такого числа *R* нет, то контрольное значение полагается равным 0.

В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

*Вычисленное контрольное значение: ...*

*Контроль пройден* (**или** — *Контроль не пройден*)

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел N. В каждой из последующих N строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

*Пример входных данных:*

```
6
70
21
997
7
9
300
21000
```

*Пример выходных данных для приведённого выше примера входных данных:*

```
Вычисленное контрольное значение: 21000
Контроль пройден
```

**13.** По каналу связи передаются данные в виде последовательности положительных целых чисел. Количество чисел заранее неизвестно, но не менее двух, признаком конца данных считается число 0. После данных передаётся контрольное значение. Оно равно такому максимально возможному произведению двух чисел из переданного набора, которое делится на 7, но не делится на 49. Если такое произведение получить нельзя, контрольное значение считается равным 1.

Напишите эффективную, в том числе по памяти, программу, которая будет моделировать процесс приёма данных. Программа должна ввести все числа и контрольное значение и напечатать краткий отчёт, включающий количество принятых чисел, принятое контрольное значение, вычисленное контрольное значение и вывод о совпадении значений. Перед текстом программы кратко опишите алгоритм решения задачи и укажите используемый язык программирования и его версию.

*Вам предлагаются два задания, связанные с этой задачей: задание А и задание Б. Вы можете решать оба задания А и Б 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 составляет 0 баллов.*

*Задание Б является усложненным вариантом задания А, оно содержит дополнительные требования к программе*. Перед программой укажите версию языка программирования.

**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.

**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 элементов последовательности N, т. е. при увеличении N в k раз время работы программы должно увеличиваться не более чем в k раз. **Обязательно** укажите, что программа является решением **задания Б**.

**Описание входных данных.**

В каждой строке исходных данных содержится одно целое число. Сначала идут строки с основными данными — положительными числами, затем число 0 (признак окончания данных), в последней строке — контрольное значение.

Программа должна вывести отчёт по форме, приведённой ниже в примере.

Пример входных данных:

```
6
7
8
9
0
64
```

Пример выходных данных для приведённого выше примера входных данных:

```
Введено чисел: 4
Контрольное значение: 64
Вычисленное значение: 63
Значения не совпали
```

**14.** По каналу связи передаётся последовательность положительных целых чисел. Все числа не превышают 1000, их количество известно, но может быть очень велико. Затем передаётся контрольное значение — наибольшее число R, удовлетворяющее следующим условиям:

1) R — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных, но равных по величине элементов допускаются);

2) R не делится на 10.

В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите программу которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

Получено чисел: …
Принятое контрольное значение: …
Вычисленное контрольное значение: …
Контроль пройден (или Контроль не пройден)

Если удовлетворяющее условию контрольное значение определить невозможно, вычисленное контрольное значение не выводится, но выводится фраза «Контроль не пройден».

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

*Вам предлагаются два задания, связанные с этой задачей: задание А и задание Б. Вы можете решать оба задания А и Б 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 составляет 0 баллов.*

*Задание Б является усложненным вариантом задания А, оно содержит дополнительные требования к программе*. Перед программой укажите версию языка программирования.

**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.

**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 элементов последовательности N, т. е. при увеличении N в k раз время работы программы должно увеличиваться не более чем в k раз. **Обязательно** укажите, что программа является решением **задания Б**.

**Входные данные**

В первой строке указывается количество чисел N. В каждой из последующих
N строк записано одно натуральное число, не превышающее 1000.
В последней строке записано контрольное значение.

*Пример входных данных:*

```
5
60
7
8
15
20
105
```

**Выходные данные**

Программа должна напечатать отчёт по образцу, приведённому в условии.

*Пример выходных данных для приведённого выше примера входных данных:*

Получено чисел: 5
Принятое контрольное значение: 105
Вычисленное контрольное значение: 105
Контроль пройден

**15.** По каналу связи передаётся последовательность положительных целых чисел. Все числа не превышают 1000, их количество известно, но может быть очень велико. Затем передаётся контрольное значение — наибольшее число R, удовлетворяющее следующим условиям:

1) R — произведение двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных, но равных по величине элементов допускаются);

2) R не делится на 15.

Гарантируется, что такое число R в последовательности будет найдено.

В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу которая будет проверять правильность контрольного значения. Программа должна напечатать отчёт по следующей форме:

Получено чисел: …
Принятое контрольное значение: …
Вычисленное контрольное значение: …
Контроль пройден (или Контроль не пройден)

Если удовлетворяющее условию контрольное значение определить невозможно, вычисленное контрольное значение не выводится, но выводится фраза «Контроль не пройден».

