МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Лабораторная работа №2  
по дисциплине «Вычислительная математика»

Численные методы решение систем линейных алгебраических уравнений

Вариант 13

Группа: АВТ-809  
Студент: Семёнов Б.В.  
Преподаватель: Балакин В.B.

НОВОСИБИРСК 2020

# Описание задания

## Цель работы

1. В соответствии с вариантом контрольного задания исследуйте существование и найдите решение системы уравнений от значений a, b, c, d (a, b, c, d - последние цифры номера зачетной книжки студента) с точностью не ниже 0,00001 тремя методами.

2. Написать программы, реализующие алгоритмы решения СЛАУ прямым и итерационным методом согласно варианту.

3. Для дублирования решений систем линейных алгебраических уравнений применить существующие стандартные функции МСАД, аналогичные заданным методам.

4. Сравнить методы и способы решения СЛАУ по быстродействию и точности, выбрать наиболее эффективный вычислительный процесс поставленной задачи.

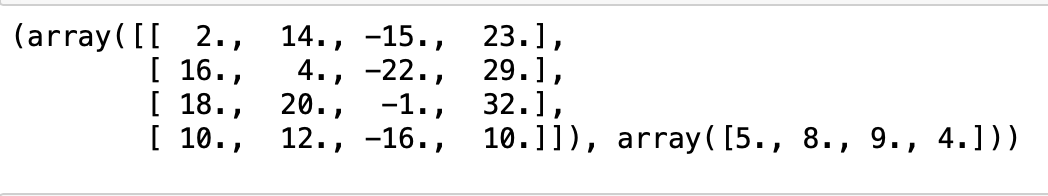
5. Проанализировать результаты работы и сделать выводы.

## 1.2. Исходные данные

Таблица 1. Исходные данные.

|  |  |
| --- | --- |
| Вариант | Методы |
| 13 | Метод обратной матрицы, Метод Гаусса с выбором главного элемента по столбцу, метод итерации. a,b,c,d = 1,3,0,9. |

Система уравнений, составленная согласно варианту:



# Теоретическое описание методов решения СЛАУ

## Метод Гаусса с выбором главного элемента по столбцу

Метод Гаусса, его еще называют методом Гауссовых исключений, состоит в том, что систему (2) приводят последовательным исключением неизвестных к эквивалентной системе с треугольной матрицей:

(2)

Предположим, что . Последовательно умножая первое уравнение на и складывая с *i*-м уравнение, исключим из всех уравнений кроме первого. Получим систему

где , ,

Аналогичным образом из полученной системы исключим . Последовательно, исключая все неизвестные, получим систему треугольного вида

Описанная процедура называется прямым ходом метода Гаусса. Заметим, что ее выполнение было возможно при условии, что все не равны нулю.

Выполняя последовательные подстановки в последней системе, (начиная с последнего уравнения) можно получить все значения неизвестных.

Эта процедура получила название обратный ход метода Гаусса. Обратный ход метода Гаусса преобразуют так эту ступенчатую матрицу, чтобы в первых  столбцах получилась единичная матрица:



Метод Гаусса может быть легко реализован на компьютере. При выполнении вычислений, как правило, не интересуют промежуточные значения матрицы *А*. Поэтому численная реализация метода сводится к преобразованию элементов массива размерности *(n×(n+1))*, где *n+1* столбец содержит элементы правой части системы.

Для контроля ошибки реализации метода используются так называемые контрольные суммы. Схема контроля основывается на следующем очевидном положении. Увеличение значения всех неизвестных на единицу равносильно замене данной системы контрольной системой, в которой свободные члены равны суммам всех коэффициентов соответствующей строки. Создадим дополнительный столбец, хранящий сумму элементов матрицы по строкам. На каждом шаге реализации прямого хода метода Гаусса будем выполнять преобразования и над элементами этого столбца, и сравнивать их значение с суммой по строке преобразованной матрицы. В случае не совпадения значений счет прерывается.

