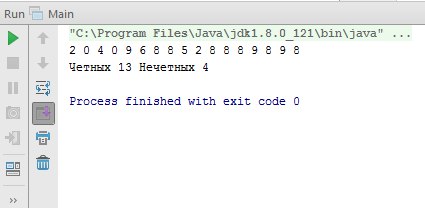
Задание №1:

Создайте массив из 17 случайных целых чисел из отрезка [0;9]. Выведите массив на экран. Подсчитайте сколько в массиве чётных и нечётных элементов и выведете количество четных и нечетных чисел на экран на отдельной строке.

Исходный код:

package com.company;   
  
import java.util.Scanner;   
import java.util.Random;   
  
public class Main {   
  
public static void main(String[] args) {   
int n =0;   
int k=0;   
Scanner sc = new Scanner([System.in](http://system.in/));   
  
int arr[] = new int[17];   
for(int i = 0; i < arr.length; i++ ){   
arr[i] = (int)((Math.random()\*10));   
}   
  
for(int i = 0; i < arr.length; i++ ){   
System.out.print(arr[i] + " ");   
}   
System.out.println();   
for(int i = 0; i < arr.length; i++ ) {   
if (arr[i] % 2 ==0 )   
{   
n = n+1;   
}   
else   
{   
k=k+1;   
}   
  
  
}   
System.out.println("Четных " + n + " Нечетных " + k);   
}   
}

Вывод:



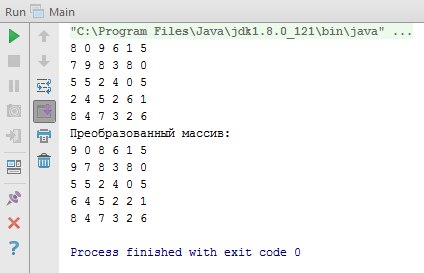
Задание №2:

Создать двумерный массив из 5 строк и 6 столбцов из случайных целых чисел из отрезка [0;9]. Вывести массив на экран. Преобразовать массив таким образом, чтобы на первом месте в каждой строке стоял её наибольший элемент. При этом изменять состав массива нельзя, а можно только переставлять элементы в рамках одной строки. Порядок остальных элементов строки не важен (т.е. можно совершить только одну перестановку, а можно отсортировать по убыванию каждую строку). Вывести преобразованный массив на экран.

Исходный код:

package com.company;   
  
public class Main{   
public static void main(String[] args){   
int [][] arr = new int[5][6];   
int i,j;   
int max = 0;   
int k,m;   
for( i =0; i<arr.length; i++){   
  
for( j=0; j<arr[i].length; j++){   
arr[i][j]=(int)(Math.random()\*10);   
  
System.out.print(arr[i][j]+" ");   
}   
System.out.println(" ");   
}   
  
for(i = 0; i < arr.length; i++)   
{   
max = a[i][0];   
m = 0;   
for(j = 0; j < arr.length; j++)   
{   
if(arr[i][j] > max) {   
max = arr[i][j];   
m=j;   
}   
}   
k=arr[i][0];   
arr[i][0] = max;   
arr[i][m] = k;   
  
}   
System.out.println("Преобразованный массив: ");   
for( i =0;i<arr.length;i++){   
  
for( j=0;j<arr[i].length;j++){   
System.out.print(arr[i][j]+" ");   
}   
System.out.println(" ");   
}   
}   
}

Вывод:



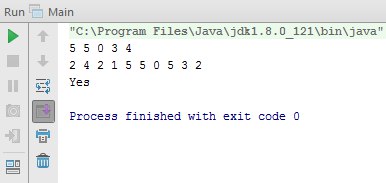
Задание №3:

Даны две последовательности целых чисел x[1]…x[n] и y[1]…y[k]. Выяснить, является ли вторая последовательность подпоследовательностью первой, т.е. при удалении некоторых членов последовательности из первой, получилась бы вторая последовательность. Число действий порядка n+k.

Исходный код:

package com.company;   
import java.util.Scanner;   
import java.util.Random;   
  
public class Main {   
public static void main(String[] args) {   
int a[],b[];   
int flag;   
int num = 0;   
a = new int[10];   
b = new int[5];   
for(int i = 0; i<5; i++){   
b[i] = (int)(Math.random()\*6);   
}   
for(int i = 0; i<10; i++){   
a[i] = (int)(Math.random()\*6);   
}   
for(int i = 0; i<5; i++){   
System.out.print(b[i]+ " ");   
}   
System.out.println();   
for(int i = 0; i<10; i++){   
System.out.print(a[i]+ " ");   
}   
for(int i = 0; i<5; i++){   
flag = 0;   
for(int j = 0; j<10; j++){   
if(flag == 0){   
if(a[j] == b[i]){   
flag = 1;   
num = num +1;   
}   
}   
}   
}   
System.out.println();   
if(num >= 5){   
System.out.println("Yes");   
}   
else{   
System.out.println("No");   
}   
}   
}

Вывод:



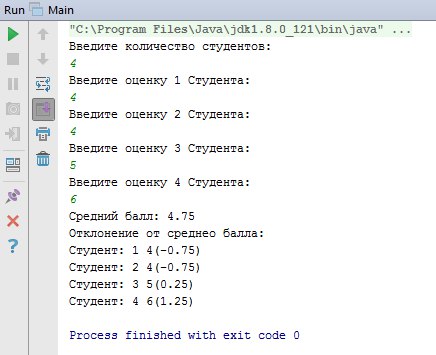
Задание №4:

Написать программу, которая позволяет пользователю вычислить среднее значение оценки группы и отклонение от среднего значения оценки после прохождения экзамена.