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

**Входные данные**

В первой строке указывается количество чисел N. В каждой из последующих N строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение.

*Пример входных данных:*

5
60
100
8
9
90
800

**Выходные данные**

Программа должна напечатать отчёт по образцу, приведённому в условии.

*Пример выходных данных для приведённого выше примера входных данных:*

Получено чисел: 5
Принятое контрольное значение: 800
Вычисленное контрольное значение: 800
Контроль пройден

**16.** По каналу связи передаётся последовательность положительных целых чисел, все числа не превышают 1000. Количество чисел известно, но может быть очень велико. Затем передаётся контрольное значение последовательности — наименьшее число R, удовлетворяющее следующим условиям:

1) R является произведением двух различных переданных элементов последовательности («различные» означает, что не рассматриваются квадраты переданных чисел, произведения различных элементов последовательности, равных по величине, допускаются);

2) R кратно 6.

Если такого числа R нет, то контрольное значение полагается равным 0. В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную, в том числе по используемой памяти, программу (укажите используемую версию языка программирования, например, Borland Pascal 7.0), которая будет проверять правильность контрольного значения.

Программа должна напечатать отчёт по следующей форме:

Вычисленное контрольное значение: …
Контроль пройден (или — Контроль не пройден)

Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел N;
в программе можно считать, что $2 \le N \le 10\,000$. В каждой из последующих N строк записано одно натуральное число, не превышающее 1000. В последней строке записано контрольное значение — натуральное число, не превышающее 1 000 000.

*Пример входных данных*:

```
6
30
6
5
3
4
300
12
```

*Пример выходных данных для приведённого выше примера входных данных*:

```
Вычисленное контрольное значение: 12
Контроль пройден
```

**17.** На спутнике «Восход» установлен прибор, предназначенный для измерения солнечной активности. В течение времени эксперимента (это время известно заранее) прибор каждую минуту передаёт в обсерваторию по каналу связи положительное целое число, не превышающее 1000, — количество энергии солнечного излучения, полученной за последнюю минуту, измеренное в условных единицах.

После окончания эксперимента передаётся контрольное значение — наибольшее число R, удовлетворяющее следующим условиям:

1) R — произведение двух чисел, переданных в разные минуты;
2) R делится на 26.

Предполагается, что удовлетворяющее условиям контрольное значение существовало в момент передачи.

В результате помех при передаче как сами числа, так и контрольное значение могут быть искажены.

Напишите эффективную по времени и используемой памяти программу (укажите используемую версию языка программирования, например Free Pascal 2.6.4), которая будет проверять правильность контрольного значения. 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.

Программа должна напечатать отчёт по следующей форме.

Вычисленное контрольное значение: …
Контроль пройден (или Контроль не пройден)

Если удовлетворяющее условию контрольное значение определить невозможно, то выводится только фраза «Контроль не пройден». Перед текстом программы кратко опишите используемый Вами алгоритм решения.

На вход программе в первой строке подаётся количество чисел $1 < N \leq 100\,000$. В каждой из последующих N строк записано одно положительное целое число, не превышающее 1000. В последней строке записано контрольное значение.

Пример входных данных:

```
5
52
12
39
55
23
2860
```

Пример выходных данных для приведённого выше примера входных данных:

```
Вычисленное контрольное значение: 2860
Контроль пройден
```

**18.** На вход программы поступает последовательность из N целых положительных чисел, все числа в последовательности различны. Рассматриваются все пары различных элементов последовательности (элементы пары не обязаны стоять в последовательности рядом, порядок элементов в паре не важен). Необходимо определить количество пар, для которых произведение элементов делится на 26.

**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 целое положительное число, не превышающее 10 000. В качестве результата программа должна напечатать одно число: количество пар, в которых произведение элементов кратно 26.

*Пример входных данных*:

```
4
2
6
13
39
```

*Пример выходных данных для приведённого выше примера входных данных*:

```
4
```

*Пояснение.* Из четырёх заданных чисел можно составить 6 попарных произведений: 2·6, 2·13, 2·39, 6·13, 6·39, 13·39 (результаты: 12, 26, 78, 78, 234, 507). Из них на 26 делятся 4 произведения (2·13=26; 2·39=78; 6·13=78; 6·39=234).

Требуется написать эффективную по времени и по памяти программу для решения описанной 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 Кбайт 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 – 3 балла.

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

Вы можете сдать одну программу или две программы решения задачи (например, одна из программ может быть менее эффективна). Если Вы сдадите две программы, то каждая из них будет оцениваться независимо от другой, итоговой станет бóльшая из двух оценок.