Один из важных факторов, предопределяющих выбор того или иного метода при решении конкретных задач, является вычислительная эффективность метода. Один из основных недостатков метода Гаусса связан с тем, что при его реализации накапливается вычислительная погрешность. Известно, что для больших систем порядка *n* число действий умножений и делений близко к n3/3.

Для того, чтобы уменьшить рост вычислительной погрешности применяются различные модификации метода Гаусса. Например, метод Гаусса с выбором главного элемента по столбцам, в этом случае на каждом этапе прямого хода строки матрицы переставляются таким образом, чтобы диагональный угловой элемент был максимальным. При исключении соответствующего неизвестного из других строк деление будет производиться на наибольший из возможных коэффициентов и, следовательно, относительная погрешность будет наименьшей.

Выполняемые в методе Гаусса преобразования прямого хода, приведшие матрицу ***А*** системы к треугольному виду позволяют вычислить определитель матрицы

.

Метод Гаусса позволяет найти обратную матрицу. Для этого необходимо решить матричное уравнение

,

где *Е*− единичная матрица. Его решение сводится к решению *n* систем

у вектора  *j* –я компонента равна единице, а остальные компоненты равны нулю.

Метод Гаусса может оказаться неустойчивым по отношению к росту вычислительной погрешности.

Теорема 1. *Для устойчивости метода Гаусса достаточно диагонального преобладания*, *т. е. выполнения неравенств*

| *aii* | > | *ai*1 | + | *ai*2 | +… + | *aii-1* | + | *aii+*1 || +… + | *ain* | + δ,

δ > 0, *i* = 1, 2, …, *n*.

При расчете на реальной ЭВМ с заданным числом разрядов, наряду с влиянием неточного задания входных данных на каждой арифметической операции, вносятся ошибки округления. Влияние последних на результат зависит не только от разрядности машины, но и от числа обусловленности матрицы системы, а также от выбранного алгоритма. Существуют алгоритмы, учитывающие влияние ошибок округления и позволяющие получить результат с гарантированной точностью, если система не обусловлена настолько плохо, что при расчете с заданной разрядностью эта точность не может быть гарантирована.

## Метод итераций

(1)

Предполагая, что диагональные коэффициенты***aii*** не равны 0 (i =1,2, …,n), разрешим первое уравнение системы (1) относительно **x1**, второе - относительно **x2** и т. д. Тогда получим эквивалентную систему

(3)

Где при *i* ≠*j* и ***αij***=0 при *i = j*  *(i,j =1,2,…,n)*.

Введя матрицы



систему (3) можно записать в матричной форме ***X = β +αX***, а любое (k+1) приближение вычисляется по формуле ***X(k+1) = β +α · X(k)***. Напишем формулы приближений в развернутом виде:

(4)

## Метод обратной матрицы.

1. Вычисляется определитель матрицы A. Если определитель равен нулю, то конец решения. Система имеет бесконечное множество решений.
2. При определителе отличном от нуля, находится обратная матрица.
3. Вектор решения получается умножением обратной матрицы на вектор результата.

# Корректность постановки задачи и понятие обусловленности

При использовании численных методов для решения тех или иных математических задач необходимо различать свойства самой задачи и свойства вычислительного алгоритма, предназначенного для ее решения.

Говорят, что задача поставлена корректно, если решение существует и единственно и, если оно непрерывно, зависит от входных данных. Последнее свойство называется также устойчивостью относительно входных данных.

Корректность исходной математической задачи еще не гарантирует хороших свойств численного метода ее решений и требует специального исследования.

Известно, что решение задачи (1) существует тогда и только тогда, когда . В этом случае можно определить обратную матрицу и решение записать в виде: .

Исследование устойчивость задачи (1) сводится к исследованию зависимости ее решения от правых частей  и элементов  матрицы А. Для того чтобы можно было говорить о непрерывной зависимости вектора решений от некоторых параметров, необходимо на множестве n - мерных векторов, принадлежащих линейному пространству L, ввести метрику.

