

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

## Задание

Создать оконное приложение для решения СЛАУ методом Гаусса.

### Теоретический материал

Метод Гаусса – это метод для решения систем линейных алгебраических уравнений (СЛАУ). Он был разработан Карлом Фридрихом Гауссом и является одним из наиболее широко используемых методов решения линейных систем.

Основная идея метода Гаусса заключается в приведении системы линейных уравнений к эквивалентной системе, в которой каждое уравнение содержит одну из переменных, а остальные переменные исключены. Это достигается путем применения элементарных преобразований над уравнениями системы.

Этапы метода Гаусса:

Этап 1

Приводим систему к треугольному виду:

Шаг 1:

$$m_{i,1} = \frac{a_{i,1}}{a_{1,1}}; i = 2; 3; \dots; n$$

$$a_{i,j}^{(1)} = a_{i,j} = m_{i,1}a_{1,j}$$

$$b_i^{(1)} = b_i - m_{i,1}b_1; i, j = 2; 3; \dots; n$$

Шаг 2:

$$m_{i,2} = \frac{a_{i,2}^{(1)}}{a_{2,2}^{(1)}}; i = 3; 4; \dots; n$$

$$a_{i,j}^{(2)} = a_{i,j}^{(1)} = m_{i,2}a_{2,j}^{(1)}$$

$$b_i^{(2)} = b_i^{(1)} - m_{i,2}b_2^{(1)}; i, j = 3; 4; \dots; n$$

Шаг n-1:

$$m_{i,n-1} = \frac{a_{i,n-1}^{(n-2)}}{a_{n-1,n-1}^{(n-2)}}; i = 3; 4; \dots; n$$

$$a_{n,n}^{(n-1)} = a_{n,n}^{(n-2)} = m_{n,n-1}a_{n-1,n}^{(n-2)}$$

© МатБюро – Консультации по математике, программированию, экономике, праву, естественным наукам

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

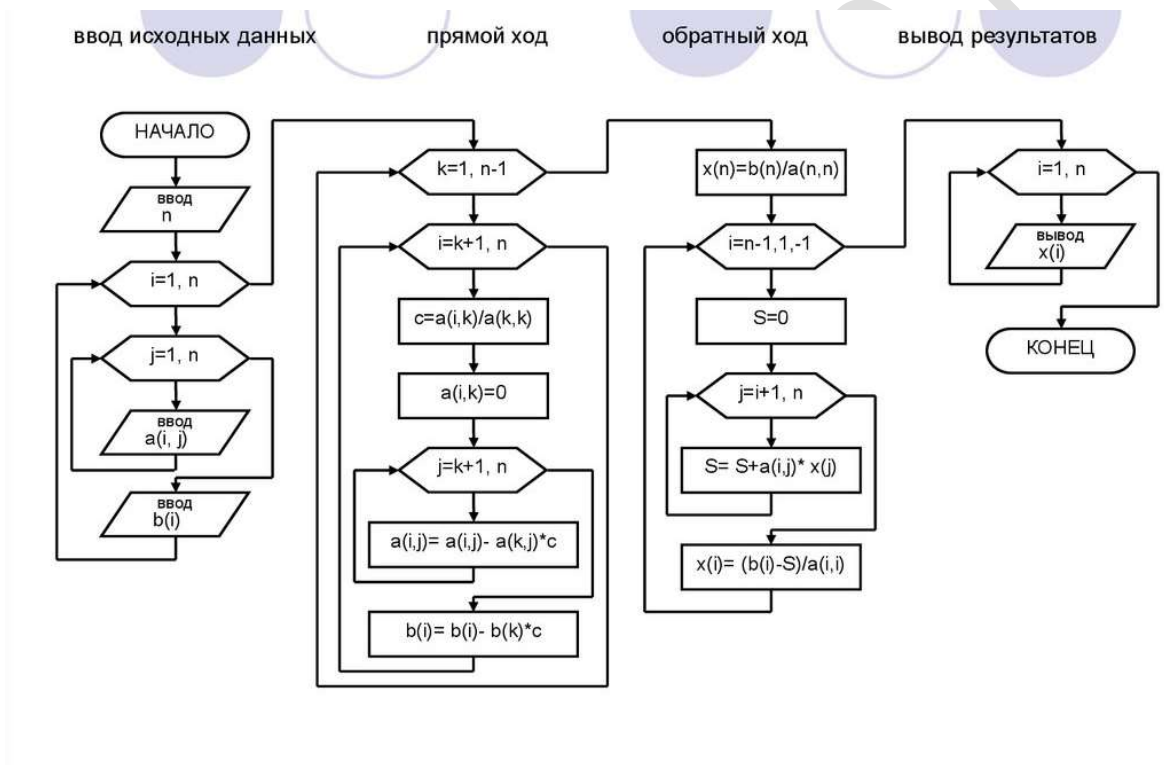
$$b_n^{(n-1)} = b_n^{(n-2)} - m_{n,n-1} b_{n-1}^{(n-2)};$$

## Этап 2

Вычисляем неизвестные:

$$x_n = \frac{b_n^{(n-1)}}{a_{nn}^{(n-1)}}; x_{n-i} = \frac{1}{a_{n-i,n-i}^{(n-i-1)}} \left( b_{n-i,n-i}^{(n-i-1)} - \sum_{j=0}^i a_{n-i,n-j}^{(n-i-1)} x_{n-j} \right); i = 1; 2; \dots; n - 1 \quad (2)$$

### Блок схема алгоритма



### Тестирование

Выполним добавление тестовых данных в поле ввода. Тестовые данные хранятся в программе в коде.

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

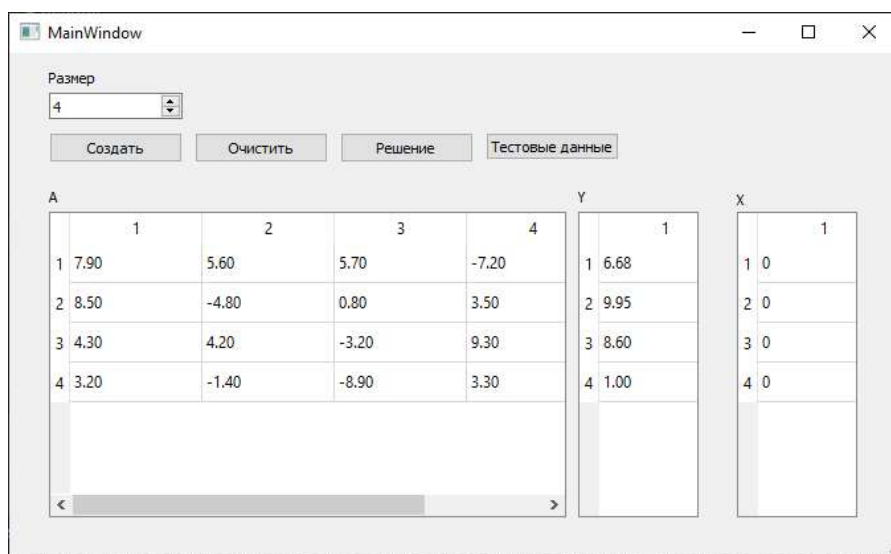


Рисунок 1. Добавление тестовых данных.

Выполним решение системы уравнений.

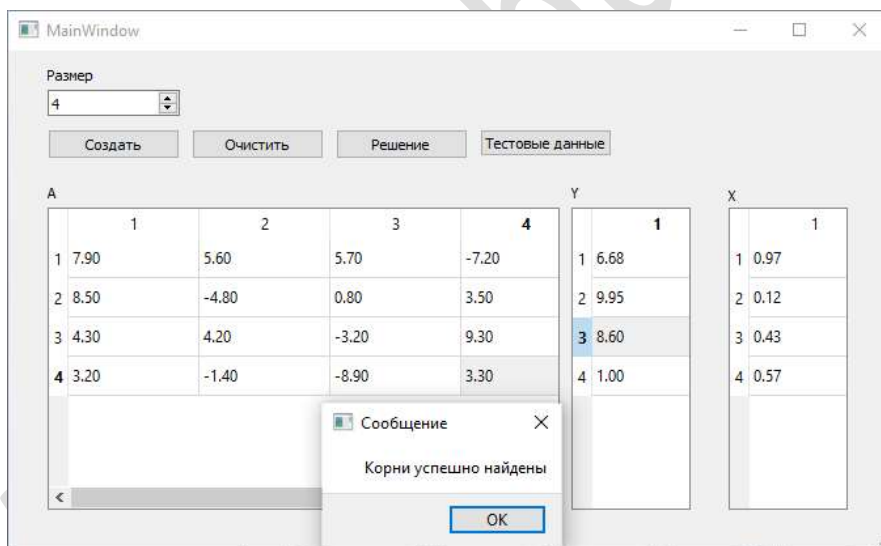


Рисунок 2. Решение системы уравнений

Изменим данные так, чтобы система имела одинаковые строки и не имела решений.