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

**19.** На вход программы поступает последовательность из *N* целых положительных чисел, все числа в последовательности различны. Рассматриваются все пары различных элементов последовательности (элементы пары не обязаны стоять в последовательности рядом, порядок элементов в паре не важен). Необходимо определить количество пар, для которых произведение элементов не делится на 34.

**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 целое положительное число, не превышающее 10 000. В качестве результата программа должна напечатать одно число: количество пар, в которых произведение элементов не кратно 34.

*Пример входных данных*:

```
5
3
4
10
11
17
```

*Пример выходных данных для приведённого выше примера входных данных*:

```
8
```

*Пояснение.* Из пяти заданных чисел можно составить 10 попарных произведений: 3·4, 3·10, 3·11, 3·17, 4·10, 4·11, 4·17, 10·11, 10·17, 11·17 (результаты: 12, 30, 33, 51, 40, 44, 68, 110, 170, 187). Из них на 34 не делятся 8 произведений (3·4=12, 3·10=30, 3·11=33, 3·17=51, 4·10=40, 4·11=44, 10·11=110, 11·17=187).

Требуется написать эффективную по времени и по памяти программу для решения описанной 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 Кбайт 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 — 3 балла.

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

Вы можете сдать одну программу или две программы решения задачи (например, одна из программ может быть менее эффективна). Если Вы сдадите две программы, то каждая из них будет оцениваться независимо от другой, итоговой станет бóльшая из двух оценок.

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

**20.** На вход программы поступает последовательность из N целых положительных чисел, все числа в последовательности различны. Рассматриваются все пары различных элементов последовательности (элементы пары не обязаны стоять в последовательности рядом, порядок элементов в паре не важен). Необходимо определить количество пар, для которых произведение элементов не кратно 14.

**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 целое положительное число, не превышающее 1000. В качестве результата программа должна напечатать одно число: количество пар, в которых произведение элементов не кратно 14.

*Пример входных данных*:

```
4
2
6
5
42
```

*Пример выходных данных для приведённого выше примера входных данных*:

```
3
```

*Пояснение.* Из четырёх заданных чисел можно составить 6 попарных произведений: $2 \cdot 6$, $2 \cdot 5$, $2 \cdot 42$, $6 \cdot 5$, $6 \cdot 42$, $5 \cdot 42$. Из них на 14 не делятся 3 произведения ($2 \cdot 6$, $2 \cdot 5$, $6 \cdot 5$).

Требуется написать эффективную по времени и по памяти программу для решения описанной 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 Кбайт 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 — 3 балла.

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

Вы можете сдать **одну** программу или **две** программы решения задачи (например, одна из программ может быть менее эффективна). Если Вы сдадите две программы, то каждая из них будет оцениваться независимо от другой, итоговой станет **бóльшая** из двух оценок.

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

**21.** На вход программы поступает последовательность из $N$ целых положительных чисел, все числа в последовательности различны. Рассматриваются все пары различных элементов последовательности (элементы пары не обязаны стоять в последовательности рядом, порядок элементов в паре неважен). Необходимо определить количество пар, для которых произведение элементов кратно 62.

**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 10000. В качестве результата программа должна вывести одно число: количество пар, в которых произведение элементов кратно 62.

Пример входных данных:

```
5
2
6
13
31
93
```

Пример выходных данных для приведённого выше примера входных данных:

```
4
```

*Пояснение.* Из 5 чисел можно составить 4 пары, удовлетворяющие условию. Для заданного набора чисел получаем пары (2, 31), (2, 93), (6, 31), (6, 93).

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 килобайта 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.

**22.** На вход программы поступает последовательность из N целых положительных чисел, все числа в последовательности различны. Рассматриваются все пары различных элементов последовательности (элементы пары могут быть расположены в последовательности не рядом, порядок элементов в паре неважен). Необходимо определить количество пар, для которых произведение элементов делится без остатка на 10.

**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 целое положительное число, не превышающее 10 000.

В качестве результата программа должна напечатать одно число: количество пар, в которых произведение элементов кратно 10.

Пример входных данных:

```
4
2
6
5
15
```

Пример выходных данных для приведённого выше примера входных данных:

```
4
```

*Пояснение*. Из четырёх заданных чисел можно составить 6 попарных произведений: $2 \cdot 6$, $2 \cdot 5$, $2 \cdot 15$, $6 \cdot 5$, $6 \cdot 15$, $5 \cdot 15$ (результаты: 12, 10, 30, 30, 90, 75). Из них на 10 без остатка делятся 4 произведения ($2 \cdot 5 = 10$; $2 \cdot 15 = 30$; $6 \cdot 5 = 30$; $6 \cdot 15 = 90$).