В линейной алгебре предлагается определение множества метрик − норма

из которого легко получить наиболее часто используемые метрики

при р =1, ,

при , ,

при , .

Подчиненные нормы матриц определяемые как  , соответственно запишутся в следующем виде:

, , .

Обычно рассматривают два вида устойчивости решения системы (1): первый − по правым частям, второй − по коэффициентам системы(1) и по правым частям..

Наряду с исходной системой (1) рассмотрим систему с «возмущенными» правыми частями

,

где − возмущенная правая часть системы, а  возмущенное решение.

Можно получить оценку, выражающую зависимость относительной погрешности решения от относительной погрешности правых частей

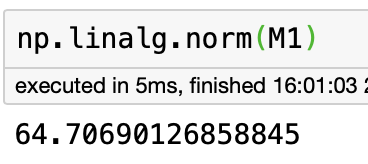
,

где − число обусловленности матрицы А ( в современной литературе это число обозначают как ) Если число обусловленности велико ( ~ ), то говорят, что матрица А плохо обусловлена. В этом случае малые возмущения правых частей системы (1), вызванные либо неточностью задания исходных данных, либо вызванные погрешностями вычисления существенно влияют на решение системы. Грубо говоря, если погрешность правых частей , то погрешность решения будет . Более подробно о свойствах числа обусловленности и оценка его величины можно прочитать в [3].

Если возмущение внесено в матрицу А, то для относительных возмущений решения запишем

.

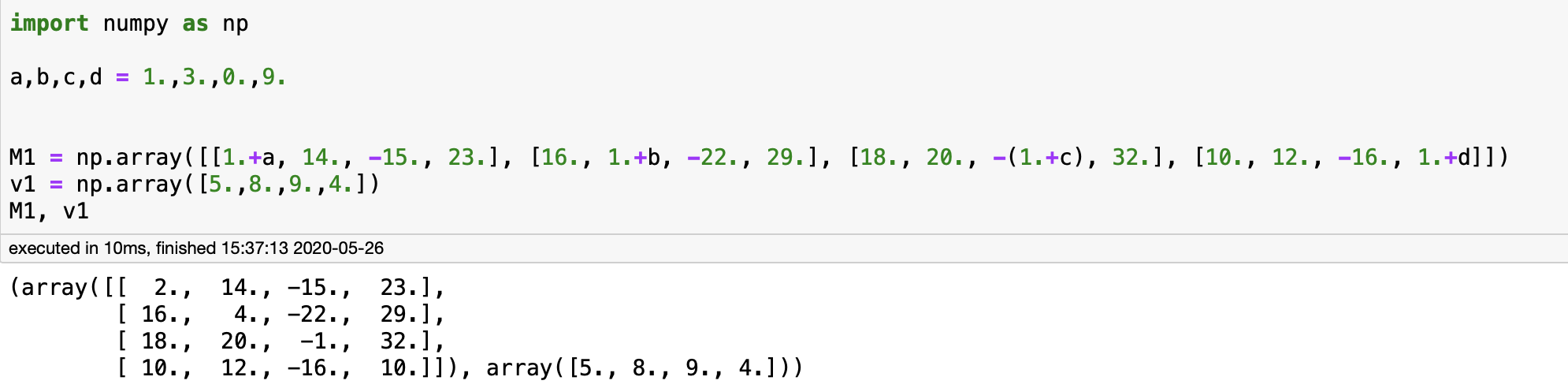
Рассчитаем число обусловленности для данной СЛАУ с помощью стандартных функций Python:



Функция numpy.linalg.norm(M) возвращает евклидову норму матрицы М.

Число обусловленности матрицы А (в современной литературе это число обозначают как *cond(A*). Если число обусловленности велико ( ~ ), то говорят, что матрица А плохо обусловлена. В нашем случае число обусловленности меньше 100, значит матрица не является плохо обусловленной.

# Исследование заданной системы уравнений, проверка условий сходимости и численные расчеты корней заданными методами

Для начала необходимо задать исходные данные. 

Проверим, что решение существует с помощью определителя матрицы.