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

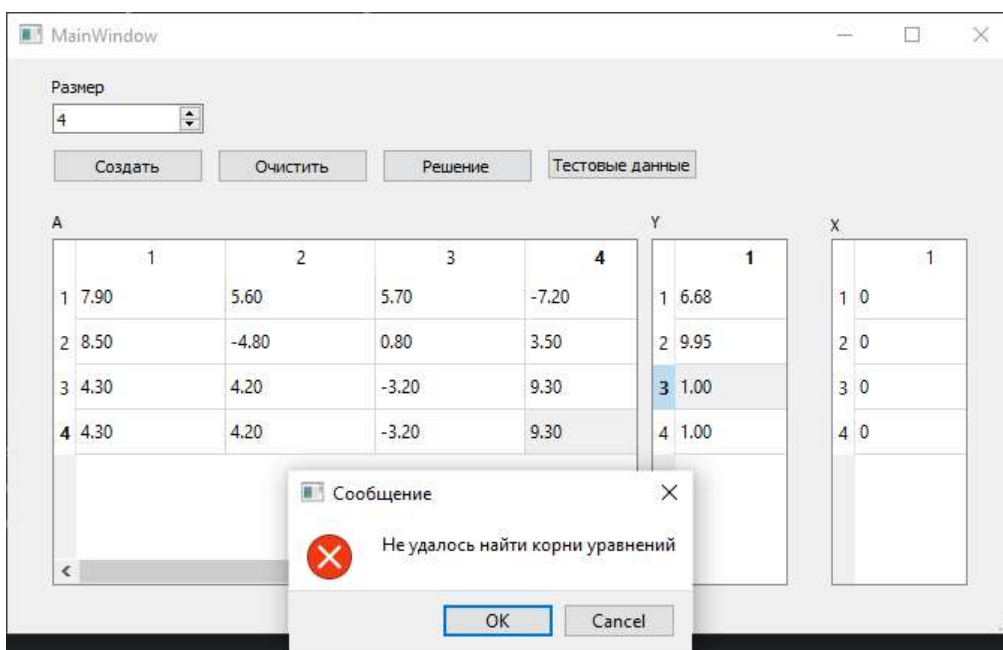


Рисунок 3. Обработка ситуации отсутствия решений.

Введем некорректные значения в поля ввода.

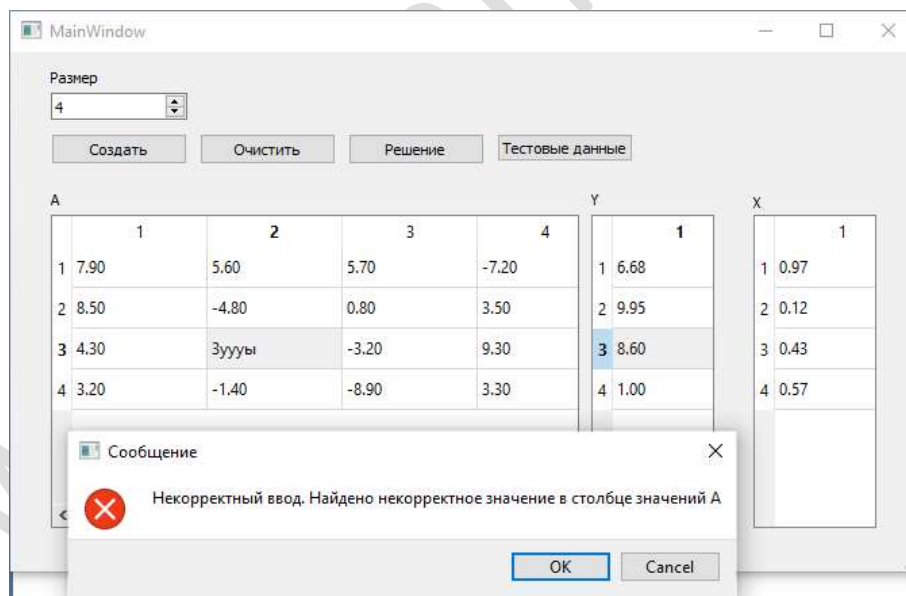


Рисунок 4. Ввод некорректных значений в поле ввода.