Требуется написать эффективную по времени и памяти программу для решения описанной 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 килобайта 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 памяти, — 4 балла.

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

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

Вы можете сдать одну программу или две программы решения задачи (например, одна из программ может быть менее эффективна). Если Вы сдадите две программы, то каждая из них будет оцениваться независимо от другой, итоговой станет бóльшая из двух оценок.

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

**23.** Дана последовательность $N$ целых положительных чисел. Рассматриваются все пары элементов последовательности, разность которых чётна, и в этих парах, по крайней мере, одно из чисел пары делится на 19. Порядок элементов в паре неважен. Среди всех таких пар нужно найти и вывести пару с максимальной суммой элементов. Если одинаковую максимальную сумму имеет несколько пар, можно вывести любую из них. Если подходящих пар в последовательности нет, нужно вывести два нуля.

Описание входных и выходных данных.

В первой строке входных данных задаётся количество чисел $N$ ($2 \le N \le 10\ 000$). В каждой из последующих $N$ строк записано одно натуральное число, не превышающее 10 000.

*Пример входных данных:*

```
5
38
12
57
16
57
```

*Пример выходных данных для приведённого выше примера входных данных*:

```
57 57
```

Пояснение. Из данных пяти чисел можно составить три различные пары, удовлетворяющие условию: (38, 12), (38, 16), (57, 57). Наибольшая сумма получается в паре (57, 57). Эта пара допустима, так как число 57 встречается в исходной последовательности дважды.

Требуется написать эффективную по времени и памяти программу для решения описанной задачи.

Программа считается эффективной по времени, если при одновременном увеличении количества элементов последовательности $n$ и параметра $m$ 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 4 килобайта 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 памяти, — 4 балла. Максимальная оценка за правильную программу, возможно, неэффективную по памяти или время выполнения которой существенно зависит от величины $m$, — 3 балла.

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

Вы можете сдать одну программу или две программы решения задачи (например, одна из программ может быть менее эффективна). Если Вы сдадите две программы, то каждая из них будет оцениваться независимо от другой, итоговой станет бóльшая из двух оценок.

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

**24.** На вход программы поступает последовательность из N целых положительных чисел. Рассматриваются все пары различных элементов последовательности (элементы пары не обязаны стоять в последовательности рядом, порядок элементов в паре не важен). Необходимо вывести пару элементов, разность которых четна, а сумма максимальна. При этом по крайней мере одно число в паре делится на 31. Если таких пар несколько, то можно вывести любую из них. Если найти такую пару невозможно, то нужно вывести два нуля.

**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 целое положительное число, не превышающее 10 000. В качестве результата программа должна напечатать два числа.

Требуется написать эффективную по времени и по памяти программу для решения описанной 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 Кбайт 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 — 3 балла. Максимальная оценка за правильную программу, не удовлетворяющую требованиям эффективности, — 2 балла. Вы можете сдать одну программу или две программы решения задачи (например, одна из программ может быть менее эффективна). Если Вы сдадите две программы, то каждая из них будет оцениваться независимо от другой, итоговой станет бóльшая из двух оценок.

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

**25.** На вход программы поступает последовательность из $N$ натуральных чисел. Рассматриваются все пары различных элементов последовательности, у которых различные остатки от деления на $d = 160$ и хотя бы одно из чисел делится на $p = 7$. Среди таких пар, необходимо найти и вывести пару с максимальной суммой элементов.

**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 10000. В качестве результата программа должна напечатать элементы искомой пары. Если среди найденных пар максимальную сумму имеют несколько, то можно напечатать любую из них. Если таких пар нет, то вывести два нуля.

Пример входных данных:

```
4
168
7
320
328
```

Пример выходных данных для приведённого выше примера входных данных:

```
168 320
```

Пояснение: Из 4 чисел можно составить 6 пар. В данном случае условиям удовлетворяют пары: 168 и 320, 168 и 7, 320 и 7, 328 и 7. Максимальную сумму дает пара 168 и 320.

Требуется написать эффективную по времени и по памяти программу для решения описанной 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 Кбайт 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 — 3 балла. Максимальная оценка за правильную программу, не удовлетворяющую требованиям эффективности, — 2 балла. Вы можете сдать одну программу или две программы решения задачи (например, одна из программ может быть менее эффективна). Если Вы сдадите две программы, то каждая из них будет оцениваться независимо от другой, итоговой станет бóльшая из двух оценок.

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