# Homework reflections

Airat Khasianov - Class 12

# Total Recall

```
int max = a[0]+a[1]+a[2] ;
for(i=3;i<n;i++) {
    sum1 = a[i] + a[i-1] + a[i-2];
    sum2 = a[i-1]+ a[i-2] + a[i-3];
    if (sum1>sum2){
        max = sum1;
    } else {
        max = sum2;
    }
}
```

Does this calculate the maximum sum of the three consecutive elements?

# Flashback

```
public static boolean chek(int i){
        while (i>0){
            if ((i%10==1)|(i%10==0)) i/=10;
            else return false;
        }
    return true;
}
```

Improve the method!

# Flashback

```
public static boolean chek(int i){
        boolean t;
        do {
            t = (i%10==1)||(i%10==0));
            i/=10;
        } while (i>0 && t)
    return t;
}
```

# Flashback 2

1. Test the array if every five elements of the array represent a binary number

Correct?

```
boolean bool = false;
int k=0;
for (int i = 0; i < n; i++) {
    int x = new Scanner(System.in).nextInt();
    if (x == 1 || x == 0) {
        bool = true;
        k++;
    } else
        k = 0;
}
if ((bool)&&(k==5))
    System.out.println("Есть такие числа");
else
    System.out.println("Нет");
}
```

# Bonus 3

3. Вводится матрица a(m,n) из 0 и 1. Найти в ней квадратную подматрицу из одних единиц максимального размера.

```
static Scanner sc = new Scanner(System.in);
public static void main(String[] args) {
    int n = sc.nextInt();
    int m = sc.nextInt();
    int[][] a = new int [n][m];
    int maxSize = 0;
    readMatrix(a, n, m);

    for (int i = n-2; i>=0; i-- )
        for (int j = m-2; j>=0; j-- ){
            if (a[i][j]!=0){
                a[i][j] = Math.min(Math.min(a[i][j+1], a[i+1][j+1]),a[i+1][j])+1;
                if (a[i][j] > maxSize){
                    maxSize = a[i][j];
                }
            }
        }
     System.out.println("Maximal submatrix size = "+maxSize);
}
```

# Angular Matrix

```
for (int j = 0; j <= n - 2; j++) {
    for (int i = j + 1; i <= n - 1; i++) {
            matrix[i][j] = matrix[i][j] - matrix[j][i] * matrix[i][j] / matrix[j][i];
        }
    }
```

# Selfmade, Task 46

```java
public static int compareStrings(String str1, String str2) {
    int minLength = (str1.length() < str2.length() ? str1.length() : str2.length());
    for (int i = 0; i < minLength; i++) {
        if (str1.charAt(i) < str2.charAt(i)) {
            return 1;
        } else if (str2.charAt(i) < str1.charAt(i)) {
            return 2;
        }
    }
    if (str1.length() < str2.length()) {
        return 1;
    } else if (str1.length() > str2.length()){
        return 2;
    }
    return 0;
}
```

# Java made, Task 46

Does this code do the job?

```
Scanner sc = new Scanner(System.in);
String s1 = sc.nextLine(), s2 = sc.nextLine();
if (s1.compareToIgnoreCase(s2) == 1) {
    System.out.println(s2);
}
else {
    if (s1.compareTo(s2) == -1) {
    System.out.println(s1);
    }
    else {
        System.out.println("These strings are equal");
    }
}
```

# Java made, Task 46

```
int switchKey = s1.compareToIgnoreCase(s2);
switchKey = switchKey > 0 ? 1 : switchKey < 0 ? -1 : 0;
switch (switchKey) {
    case 1 : resultString = s2; break;
    case -1 : resultString = s1; break;
    default : resultString = "These strings are equal";
}
System.out.println(resultString);
```

# Very primitive, Task 47

```
class Task47Demo{
    private
    static void doTask008(){…}
    static void doTask024(){…}
    static void doTask033(){…}
public static void main(String[] args) {
    doTask008();
    doTask024();
    doTask033();
}
```

# Или так... Task 47

```java
import java.util.Scanner;
public class Task047_024 {
    private static final double EPS = 0.000000001;

    public static void main(String[] args) {
        double x = new Scanner(System.in).nextDouble();
        System.out.println(getResult(x, EPS));
    }

    private static double getResult(double x, double e) {
        double result = 0;
        int n = 1;
        double temp;
        do {
            temp = getSummand(x, n);
            result += temp;
            n++;
        } while (temp > EPS);
        return result;
    }
    private static double getSummand(double x, int n) {
        return 1 / (n * pow(9, n) * pow(x - 1, 2 * n));
    }
    private static double pow(double a, int n) {
        if (n == 0) return 1;
        return a * pow(a, n - 1);
    }
}
```