Программу успешно отработала все возможные варианты ситуаций.

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

**main.py**

```
import sys

from PyQt5 import uic, QtWidgets
from PyQt5.QtWidgets import QMainWindow, QMessageBox
from PyQt5.QtWidgets import QTableWidgetItem, QTableWidgetItem

class Matrix:
    """Класс матрицы для решения методом Гаусса"""
    ai: list[list[float]] # матрица коэффициентов
    bi: list[float] # матрица значений Y
    xs: list[float] # найденные корни
    rows: int
    cols: int
    n: int # размер матрицы

    def __init__(self, rows: int, cols: int, ai: list[list[float]] | None = None, bi: list[float] | None = None):
        """Создание матрицы"""
        self.ai = ai
        self.bi = bi
        self.xs = list([0 for _ in range(cols)])
        self.rows = rows
        self.cols = cols
        self.n = rows

    def gauss_method(self):
        """Метод гаусса"""
```

©МатБюро – Консультации по математике, программированию, экономике, праву, естественным наукам

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

```
# Преобразуем матрицу к треугольному виду с 1 на главной диагонали
(прямой ход метода Гаусса)

for i in range(self.n):

    max_el = abs(self.ai[i][i])

    max_row = i

    for k in range(i + 1, self.n):

        if abs(self.ai[k][i]) > max_el:

            max_el = abs(self.ai[k][i])

            max_row = k

    # Если весь столбец нулевой, то предоставлена некорректная СЛАУ

    if max_el == 0.0:

        self.xs = []

        return self.xs

    self.ai[i], self.ai[max_row] = self.ai[max_row], self.ai[i]

    self.bi[i], self.bi[max_row] = self.bi[max_row], self.bi[i]

    for k in range(i + 1, self.n):

        c = -self.ai[k][i] / self.ai[i][i]

        for j in range(i, self.n):

            if i == j:

                self.ai[k][j] = 0

            else:

                self.ai[k][j] += c * self.ai[i][j]

        self.bi[k] += c * self.bi[i]

# Найдем корни СЛАУ по преобразованной матрице (обратный ход метода
Гаусса)

self.xs = [0 for _ in range(self.n)]

try:

    for i in range(self.n - 1, -1, -1):

        self.xs[i] = self.bi[i] / self.ai[i][i]
```

©МатБюро – Консультации по математике, программированию, экономике, праву, естественным наукам

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

```
        for k in range(i - 1, -1, -1):
            self.bi[k] -= self.ai[k][i] * self.xs[i]
except:
    self.xs = []
return self.xs

class MainWindow(QMainWindow):
    """Основное окно приложения"""
    def __init__(self):
        # Обязательно нужно вызвать метод супер класса
        QMainWindow.__init__(self)
        self.ui = uic.loadUi('main.ui') # подключаем файл формы
        self.ui.show()

        # добавляем слоты для обработки нажатий на кнопки
        self.ui.b_init.clicked.connect(self.init_as)
        self.ui.b_clean.clicked.connect(self.clean)
        self.ui.b_solve.clicked.connect(self.solve)
        self.ui.b_test.clicked.connect(self.test_data)

    def init_as(self):
        """Создается поле заданного размера"""
        count = self.ui.input_count.value()
        self.init_table(self.ui.t_as, rows=count, columns=count)
        self.init_table(self.ui.t_bs, rows=count, columns=1)
        self.init_table(self.ui.t_xs, rows=count, columns=1)

    def clean(self):
        """Поле очищается"""
        self.clean_table(self.ui.t_as)
```

©МатБюро – Консультации по математике, программированию, экономике, праву, естественным наукам

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

```
self.clean_table(self.ui.t_bs)
self.clean_table(self.ui.t_xs)
...

def solve(self):
    """Решение системы"""
    count = self.ui.input_count.value()
    as_values = self.read_as()
    bs_values = self.read_bs()
    if not as_values or not bs_values:
        return False
    matrix = Matrix(rows=count, cols=count, ai=as_values, bi=bs_values)
    xs = matrix.gauss_method()
    if xs:
        self.write_xs(xs)
        QMessageBox.about(self,
                           "Сообщение",
                           "Корни успешно найдены")
    else:
        QMessageBox.critical(self,
                              "Сообщение",
                              "Не удалось найти корни уравнений",
                              QMessageBox.Ok | QMessageBox.Cancel)
    ...

def read_as(self) -> list[list[float]]:
    """Считывание таблицы коэффициентов А"""
    as_values = list[list[float]]()
    for r in range(self.ui.t_as.model().rowCount()):
        row = list[float]()
        for k in range(self.ui.t_as.model().columnCount()):
```



Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

```
try:
    a_value = float(self.ui.t_as.item(r, k).text())
except:
    QMessageBox.critical(self,
                          "Сообщение",
                          "Некорректный ввод. Найдено
некорректное значение в столбце значений A",
                          QMessageBox.Ok | QMessageBox.Cancel)

    return []
    row.append(a_value)
    as_values.append(row)
return as_values
```

```
def read_bs(self) -> list[float]:
    """Считывание таблицы результатов B"""
    bs_values = list[float]()
    rows = self.ui.t_bs.model().rowCount()
    for r in range(rows):
        try:
            b_value = float(self.ui.t_bs.item(r, 0).text())
        except:
            QMessageBox.critical(self,
                                  "Сообщение",
                                  "Некорректный ввод. Найдено некорректное
значение в столбце значений Y",
                                  QMessageBox.Ok | QMessageBox.Cancel)

            return []
            bs_values.append(b_value)
    return bs_values
```

```
def write_xs(self, xs: list[float]):
    """Заполнение поля с решениями уравнений """
```

©МатБюро – Консультации по математике, программированию, экономике, праву, естественным наукам

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

```
rows = self.ui.t_as.model().rowCount()
for i in range(rows):
    item = QTableWidgetItem(f"{xs[i]:.2f}")
    self.ui.t_xs.setItem(i, 0, item)
return True

def clean_table(self, table: QTableWidgetItem):
    """Очистить таблицу"""
    rows = table.rowCount()
    columns = table.columnCount()
    for i in range(rows):
        for k in range(columns):
            item = QTableWidgetItem("0")
            table.setItem(i, k, item)
    return True

def init_table(self, table: QTableWidgetItem, rows, columns=1):
    """Создать таблицу нужного размера"""
    table.setRowCount(rows)
    table.setColumnCount(columns)
    for r in range(rows):
        for k in range(columns):
            item = QTableWidgetItem("0")
            table.setItem(r, k, item)
    self.clean_table(table)

def test_data(self):

    ai = [ # исходные данные
          [7.9, 5.6, 5.7, -7.2],
          [8.5, -4.8, 0.8, 3.5],
```

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

```
[4.3, 4.2, -3.2, 9.3],
[3.2, -1.4, -8.9, 3.3]
]

rows = len(ai)
columns = len(ai[0])
self.init_table(self.ui.t_as, rows, columns)
self.init_table(self.ui.t_bs, rows, 1)
self.init_table(self.ui.t_xs, rows, 1)
self.clean_table(self.ui.t_xs)
for i in range(rows):
    for k in range(columns):
        item = QTableWidgetItem(f"{ai[i][k]:.2f}")
        self.ui.t_as.setItem(i, k, item)

bi = (6.68, 9.95, 8.6, 1)
for i in range(rows):
    item = QTableWidgetItem(f"{bi[i]:.2f}")
    self.ui.t_bs.setItem(i, 0, item)
self.ui.input_count.setValue(len(ai))
return True

if __name__ == '__main__':
    app = QtWidgets.QApplication([])
    application = MainWindow()
    sys.exit(app.exec())
```