Определитель не равен 0, следовательно, решение существует.

Мы установили, что решение существует, теперь проверим его на единственность.

Условие существования и единственности решения СЛАУ может служить условие:

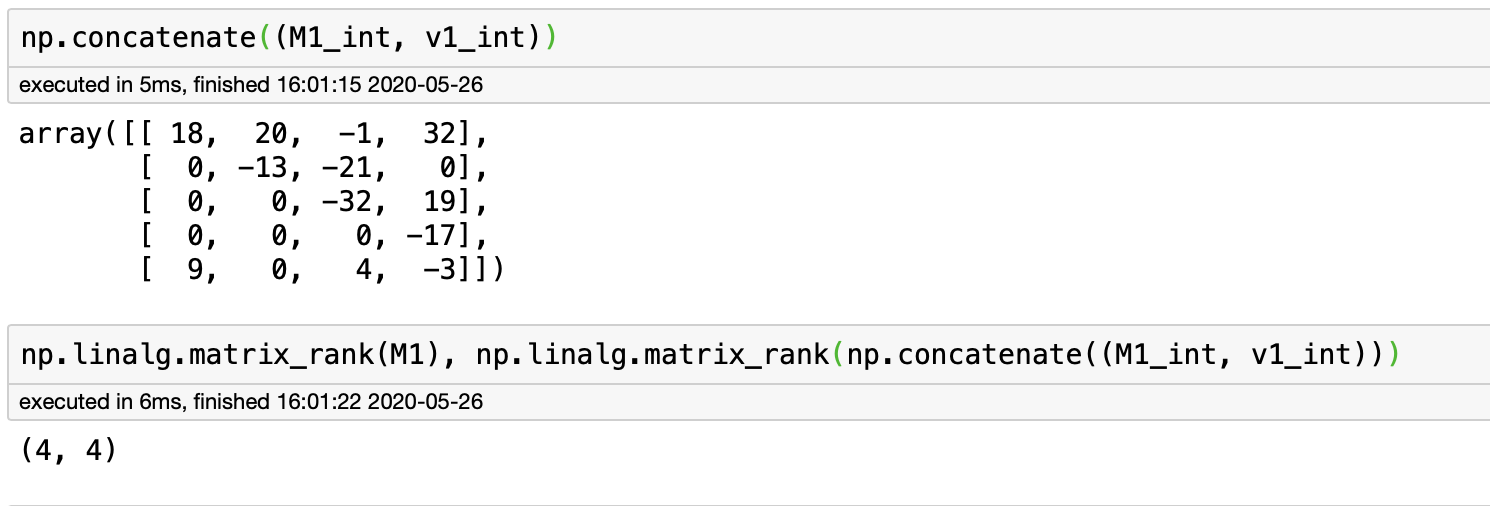
rank(M1) = 4, то есть ранг матрицы равен 4.

rows(M1) = 4, то есть число строк в матрице А равно 4.

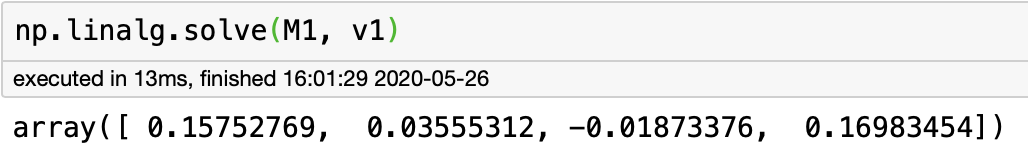
Проверка СЛАУ на совместимость.

rank(concatenate (M1,v1)) = 4. Сформирована матрица, в первых столбцах которой содержится матрица M1, а в последних – матрица v1 (матрицы M1 и v1 имеют одинаковое число строк.

