

# Основы программирования (Java)

ФИСТ 1 курс

Власенко

Олег

Федосович

**Лекция 4**

**Массивы**

## Массивы в Java – hard coded

```
public static void main(String[] args) {  
    System.out.println("Hi!");  
  
    int [] a1 = {1, 2, 3, 4};  
  
    for (int i = 0; i < a1.length; i++) {  
        System.out.printf("%d ", a1[i]);  
  
    }  
    System.out.println();  
}
```

# Массивы в Java – динамическое создание

```
private static final int ARRAY_SIZE = 5;
```

```
...
```

```
int [] a2 = new int[ARRAY_SIZE];
```

```
for (int i = 0; i < a2.length; i++) {  
    System.out.printf("%d ", a2[i]);  
}
```

```
System.out.println();
```

```
for (int i = 0; i < ARRAY_SIZE; i++) {  
    a2[i] = i * 10 + 1;  
}
```

## Вывод массива – for 2 видов

```
for (int i = 0; i < a2.length; i++) {  
    System.out.printf("%d ", a2[i]);  
  
}  
System.out.println();
```

```
for(int ai : a1) {  
    System.out.printf("%d ", ai);  
}  
System.out.println();
```

## For как foreach

```
for(int ai : a1) {  
    ai *= 10;  
}
```

```
for(int ai : a1) {  
    System.out.printf("%d ", ai);  
  
}  
System.out.println();
```

# Двумерный массив – hard coded

```
public static void main(String[] args) {  
    System.out.println("Hi!");  
  
    int [][] aa1 = {  
        {1, 2, 3, 4},  
        {10, 20, 30, 40},  
        {15, 25, 35, 45}  
    };  
  
    for (int i = 0; i < aa1.length; i++) {  
        for (int j = 0; j < aa1[i].length; j++) {  
            System.out.printf("%d\t", aa1[i][j]);  
        }  
        System.out.println();  
    }  
    System.out.println();  
}
```

# Двумерный массив – динамическое создание

```
private static final int N = 5;
private static final int M = 3;
...
int[][] aa2 = new int[N][M];

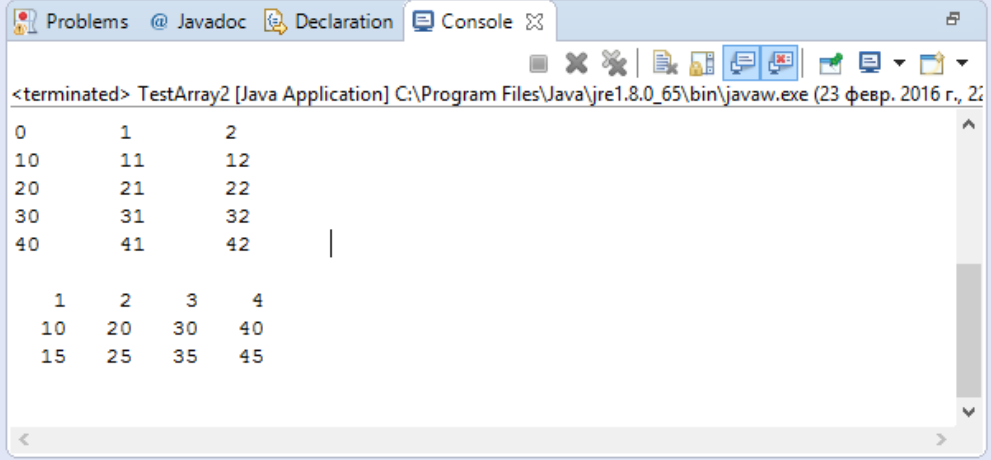
for (int i = 0; i < aa2.length; i++) {
    for (int j = 0; j < aa2[i].length; j++) {
        System.out.printf("%4d ", aa2[i][j]);
    }
    System.out.println();
}
System.out.println();

for (int i = 0; i < N; i++) {
    for (int j = 0; j < M; j++) {
        aa2[i][j] = i * 10 + j;
    }
}
```

# Двумерный массив – вывод в виде матрицы

```
for (int i = 0; i < aa2.length; i++) {  
    for (int j = 0; j < aa2[i].length; j++) {  
        System.out.printf("%d\t", aa2[i][j]);  
    }  
    System.out.println();  
}  
System.out.println();
```

```
for(int[] aa_i : aa2) {  
    for(int aa_ij : aa_i) {  
        System.out.printf("%4d ", aa_ij);  
    }  
    System.out.println();  
}  
System.out.println();
```



```
<terminated> TestArray2 [Java Application] C:\Program Files\Java\jre1.8.0_65\bin\javaw.exe (23 февр. 2016 г., 2:...
```

0	1	2
10	11	12
20	21	22
30	31	32
40	41	42

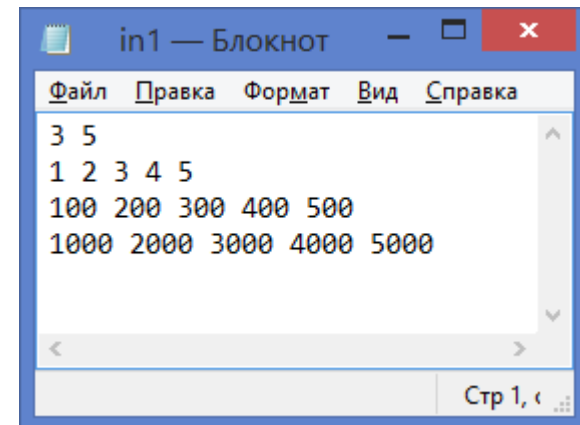
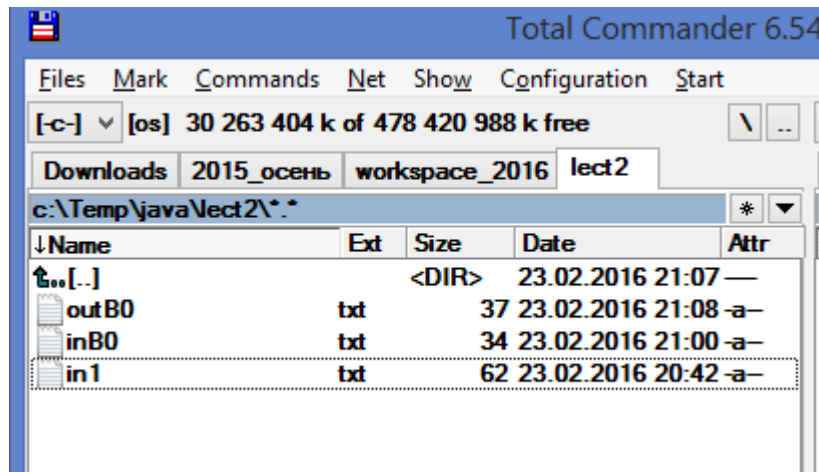
1	2	3	4
10	20	30	40
15	25	35	45



# Двумерный массив – ввод из текстового файла

## «Чтение матриц из текстового файл - Java SE»

<http://www.cyberforum.ru/java-j2se/thread269776.html>



## Двумерный массив – ввод из текстового файла (2)

```
import java.io.FileInputStream;
```

```
import java.io.IOException;
```

```
public class TestArray2File {
```

```
    private static int currentIndex = -1;
```

```
    private static int next(String numbers[]) {
```

```
        ++currentIndex;
```

```
        while (currentIndex < numbers.length  
                && numbers[currentIndex].equals(""))
```

```
            ++currentIndex;
```

```
        return currentIndex < numbers.length
```

```
            ? Integer.parseInt(numbers[currentIndex])
```

```
            : null;
```

```
    }
```

## Двумерный массив – ввод из текстового файла (3)

```
public static void main(String args[]) throws IOException {  
    FileInputStream inFile = new  
        FileInputStream("C:\\Temp\\java\\lect2\\in1.txt");  
  
    byte[] str = new byte[inFile.available()];  
    inFile.read(str);  
    inFile.close();  
  
    String text = new String(str);  
  
    String[] numbers = text.split("\\D");
```

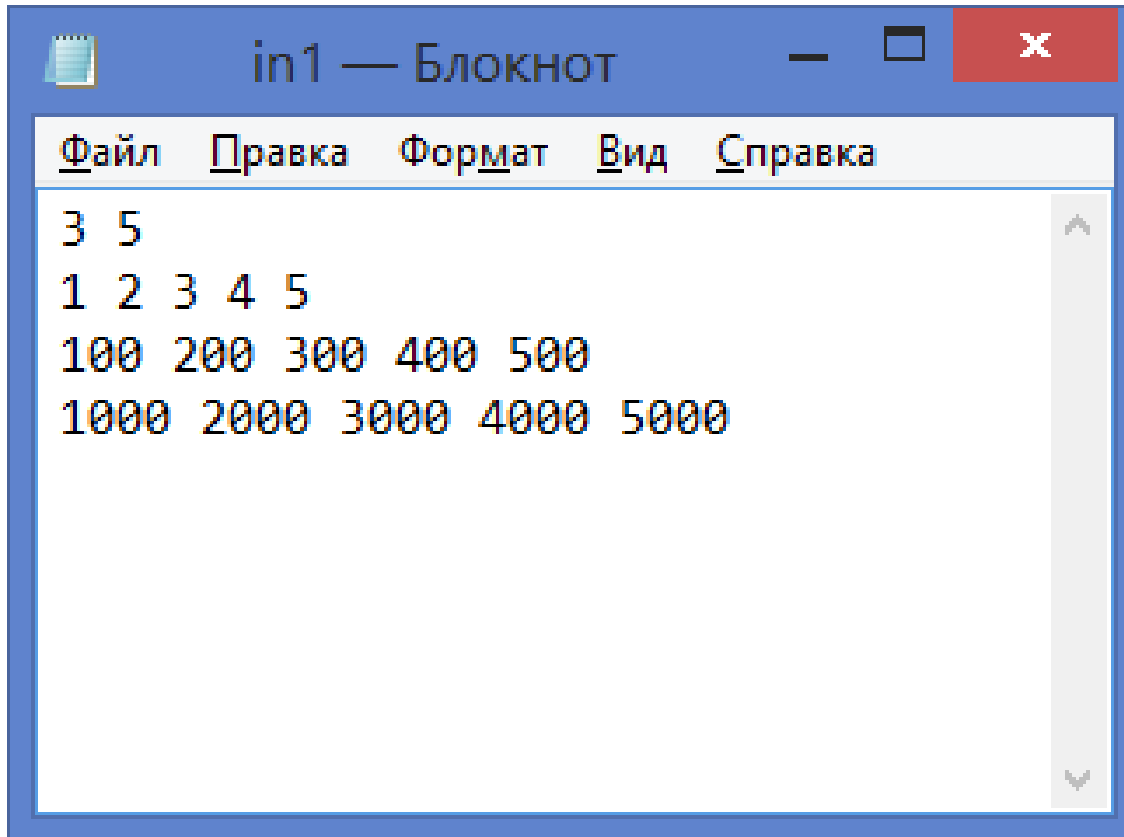
## Двумерный массив – ввод из текстового файла (4)

```
int n = next(numbers);
int m = next(numbers);
int matr[][] = new int[n][m];
for (int i = 0; i < n; ++i) {
    for (int j = 0; j < m; ++j) {
        matr[i][j] = next(numbers);
    }
}

for (int i = 0; i < n; ++i) {
    for (int j = 0; j < m; ++j) {
        System.out.printf("%6d ", matr[i][j]);
    }
    System.out.println();
}
System.out.println();
}
}
```

# Двумерный массив – ввод из текстового файла (5)

Содержимое файла "C:\\Temp\\java\\lect2\\in1.txt"



```
3 5
1 2 3 4 5
100 200 300 400 500
1000 2000 3000 4000 5000
```

# Экзаменационная задача В0

## Общее задание

- Ввести двумерный массив из файла. Количество элементов не более 10x10.
- Каждый элемент – целое число в интервале значений -1000..+1000.
- Количество строк (N) и столбцов (M) задано первой строке входного файла. Далее в N строках записаны по M числу.
- Обработать массив согласно варианту.
- Сохранить результат в формате, аналогичном входному.

Номер варианта	Задание	Входные данные (вводятся из текстового файла)	Выходные данные (выводятся в текстовый файл)
В0	Переставить столбцы, содержащие минимальный и максимальный элементы	4 3 0 2 3 1 3 4 2 -1 5 3 4 6	4 3 0 3 2 1 4 3 2 5 -1 3 6 4

# Экзаменационная задача B0 (2)

```
import java.io.File;  
import java.io.FileInputStream;  
import java.io.FileNotFoundException;  
import java.io.IOException;  
import java.io.PrintWriter;
```

```
public class ExamB0Solution {
```

```
    static int [][] array;  
    static int n;  
    static int m;
```

```
    public static void main(String[] args) {  
        if (!loadArray("C:\\Temp\\java\\lect2\\inB0.txt")) {  
            System.out.println("File has not been loaded");  
            return;  
        }
```

```
        printArray();  
        replaceColumns();  
        printArray();  
        saveArray("C:\\Temp\\java\\lect2\\outB0.txt");
```

```
}
```

## Экзаменационная задача B0 (3)

```
private static int currentIndex = -1;
```

```
private static int next(String [] numbers) {  
    ++currentIndex;  
    while (currentIndex < numbers.length  
           && numbers[currentIndex].equals(""))  
        ++currentIndex;  
    return currentIndex < numbers.length  
           ? Integer.parseInt(numbers[currentIndex]) : null;  
}
```

```
private static boolean loadArray(String fileName) {  
    FileInputStream inFile;  
    try {  
        inFile = new FileInputStream(fileName);  
  
        byte[] str = new byte[inFile.available()];  
        inFile.read(str);  
        inFile.close();  
    }  
}
```



# Экзаменационная задача В0 (4)

```
String text = new String(str);
String[] numbers = text.split("\\D");

n = next(numbers);
m = next(numbers);
array = new int[n][m];

for (int i = 0; i < n; ++i) {
    for (int j = 0; j < m; ++j) {
        array[i][j] = next(numbers);
    }
}
return true;
} catch (FileNotFoundException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
    return false;
} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
    return false;
}
}
```

# Экзаменационная задача В0 (5)

```
/*
 * Переставить столбцы, содержащие минимальный и максимальный элементы
 */
private static void replaceColumns() {
    int maxJ = 0;
    int minJ = 0;
    int min = array[0][0];
    int max = array[0][0];
    for (int i = 0; i < n; ++i) {
        for (int j = 0; j < m; ++j) {
            if (array[i][j] > max) {
                max = array[i][j];
                maxJ = j;
            }
            if (array[i][j] < min) {
                min = array[i][j];
                minJ = j;
            }
        }
    }
}
```

## Экзаменационная задача B0 (6)

```
for (int i = 0; i < n; ++i) {  
    int tmp = array[i][minJ];  
    array[i][minJ] = array[i][maxJ];  
    array[i][maxJ] = tmp;  
}  
}
```

## Экзаменационная задача В0 (7)

```
private static void printArray() {  
    for (int i = 0; i < n; ++i) {  
        for (int j = 0; j < m; ++j) {  
  
            System.out.printf("%4d ", array[i][j]);  
        }  
        System.out.println();  
    }  
    System.out.println();  
}
```

# Экзаменационная задача В0 (8)

«Как записывать в файл?» <http://devcolibri.com/1141>

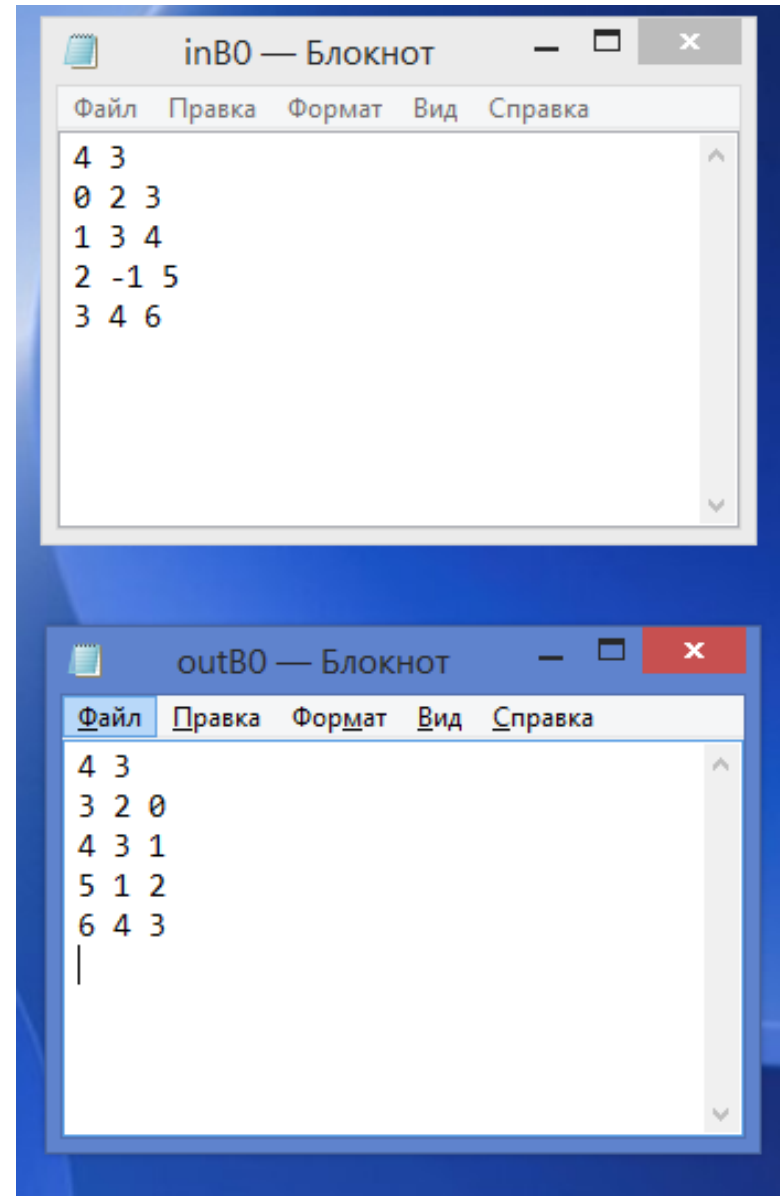
```
private static void saveArray(String fileName) {  
    //Определяем файл  
    File file = new File(fileName);  
    try {  
        // проверяем, что если файл не существует  
        // то создаем его  
        if(!file.exists()){  
            file.createNewFile();  
        }  
  
        //PrintWriter обеспечит возможности записи в файл  
        PrintWriter out = new PrintWriter(file.getAbsolutePath());
```

# Экзаменационная задача B0 (9)

```
try {  
    out.println(n + " " + m);  
    for (int i = 0; i < n; ++i) {  
        for (int j = 0; j < m; ++j) {  
            out.printf("%d ", array[i][j]);  
        }  
        out.println();  
    }  
    System.out.println();  
} finally {  
    //После чего мы должны закрыть файл  
    //Иначе файл не запишется  
    out.close();  
}  
} catch (IOException e) {  
    throw new RuntimeException(e);  
}  
}
```

# Экзаменационная задача В0 (10)

Входной и выходной файл

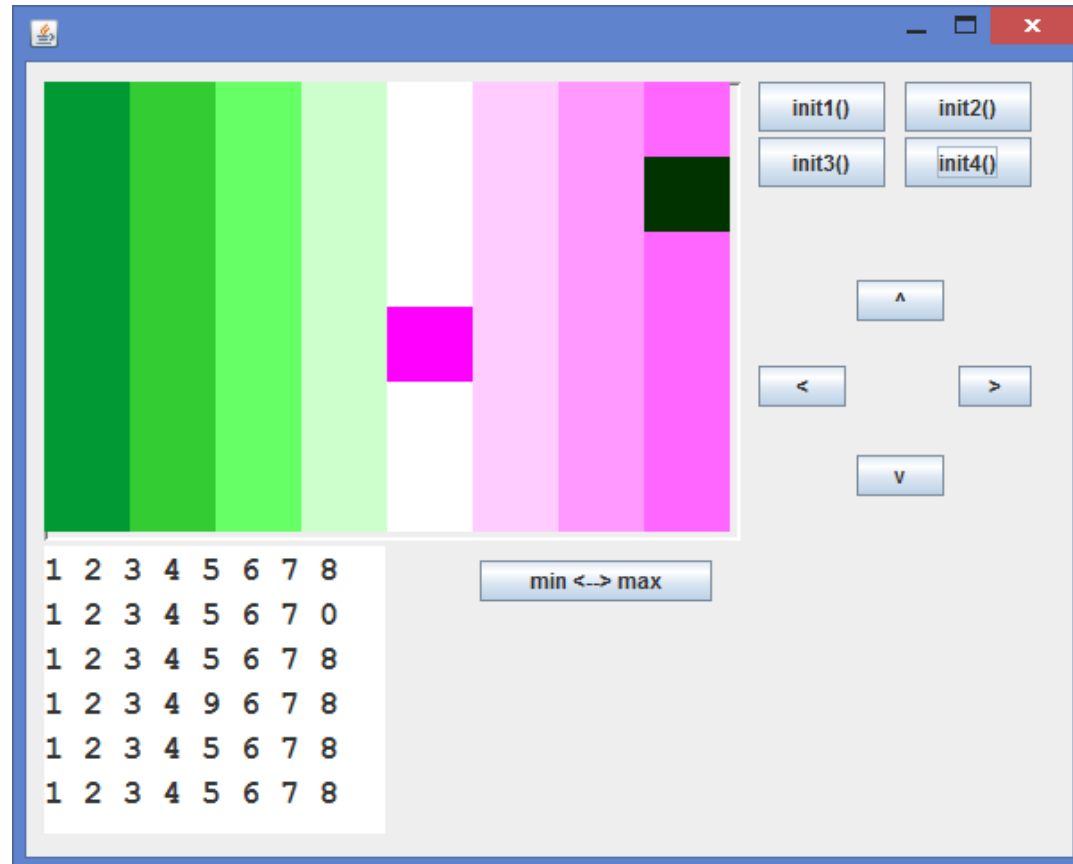


# Демо – визуализация массива в Java

Исследуем приложение, которое визуализирует обработку массива.

Предусмотрим следующие возможности («фичи» = **features**)

- 1) Несколько способов инициализации массива
- 2) Сдвиг элементов влево, вправо, вверх, вниз
- 3) Задача B0 – меняем местами столбцы с минимальным и максимальным элементами





# Класс панели для визуализации массива

```
import java.awt.Graphics;
```

```
import javax.swing.JPanel;
```

```
public class ArrayPanel extends JPanel {
```

```
    ArrayField arr;
```

```
    public ArrayPanel(ArrayField array) {
```

```
        arr = array;
```

```
    }
```

```
    public void paint(Graphics g) {
```

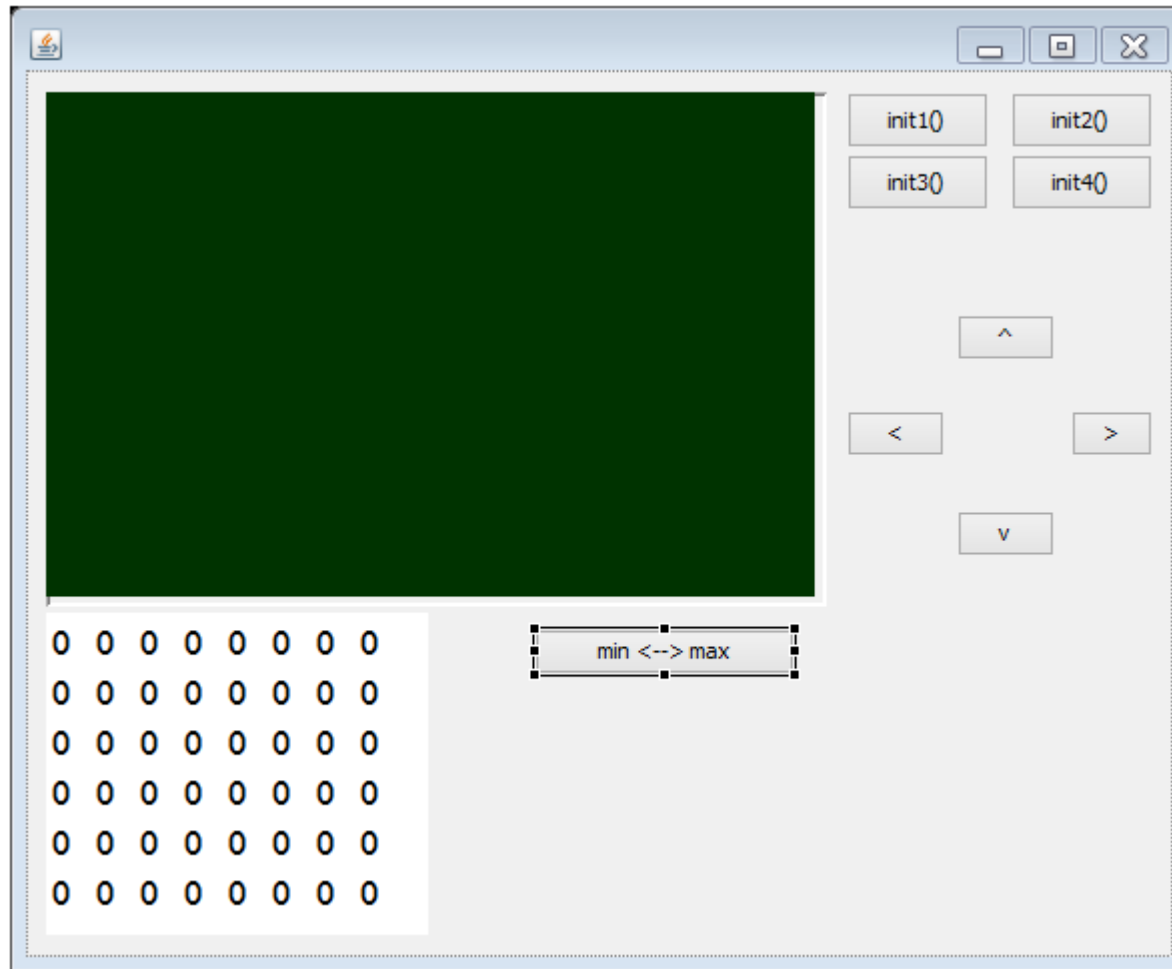
```
        super.paint(g);
```

```
        arr.drawArray(g, this.getWidth(), this.getHeight());
```

```
    }
```

```
}
```

# Класс главного окна приложения



## Класс главного окна приложения (2)

```
public class Lect2Win {
```

```
    private JFrame frame;
```

```
    private ArrayField arrayField;
```

```
    private JPanel panel;
```

```
    private JButton btnInit_1;
```

```
    private JButton btnInit_2;
```

```
    private JButton btnInit_3;
```

```
    private JButton buttonUp;
```

```
    private JButton buttonDown;
```

```
    private JButton buttonLeft;
```

```
    private JButton buttonRight;
```

```
    private JTextArea textArea;
```

```
    private JButton btnMinmax;
```

```
...
```

## Класс главного окна приложения (3)

...

```
private void initialize() {  
    frame = new JFrame();  
    frame.setBounds(100, 100, 589, 483);  
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
    frame.getContentPane().setLayout(null);  
  
    arrayField = new ArrayField();  
    panel = new ArrayPanel(arrayField);  
    panel.setBorder(new BevelBorder(BevelBorder.LOWERCED,  
                                     null, null, null, null));  
    panel.setBounds(10, 11, 390, 257);  
    frame.getContentPane().add(panel);  
}
```

## Класс главного окна приложения (4)

```
JButton btnInit = new JButton("init1()");  
btnInit.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent arg0) {  
  
        arrayField.init1();  
        panel.repaint();  
        textArea.setText(arrayField.toString());  
    }  
});  
btnInit.setBounds(410, 11, 71, 28);  
frame.getContentPane().add(btnInit);
```

## Класс главного окна приложения (5)

```
btnInit_1 = new JButton("init2()");  
btnInit_1.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent e) {  
        arrayField.init2();  
        panel.repaint();  
        textArea.setText(arrayField.toString());  
    }  
});  
btnInit_1.setBounds(492, 11, 71, 28);  
frame.getContentPane().add(btnInit_1);
```

## Класс главного окна приложения (6)

```
btnInit_2 = new JButton("init3()");  
btnInit_2.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent e) {  
        arrayField.init3();  
        panel.repaint();  
        textArea.setText(arrayField.toString());  
    }  
});  
btnInit_2.setBounds(410, 42, 71, 28);  
frame.getContentPane().add(btnInit_2);
```

## Класс главного окна приложения (7)

```
btnInit_3 = new JButton("init4()");
btnInit_3.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        arrayField.init4();
        panel.repaint();

        textArea.setText(arrayField.toString());

    }
});
btnInit_3.setBounds(492, 42, 71, 28);
frame.getContentPane().add(btnInit_3);
```



## Класс главного окна приложения (8)

```
buttonUp = new JButton("^");
buttonUp.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        arrayField.up();
        panel.repaint();

        textArea.setText(arrayField.toString());
    }
});
buttonUp.setBounds(465, 122, 49, 23);
frame.getContentPane().add(buttonUp);
```

## Класс главного окна приложения (9)

```
buttonDown = new JButton("v");
buttonDown.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        arrayField.down();
        panel.repaint();

        textArea.setText(arrayField.toString());
    }
});
buttonDown.setBounds(465, 220, 49, 23);
frame.getContentPane().add(buttonDown);
```