**main.ui**

```
<?xml version="1.0" encoding="UTF-8"?>
<ui version="4.0">
  <class>MainWindow</class>
```

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

```
<widget class="QMainWindow" name="MainWindow">
  <property name="geometry">
    <rect>
      <x>0</x>
      <y>0</y>
      <width>673</width>
      <height>383</height>
    </rect>
  </property>
  <property name="windowTitle">
    <string>MainWindow</string>
  </property>
  <widget class="QWidget" name="centralwidget">
    <widget class="QFrame" name="frame">
      <property name="geometry">
        <rect>
          <x>20</x>
          <y>100</y>
          <width>511</width>
          <height>261</height>
        </rect>
      </property>
      <property name="frameShape">
        <enum>QFrame::StyledPanel</enum>
      </property>
      <property name="frameShadow">
        <enum>QFrame::Raised</enum>
      </property>
      <widget class="QLabel" name="label">
        <property name="geometry">
          <rect>
            <x>10</x>
            <y>0</y>
            <width>31</width>
            <height>16</height>
          </rect>
        </property>
        <property name="text">
          <string>A</string>
        </property>
      </widget>
      <widget class="QLabel" name="label_2">
        <property name="geometry">
          <rect>
            <x>410</x>
            <y>0</y>
            <width>21</width>
            <height>16</height>
          </rect>
        </property>
        <property name="maximumSize">
          <size>
            <width>243</width>
            <height>16777215</height>
          </size>
        </property>
        <property name="text">
```

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

```
<string>Y</string>
</property>
</widget>
<widget class="QTableWidget" name="t_as">
  <property name="geometry">
    <rect>
      <x>10</x>
      <y>20</y>
      <width>391</width>
      <height>231</height>
    </rect>
  </property>
</widget>
<widget class="QTableWidget" name="t_bs">
  <property name="geometry">
    <rect>
      <x>410</x>
      <y>20</y>
      <width>91</width>
      <height>231</height>
    </rect>
  </property>
</widget>
</widget>
<widget class="QFrame" name="frame_2">
  <property name="geometry">
    <rect>
      <x>540</x>
      <y>100</y>
      <width>111</width>
      <height>261</height>
    </rect>
  </property>
  <property name="frameShape">
    <enum>QFrame::StyledPanel</enum>
  </property>
  <property name="frameShadow">
    <enum>QFrame::Raised</enum>
  </property>
  <widget class="QLabel" name="label_3">
    <property name="geometry">
      <rect>
        <x>10</x>
        <y>0</y>
        <width>21</width>
        <height>20</height>
      </rect>
    </property>
    <property name="text">
      <string>X</string>
    </property>
  </widget>
  <widget class="QTableWidget" name="t_xs">
    <property name="geometry">
      <rect>
        <x>10</x>
        <y>20</y>
```

© МатБюро – Консультации по математике, программированию, экономике, праву, естественным наукам

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

```
<width>91</width>
<height>231</height>
</rect>
</property>
</widget>
</widget>
<widget class="QFrame" name="frame_3">
<property name="geometry">
<rect>
<x>20</x>
<y>10</y>
<width>471</width>
<height>80</height>
</rect>
</property>
<property name="frameShape">
<enum>QFrame::StyledPanel</enum>
</property>
<property name="frameShadow">
<enum>QFrame::Raised</enum>
</property>
<widget class="QLabel" name="label_4">
<property name="geometry">
<rect>
<x>10</x>
<y>0</y>
<width>51</width>
<height>16</height>
</rect>
</property>
<property name="text">
<string>Размер</string>
</property>
</widget>
<widget class="QSpinBox" name="input_count">
<property name="geometry">
<rect>
<x>10</x>
<y>20</y>
<width>101</width>
<height>20</height>
</rect>
</property>
</widget>
<widget class="QPushButton" name="b_init">
<property name="geometry">
<rect>
<x>10</x>
<y>50</y>
<width>101</width>
<height>23</height>
</rect>
</property>
<property name="text">
<string>Создать</string>
</property>
</widget>
```

Поможем вам с написанием программ: [www.matburo.ru/sub\\_subject.php?p=pz](http://www.matburo.ru/sub_subject.php?p=pz)

```
<widget class="QPushButton" name="b_clean">
  <property name="geometry">
    <rect>
      <x>120</x>
      <y>50</y>
      <width>101</width>
      <height>23</height>
    </rect>
  </property>
  <property name="text">
    <string>Очистить</string>
  </property>
</widget>
<widget class="QPushButton" name="b_solve">
  <property name="geometry">
    <rect>
      <x>230</x>
      <y>50</y>
      <width>101</width>
      <height>23</height>
    </rect>
  </property>
  <property name="text">
    <string>Решение</string>
  </property>
</widget>
<widget class="QPushButton" name="b_test">
  <property name="geometry">
    <rect>
      <x>340</x>
      <y>50</y>
      <width>101</width>
      <height>21</height>
    </rect>
  </property>
  <property name="text">
    <string>Тестовые данные</string>
  </property>
</widget>
</widget>
<widget class="QStatusBar" name="statusbar"/>
</widget>
<resources/>
<connections/>
</ui>
```