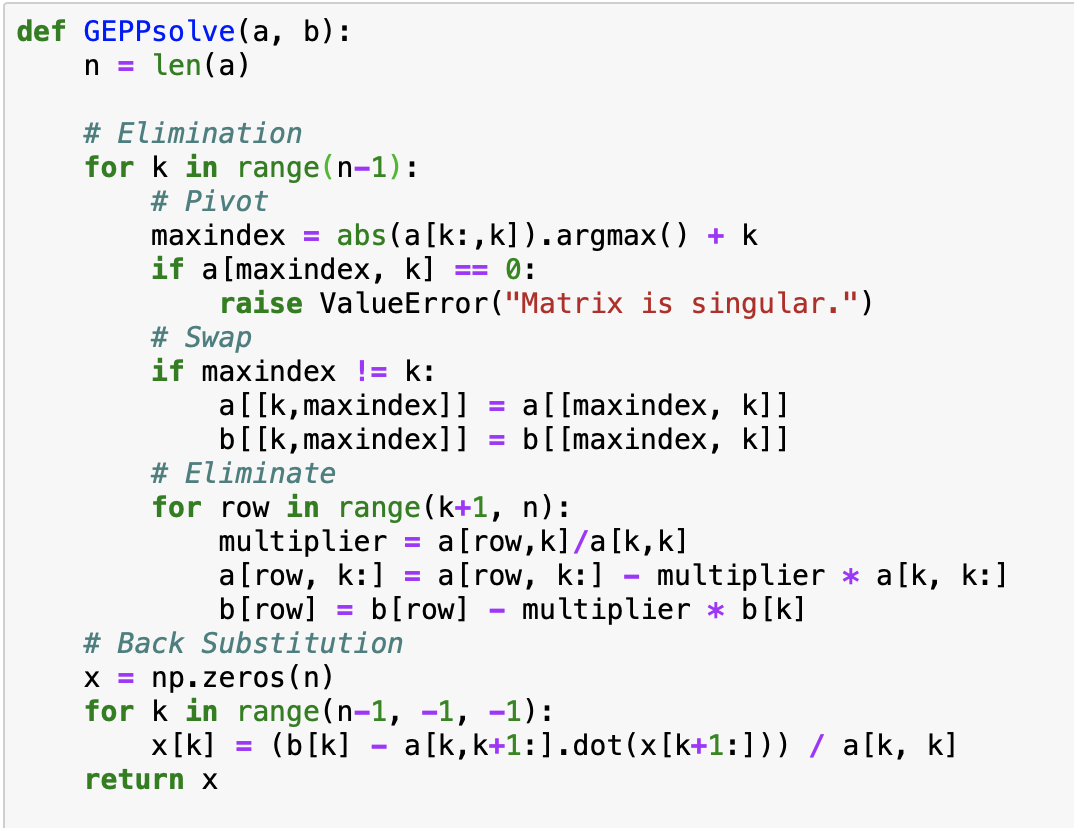
Так как ранги равны, СЛАУ имеет единственное решение

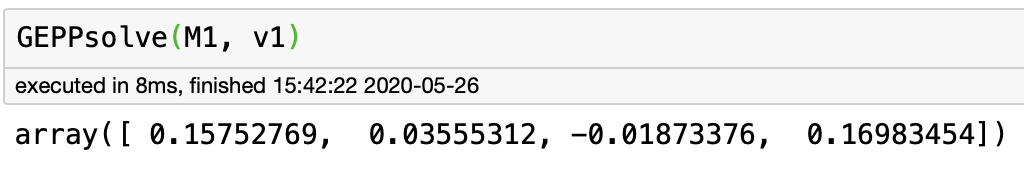


Найдем решение СЛАУ:



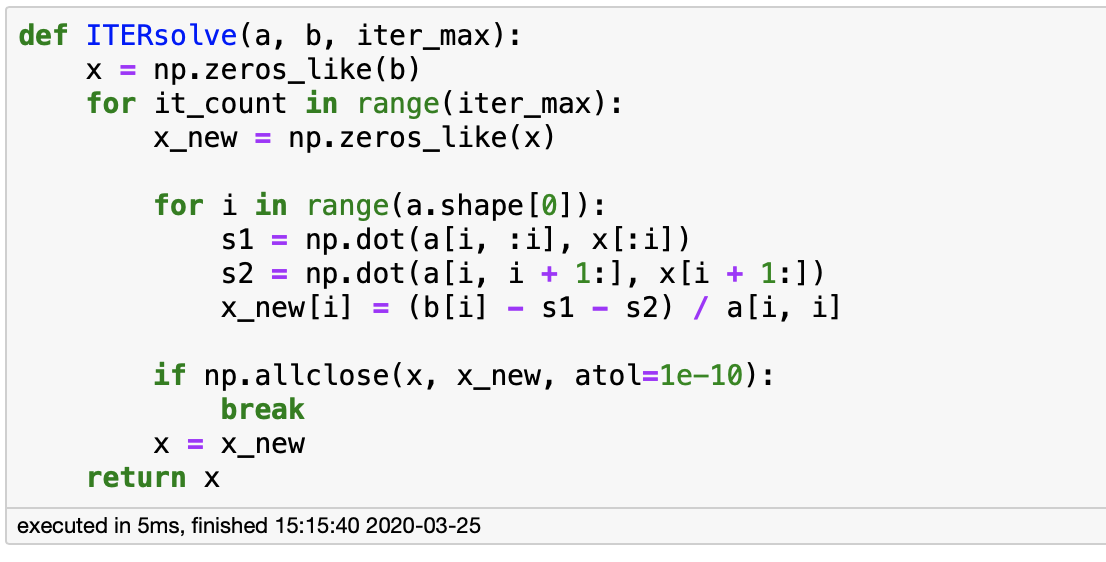
## Метод гаусса с выбором главного элемента по столбцу





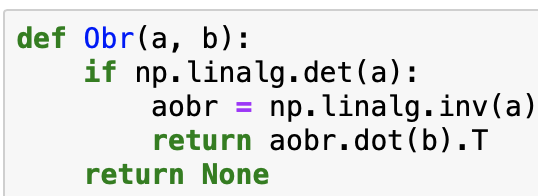
Функция находит главный элемент по столбцу j и меняет соответствующие строки, далее реализуется прямой ход до строки i и обратный ход. Трудоемкость для метода Гаусса: (где t - время выполнения одной операции). Можно наблюдать, что значения, полученные в результате вычислений, совпадают с вычислениями стандартной функции np.linalg.solve().

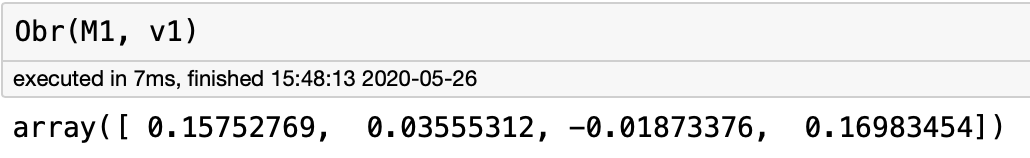
## Метод итераций





## Метод обратной матрицы





# Таблица результатов

Таблица 2. Результаты

|  |  |  |
| --- | --- | --- |
| Метод | Время (мкс) | Величина ошибки |
| GE (встроенный) | 10.95 | [0. 0. 0. 0.] |
| GE | 45.17 | [0. 0. 0. 0.] |
| Iter | 278.6 | [0. 0. 0. 0.] |
| Obr | 117.84 | [0.00000000e+00 2.22044605e-16 0.00000000e+00 0.00000000e+00] |

Из полученной таблицы можно увидеть, что на малых размерностях качество реализации может играть гораздо большую роль, чем алгоритмическая сложность. Также все методы дали гораздо меньшую погрешность, чем требовалось, так что с этой точки зрения к ним никаких претензий быть не может.

# Вывод

При решении СЛАУ малых размерностей (система 4-ого порядка) сложно на практике выявить сильные и слабые стороны различных методов. Но уже можно сказать, что применять итерационные методы в общем менее удобно из-за необходимости тщательного контроля входных данных, либо их преобразования.