## Класс главного окна приложения (10)

```
buttonLeft = new JButton("<");
buttonLeft.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        arrayField.toLeft();
        panel.repaint();

        textArea.setText(arrayField.toString());
    }
});
buttonLeft.setBounds(410, 170, 49, 23);
frame.getContentPane().add(buttonLeft);
```

## Класс главного окна приложения (11)

```
buttonRight = new JButton(">");
buttonRight.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        arrayField.toRight();
        panel.repaint();
        textArea.setText(arrayField.toString());
    }
});
buttonRight.setBounds(522, 170, 41, 23);
frame.getContentPane().add(buttonRight);
```

## Класс главного окна приложения (12)

```
textArea = new JTextArea();  
textArea.setFont(new Font("Monospaced", Font.BOLD, 18));  
textArea.setBounds(10, 271, 191, 161);  
frame.getContentPane().add(textArea);  
textArea.setText(arrayField.toString());
```

```
btnMinmax = new JButton("min <--> max");  
btnMinmax.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent e) {  
  
        arrayField.replaceMinMaxColumns();  
        panel.repaint();  
        textArea.setText(arrayField.toString());
```

```
    }
```

```
});  
btnMinmax.setBounds(254, 279, 130, 23);  
frame.getContentPane().add(btnMinmax);
```

```
}
```

# Класс массива

```
import java.awt.Color;
```

```
import java.awt.Graphics;
```

```
public class ArrayField {
```

```
    private int [][]array;
```

```
    private int n;
```

```
    private int m;
```

```
    public ArrayField() {
```

```
        n = 6;
```

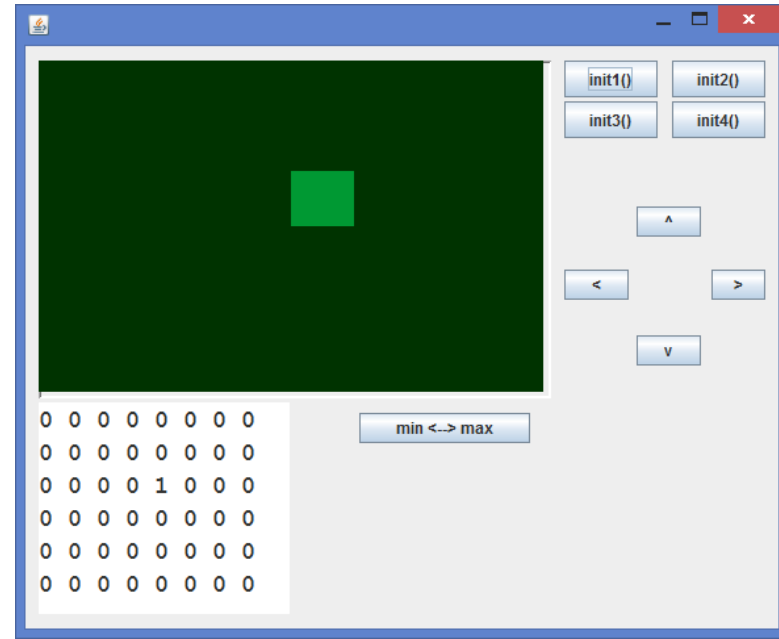
```
        m = 8;
```

```
        array = new int[n][m];
```

```
    }
```

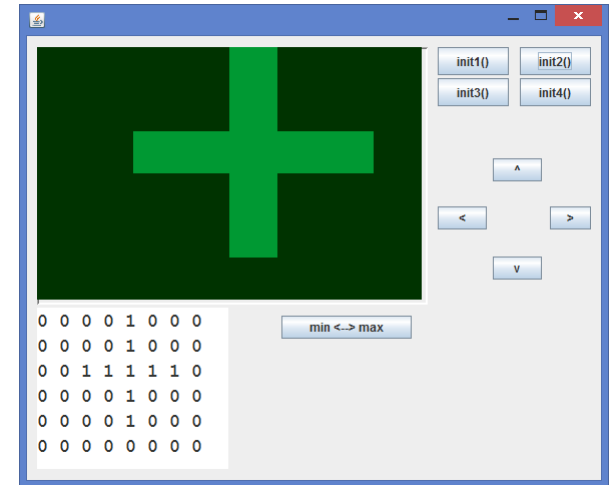
## Класс массива (2)

```
public void init1() {  
    array = new int[][] {  
        {0, 0, 0, 0, 0, 0, 0, 0},  
        {0, 0, 0, 0, 0, 0, 0, 0},  
        {0, 0, 0, 0, 1, 0, 0, 0},  
        {0, 0, 0, 0, 0, 0, 0, 0},  
        {0, 0, 0, 0, 0, 0, 0, 0},  
        {0, 0, 0, 0, 0, 0, 0, 0}  
    };  
}
```

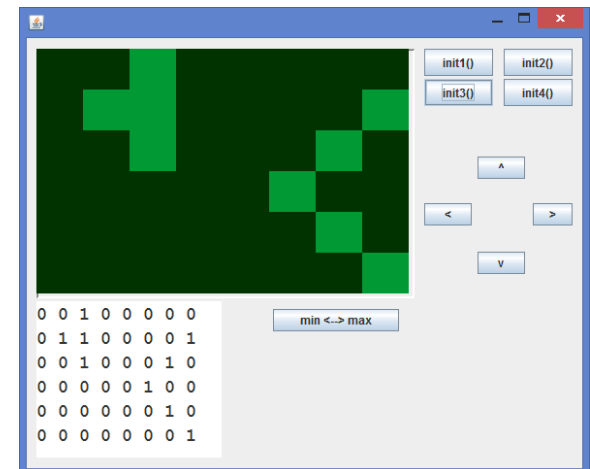


# Класс массива (3)

```
public void init2() {  
    array = new int[][] {  
        {0, 0, 0, 0, 1, 0, 0, 0},  
        {0, 0, 0, 0, 1, 0, 0, 0},  
        {0, 0, 1, 1, 1, 1, 1, 0},  
        {0, 0, 0, 0, 1, 0, 0, 0},  
        {0, 0, 0, 0, 1, 0, 0, 0},  
        {0, 0, 0, 0, 0, 0, 0, 0}  
    };  
}
```



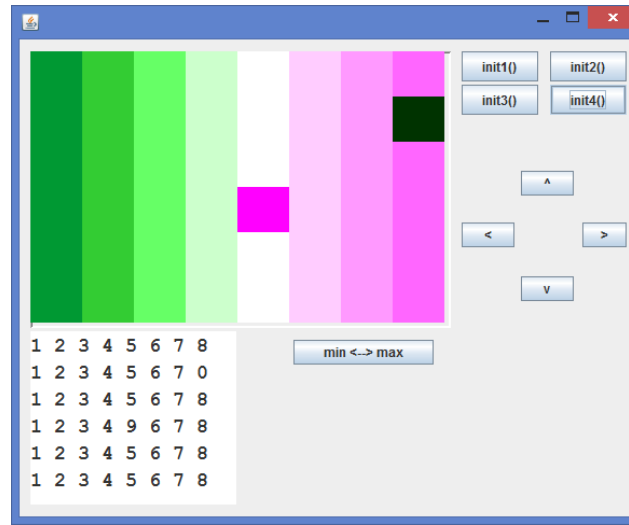
```
public void init3() {  
    array = new int[][] {  
        {0, 0, 1, 0, 0, 0, 0, 0},  
        {0, 1, 1, 0, 0, 0, 0, 1},  
        {0, 0, 1, 0, 0, 0, 1, 0},  
        {0, 0, 0, 0, 0, 1, 0, 0},  
        {0, 0, 0, 0, 0, 0, 1, 0},  
        {0, 0, 0, 0, 0, 0, 0, 1}  
    };  
}
```





# Класс массива (4)

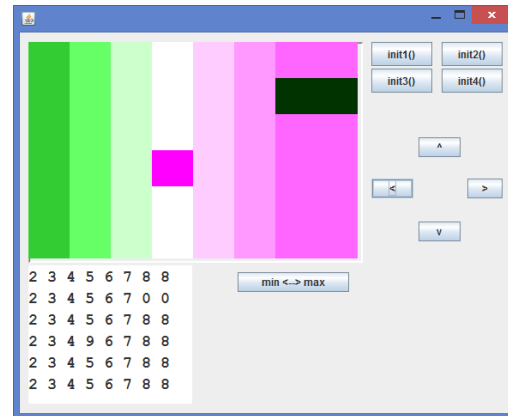
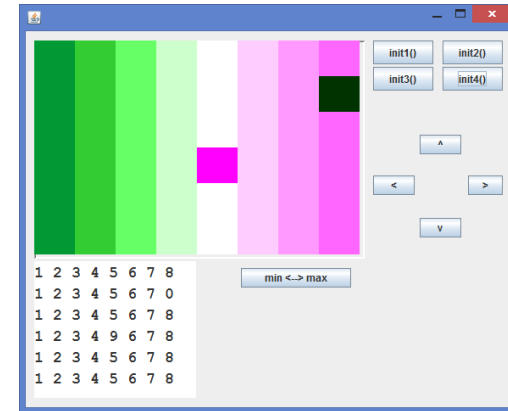
```
public void init4() {  
    array = new int[][] {  
        {1, 2, 3, 4, 5, 6, 7, 8},  
        {1, 2, 3, 4, 5, 6, 7, 0},  
        {1, 2, 3, 4, 5, 6, 7, 8},  
        {1, 2, 3, 4, 9, 6, 7, 8},  
        {1, 2, 3, 4, 5, 6, 7, 8},  
        {1, 2, 3, 4, 5, 6, 7, 8}  
    };  
}
```



# Класс массива (5)

```
public void toLeft() {  
    for (int i = 0; i < n; i++) {  
        for (int j = 1; j < m; j++) {  
            array[i][j - 1] = array[i][j];  
        }  
    }  
}
```

```
public void toRight() {  
    for (int i = 0; i < n; i++) {  
        for (int j = m - 1; j > 0; j--) {  
            array[i][j] = array[i][j - 1];  
        }  
    }  
}
```



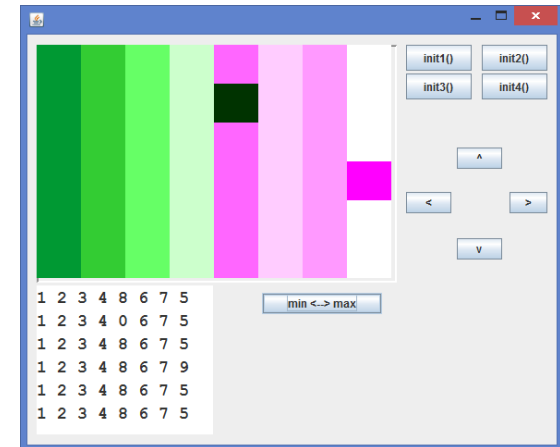
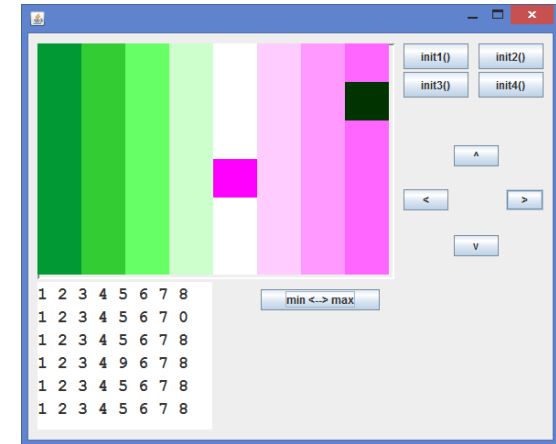
## Класс массива (6)

```
public void down() {  
    for (int i = n - 1; i > 0; i--) {  
        for (int j = 0; j < m; j++) {  
            array[i][j] = array[i - 1][j];  
        }  
    }  
}
```

```
public void up() {  
    for (int i = 0; i < n - 1; i++) {  
        for (int j = 0; j < m; j++) {  
            array[i][j] = array[i + 1][j];  
        }  
    }  
}
```

# Класс массива (7)

```
public void replaceMinMaxColumns () {  
    int maxJ = 0;  
    int minJ = 0;  
    int min = array[0][0];  
    int max = array[0][0];  
    for (int i = 0; i < n; ++i) {  
        for (int j = 0; j < m; ++j) {  
            if (array[i][j] > max) {  
                max = array[i][j];  
                maxJ = j;  
            }  
            if (array[i][j] < min) {  
                min = array[i][j];  
                minJ = j;  
            }  
        }  
    }  
    for (int i = 0; i < n; ++i) {  
        int tmp = array[i][minJ];  
        array[i][minJ] = array[i][maxJ];  
        array[i][maxJ] = tmp;  
    }  
}
```



## Класс массива (8)

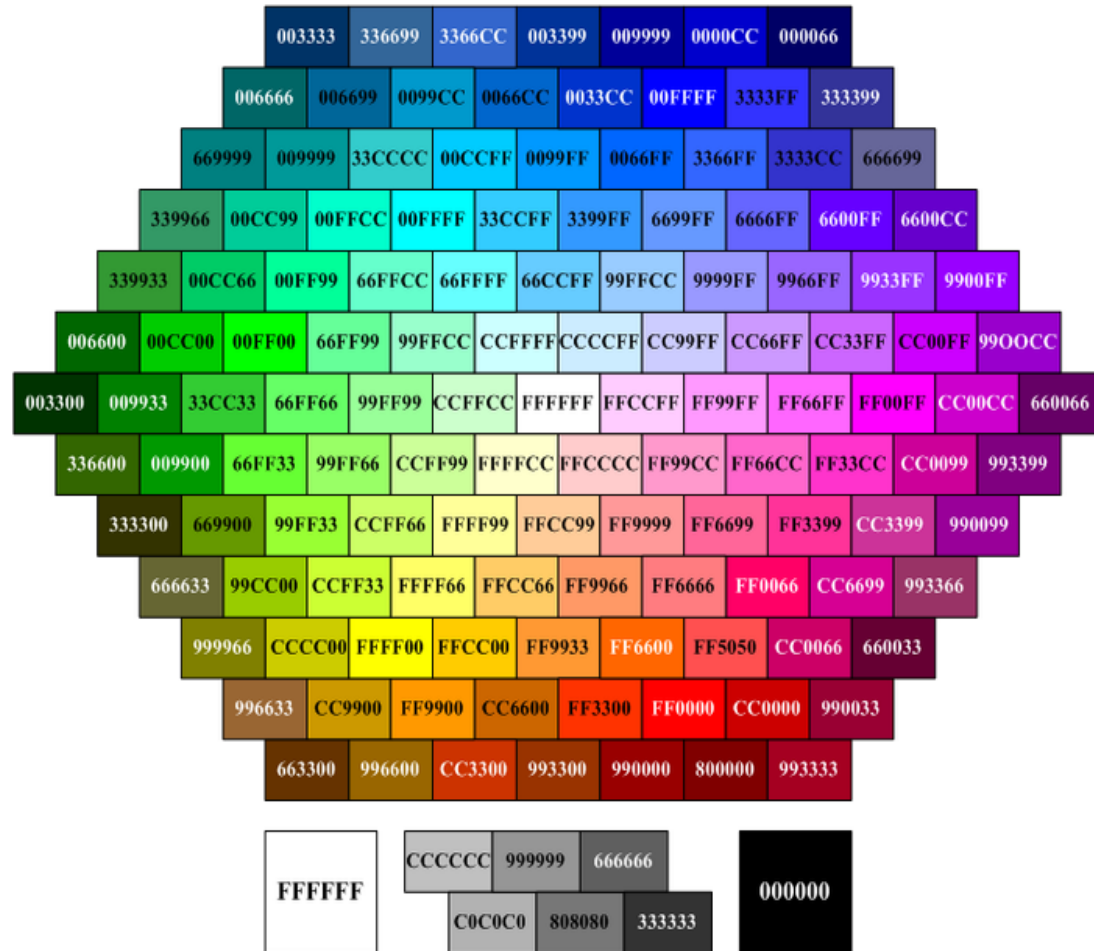
```
public String toString() {  
  
    String str = "";  
  
    for (int i = 0; i < n; i++) {  
  
        for (int j = 0; j < m ; j++) {  
            str = str + array[i][j] + " ";  
        }  
        str = str + "\n";  
    }  
  
    return str;  
}
```

## Класс массива (9)

```
final public static int CELL_HEIGHT = 24;
final public static int CELL_WIDTH = 24;
public void drawArray(Graphics g, int width, int height) {
    int cellHeight = height / n;
    int cellWidth = width / m;
    for (int i = 0; i < n; i++) {
        int top = i * cellHeight;
        for (int j = 0; j < m; j++) {
            int left = j * cellWidth;
            Color []colors = { new Color(0, 0x33, 0),
                               ... };
            g.setColor(colors[array[i][j]]);
            g.fillRect(left, top, cellWidth, cellHeight);
        }
    }
}
```

# Цветовая палитра

```
Color []colors = {
    new Color(0, 0x33, 0),
    new Color(0, 0x99, 0x33),
    new Color(0x33, 0xCC, 0x33),
    new Color(0x66, 0xFF, 0x66),
    new Color(0xCC, 0xFF, 0xCC),
    new Color(0xFF, 0xFF, 0xFF),
    new Color(0xFF, 0xCC, 0xFF),
    new Color(0xFF, 0x99, 0xFF),
    new Color(0xFF, 0x66, 0xFF),
    new Color(0xFF, 0x00, 0xFF),
    new Color(0xCC, 0x00, 0xCC),
    new Color(0x66, 0x00, 0x66)
};
```



«Палитра цветов с кодировкой» - <http://www.liveinternet.ru/users/4462189/rubric/2263007/>

# Домашнее задание

1. Делайте лабы
2. Делайте курсовую работу