# Class Student

What's wrong?

```
package ru.kfu.itis;
public class Student {
    private String full_name;
    Student(){
        full_name = "";
    }
    public void setFull_name(String full_name) {
        this.full_name = full_name;
    }
    public String getFull_name() {
        return full_name;
    }
}
```

# Class Student

```java
package ru.kfu.itis;
public class Student {
    private String full_name;
    Student(String name){
        full_name = name;
    }
    public void setFull_name(String full_name) {
        this.full_name = full_name;
    }
    public String getFull_name() {
        return full_name;
    }
}
```

# How teacher.evaluate()

subject, name?

```
public String evaluateStudent (Student student) {
        Random random = new Random();
        int mark = random.nextInt(4) + 2;
        String strMark = "";
        switch (mark) {
            case 2:
                strMark = "неудовлетворительно";
                break;
            case 3:
                strMark = "удовлетворительно";
                break;
            case 4:
                strMark = "хорошо";
                break;
            case 5:
                strMark = "отлично";
                break;
        }
        System.out.printf("преподаватель %s оценил студента с именем %s по предмету %s на оценку %s.",
                name, student.getName(), subject, strMark);
    }
```

# Easy Vector2D

```
public class Vector2D {
    private double[] vector;
    public Vector2D() {
        vector = new double[2];
        vector[0] = 0;
        vector[1] = 0;
    }

    public Vector2D(double x, double y) {
        vector = new double[2];
        vector[0] = x;
        vector[1] = y;
    }

. . .

}
```

# RationalFraction

```
public class RationalFraction {
    private int numerator;
    private int denominator;
    public RationalFraction() {
        numerator = 0;
        denominator = 1;
    }
    public RationalFraction(int numerator, int denominator) {
        this.numerator = numerator;
        this.denominator = denominator;
    }
    public void setNumerator(int numerator) {
        this.numerator = numerator;
    }
    public void setDenominator(int denominator) {
        this.denominator = denominator;
    }
    public int getNumerator() {
        return numerator;
    }
    public int getDenominator() {
        return denominator;
    }
    public void reduce() {
        int a = Math.abs(numerator);
        int b = Math.abs(denominator);
        while (a != 0 && b != 0) {
            if (a > b) a %= b;
            else b %= a;
        }
        int gcd = a + b;
        numerator /= gcd;
        denominator /= gcd;
    }
. . .
}
```

# Any Difficulties…

- With Matrix2x2?
- With Complex numbers?

# Numbers are Complex

```java
public class ComplexNumber {
    private double real;
    private double imaginary;
    public ComplexNumber(double real, double imaginary) {
        this.real = real;
        this.imaginary = imaginary;
    }
    public ComplexNumber() {
         return new ComplexNumber(0,0);
    }
    public ComplexNumber add(ComplexNumber number) {}
    public ComplexNumber sub(ComplexNumber number) {}
    public ComplexNumber multNumber(double k) {}
    public void add2(ComplexNumber number) {}
    public void sub2(ComplexNumber number) {}
    public void multNumber2(double k) {}
    public ComplexNumber mult(ComplexNumber number) {}
    public void mult2(ComplexNumber number) {}
    public ComplexNumber div(ComplexNumber number) {}
    public void div2(ComplexNumber number) {}
    public double length() {}

@Override
    public String toString() {
        String result = "" + imaginary + " * i ";
        if (real < 0) result += "- " + Math.abs(real);
        else result += "+ " + real;
        return result;
    }
    public double arg() {}
    public ComplexNumber pow(double n) {}
    public boolean equals(ComplexNumber number) {}
}
```

# the matrix

```
File file = new File("input.txt");
Scanner sc = new Scanner(file);
String line = sc.nextLine();
String[] words = line.split(" ");
int k = words.length;//количество столбцов в кв. матрице
int[][] matrix = new int[k][k];
int[][] matrix2 = new int[k][k];
sc = new Scanner(file);

for (int y = 0; y < k; y++) {
    for (int x = 0; x < k; x++) {
        matrix[y][x] = sc.nextInt();
    }
}

for (int y = 0; y < k; y++) {
    for (int x = 0; x < k; x++) {
        matrix2[y][x] = matrix[x][y];
    }
}
```

# Transpose

ComplexVector2D

# Is everything fine?

```
private ComplexNumber numberX;
private ComplexNumber numberY;
ComplexVector2D() {
    numberX = new ComplexNumber();
    numberY = new ComplexNumber();
}
ComplexVector2D(ComplexNumber x, ComplexNumber y) {
    numberX = new ComplexNumber();
    numberX = x;
    numberY = new ComplexNumber();
    numberY = y;
}
ComplexVector2D add(ComplexVector2D vector2) {
    ComplexVector2D res = new ComplexVector2D();
    res.numberX = numberX.add(vector2.numberX);
    res.numberY = numberY.add(vector2.numberY);
    return res;
}
```

# How does your portfolio progresses?

# Home Work

- 55 + Weekly Project + Your Debts
- Please, only latin text in the output!