Исходный код:

package com.company;   
import java.util.Scanner;   
import java.util.Random;   
public class Main {   
  
  
public static void main(String[] args) {   
int k;   
double sr,s;   
Scanner sc = new Scanner([System.in](http://system.in/));   
System.out.println("Введите количество студентов: ");   
k = sc.nextInt();   
s = 0;   
int a[] = new int[k];   
  
for(int i = 0; i < a.length; i++ )   
{   
System.out.println("Введите оценку " + (i+1) + " Студента: ");   
a[i] = sc.nextInt();   
s=s+a[i];   
}   
sr = s/k;   
System.out.println("Средний балл: " + sr);   
System.out.println("Отклонение от среднео балла: ");   
for (int i =0; i<a.length; i++)   
{   
System.out.println("Студент: " + (i+1) + " " +a[i] + "(" + (a[i]-sr) + ")");   
}   
  
}   
}

Вывод:



Задание №5:

В целочисленном массиве a[1]..a[n] хранится перестановка чисел 1..n (каждое из чисел встречается по одному разу).

1) Определить четность перестановки (количество действий порядка n).

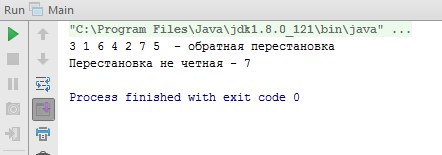
2) Не используя других массивов, заменить перестановку на обратную.

(Четность перестановки определяется количеством циклов.)

Исходный код:

import java.util.ArrayList;   
  
  
public class laba3\_1 {   
  
public static void main(String[] args) {   
int[] p = {2,5,1,4,7,3,6};   
int n=p.length;   
int[] op = new int[n];   
  
for (int i = 0; i < n; i++) {   
for (int j = 0; j < n; j++) {   
if (p[j] == i + 1) {   
op[i] = j + 1;   
}   
}   
}   
  
for (int i=0;i<n;i++){   
System.out.print(op[i]+ " ");   
}   
System.out.print(" - обратная перестановка");   
  
int sum = 0;   
int iter = 0;   
  
ArrayList<Integer> buff = new ArrayList<Integer>();   
  
for (int el:p){   
buff.add(el);   
}   
for(int i=1;i<=n;i++){   
iter = 0;   
for (int j=0;j<buff.size();j++){   
if (buff.get(j) != i ){   
iter += 1;   
} else{   
sum = sum + iter;   
buff.remove(j);   
break;   
}   
}   
  
}   
System.out.println();   
  
if (sum%2==0){   
System.out.println("Перестановка четная - "+sum);   
}   
else System.out.println("Перестановка не четная - "+sum);   
}   
}

Вывод:



Задание №6:

Сортировка массива из 10 элементов методом слияния.

Исходный код:

**import** java.util.\*;

**public** **class** MergeSort {

**public** **static** **void** main(String[] args) {

**int** a[] = **new** **int** [10];

**int** b[] = **new** **int** [10];

**for**(**int** i = 0; i < 10; i++){

a[i] =(**int**)(Math.*random*()\*10);

}

**for**(**int** i = 0; i < 10; i++){

System.*out*.print(a[i]+ " ");

}

System.*out*.print('\n');

b = *sortMerge*(a);

**for**(**int** i = 0; i < 10; i++){

System.*out*.print(b[i]+ " ");

}

}

**private** **static** **int**[] sortMerge(**int**[] arr) {

**int** len = arr.length;

**if** (len < 2) **return** arr;

**int** middle = len / 2;

**return** *merge*(*sortMerge*(Arrays.*copyOfRange*(arr, 0, middle)),

*sortMerge*(Arrays.*copyOfRange*(arr, middle, len)));

}

**private** **static** **int**[] merge(**int**[] arr\_1, **int**[] arr\_2) {

**int** len\_1 = arr\_1.length, len\_2 = arr\_2.length;

**int** a = 0, b = 0, len = len\_1 + len\_2;

**int**[] result = **new** **int**[len];

**for** (**int** i = 0; i < len; i++) {

**if** (b < len\_2 && a < len\_1) {

**if** (arr\_1[a] > arr\_2[b]) result[i] = arr\_2[b++];

**else** result[i] = arr\_1[a++];

} **else** **if** (b < len\_2) {

result[i] = arr\_2[b++];

} **else** {

result[i] = arr\_1[a++];

}

}

**return** result;

}

}

Вывод:

