

## 3222: Προγραμματισμός Υπολογιστών με Java (Μ-Ω)

### Άσκηση

19/4/2021

e-mail: [mst@aueb.gr](mailto:mst@aueb.gr)

**Αντικείμενο εργασίας:** Κληρονομικότητα, Συναρτήσεις Δημιουργίας και Υποκλάσεις, Υπερκάλυψη και Απόκρυψη Μεθόδων, Πολυμορφισμός και Ομοιόμορφη Κλήση Μεθόδων, Αφηρημένες Κλάσεις, Συλλογές (Πίνακες) Αντικειμένων, Προβιβασμός (upcasting) και υποβιβασμός (downcasting).

Απαντήστε σε όλες τις ερωτήσεις, πάνω το παρόν έντυπο (doc file), με κόκκινο χρώμα χαρακτήρων.

Επίσης διορθώστε-συμπληρώστε τα αντίστοιχα αρχεία (java files) που επισυνάπτονται, έτσι ώστε να μεταγλωττίζονται και να εκτελούνται σωστά, με βάση τις οδηγίες που δίνονται για κάθε επιμέρους άσκηση.

#### **Παραδοτέα:**

Συμπιέστε σε ένα αρχείο, το doc file και όλα τα java files, με όνομα τον αριθμό φοιτητικού μητρώου σας.

**Ημερομηνία παράδοσης:** Παρασκευή, 7/5/2021 (e-class)

Καλή επιτυχία!

## Άσκηση 2.1 (5 μονάδες)

Διορθώστε το πρόγραμμα **ex21** έτσι ώστε να μεταγλωττίζεται και να εκτελείται σωστά:

```
class Rectangle {  
    int width;  
    int height;  
    int area(){  
        return width * height;  
    }  
}  
  
public class ex21 {  
    public static void main(String[] args) {  
        Rectangle myRect = new Rectangle();  
        myRect.width = 4;  
        myRect.height = 5;  
        System.out.println("myRect's area is " + myRect.area());  
    }  
}
```

## Άσκηση 2.2 (5 μονάδες)

Δίνεται το πρόγραμμα **ex22**:

```
class ClassA {
    public void methodOne() {
        System.out.println ("Class A methodOne");
    }
    public static void methodTwo() {
        System.out.println ("Class A methodTwo");
    }
    public void methodThree() {
        System.out.println ("Class A methodThree");
    }
}

class ClassB extends ClassA {
    public void methodOne() {
        System.out.println ("Class B methodOne");
    }
    public static void methodTwo() {
        System.out.println ("Class B methodTwo");
    }
}

public class ex22{
    public static void main(String args[]){
        ClassB b = new ClassB();
        b.methodOne();
        b.methodTwo();
        b.methodThree();
    }
}
```

1. Ποια μέθοδος της υποκλάσης υπερκαλύπτει μέθοδο της υπερκλάσης (Method overriding);

**Απάντηση:**

Η μέθοδος **methodOne()** της υποκλάσης (ClassB) υπερκαλύπτει την αντίστοιχη μέθοδο της υπερκλάσης (ClassA).

2. Ποια μέθοδος της υποκλάσης κάνει απόκρυψη σε μέθοδο της υπερκλάσης (Method hiding);

**Απάντηση:**

Η μέθοδος **methodTwo()** της ClassB αποκρύπτει την αντίστοιχη μέθοδο της ClassA.

3. Η εντολή b.methodThree() προκαλεί σφάλμα διότι η classB δεν έχει μέθοδο methodThree(), σωστό ή λάθος; Αιτιολογείστε:

**Απάντηση:**

**Λάθος.** Η κλάση ClassB κληρονομεί τη μέθοδο **methodThree()** από την υπερκλάση της (ClassA).

## Άσκηση 2.3 (10 μονάδες)

Δίνεται το πρόγραμμα Q2:

```
class Test {
    static void show() {
        System.out.println("Show method in Test class");
    }
}

public class Q2 extends Test {

    static void show() {
        System.out.println("Show method in Q2 class");
    }

    public static void main(String[] args) {
        Test t = new Test();
        t.show();
        Q2 q = new Q2();
        q.show();
        t = q;
        t.show();
        q = t;
        q.show();
    }
}
```

Στο κύριο πρόγραμμα:

1. Διορθώστε την εντολή **q = t** έτσι ώστε να μη δημιουργεί σφάλμα κατά τη μεταγλώττιση.

### Απάντηση:

Η εντολή γίνεται: **q = (Q2) t;**     *-(type casting)*

2. Η 2<sup>η</sup> εντολή **t.show()** του main, καλεί τη μέθοδο show() της Test και όχι της Q2, παρόλο που αμέσως πριν έχει γίνει t = q. Είναι δυνατό, χωρίς να αλλάξουμε το κύριο πρόγραμμα, η εντολή t.show() να καλεί τη show() της Q2;

Ναι ή Όχι. Αν ναι, με ποιο τρόπο;

### Απάντηση:



**Ναι.** Κάνοντας τις μεθόδους **show()** μη-static η εντολή `t.show()` θα καλεί τη μέθοδο της `Test`.

Προηγουμένως, όντας οι μέθοδοι `static`, γινόταν **hiding** (απόκρυψη). Κάνοντας τις μεθόδους `μη-static` γίνεται **overriding** (υπερκάλυψη). Εκμεταλλευόμαστε τη **διαφορά** των δυο αυτών εννοιών/λειτουργιών, δηλ. το γεγονός ότι όταν γίνεται υπερκάλυψη καλείται η μέθοδος που αντιστοιχεί στον τύπο του αντικειμένου (`q`) και όχι της αναφοράς (`t`).

## Άσκηση 2.4 (15 μονάδες)

Δίνεται το πρόγραμμα **ex24**:

```
class Point {
    private int x = 0;
    private int y = 0;
    public Point(int x, int y) {
        this.x = x;
        this.y = y;
    }
}

class Rectangle {
    private int width = 0;
    private int height = 0;
    private Point origin;
    public Rectangle() {
        origin = new Point(0,0);
    }
    public Rectangle(Point p) {
        origin = p;
    }
    public Rectangle(Point p, int w, int h) {
        origin = p;
        width = w;
        height = h;
    }
    public Rectangle(int w, int h) {
        this(new Point(0,0), w, h);
    }
    // a method for computing the area of the rectangle
    public int area() {
        return width * height;
    }
}

public class ex24 {

    public static void main (String args[]){
        Rectangle r1 = new Rectangle();
        Rectangle r2 = new Rectangle(new Point(2,3));
        Rectangle r3 = new Rectangle(new Point(2,3),3,7);
        Rectangle r4 = new Rectangle(4,9);
    }
}
```

Ένα ορθογώνιο παραλληλόγραμμο μπορεί να οριστεί από ένα σημείο (Point), το ύψος (height) και το πλάτος του (width).

Συμπληρώστε τις εντολές (κενά) στους **κατασκευαστές** της κλάσης **Rectangle**, χωρίς να προσθέσετε νέες εντολές, ως εξής:

- **Rectangle():** ορίζει μόνο ένα σημείο με συντεταγμένες (0, 0) ως ένα αντικείμενο της Point.
- **Rectangle(Point p):** ορίζει μόνο ένα σημείο (αυτό που δίνεται ως παράμετρος).
- **Rectangle(Point p, int w, int h):** ορίζει ένα σημείο, το πλάτος και το ύψος του παραλληλογράμμου (όλα δίνονται ως παράμετροι).
- **Rectangle(int w, int h):** ορίζει ένα σημείο με συντεταγμένες (0, 0) ως ένα αντικείμενο της Point και το πλάτος και το ύψος του παραλληλογράμμου (το πλάτος και το ύψος δίνονται ως παράμετροι), με κλήση του κατασκευαστή Rectangle(Point p, int w, int h).

Επίσης, συμπληρώστε τα κενά στην **κύρια μέθοδο**, χωρίς να προσθέσετε νέες εντολές, έτσι ώστε να δημιουργεί αντικείμενα τύπου Rectangle, δίνοντας από ένα παράδειγμα κλήσης κάθε κατασκευαστή της Rectangle (ορίστε όποιες αριθμητικές τιμές θέλετε ως δεδομένα χωρίς scanner).

**Η απάντηση να δοθεί παραπάνω. Επίσης συμπληρώστε και το αντίστοιχο java file.**

## Άσκηση 2.5 (10 μονάδες)

Δίνεται το πρόγραμμα **ex25**:

Συμπληρώστε το πρόγραμμα ex25, που δημιουργεί ένα Array με όνομα **grades** και με την ακόλουθη βαθμολογία 10 μαθητών σε ένα μάθημα: 7, 9, 10, 8.5, 8, 9.5, 2, 4, 7, 8.

Επίσης υπολογίζει μέσω των αντίστοιχων μεθόδων:

α) τη **μέση τιμή** της βαθμολογίας όλων των μαθητών,

β) τη **μεγαλύτερη** βαθμολογία,

γ) τη **μικρότερη** βαθμολογία,

δ) Συμπληρώστε τη μέθοδο myCopy που να παίρνει δύο ορίσματα: το πρώτο όρισμα η αναφορά στον πίνακα που έχετε ορίσει παραπάνω και το δεύτερο όρισμα είναι η αναφορά σε νέο πίνακα με όνομα **grades\_backup**. Η μέθοδος myCopy θα **αντιγράφει** τα στοιχεία του πρώτου πίνακα στον δεύτερο πίνακα.

**Συμπληρώστε ΜΟΝΟ τα κενά χωρίς να προσθέσετε νέες εντολές.**

```

public class ex25 {

    public static void main(String args[]) {

        double[] grades = {7, 9, 10, 8.5, 8, 9.5, 2, 4, 7, 8};

        double[] grades_backup = new double[grades.length];

        System.out.println( "mean value: " + meanValue (grades) );
        System.out.println( "max value: " + maxValue (grades) );
        System.out.println( "min value: " + minValue (grades) );
        myCopy (grades, grades_backup);
        printArray (grades);
        printArray (grades_backup);
    }
    private static double meanValue(double[] array) {
        double sum = 0;
        for (double item: array)
            sum+=item;
        return sum/(array.length);
    }
    private static double maxValue(double[] array) {
        double max = array[0];
        for (double item: array)
            if(item > max) max = item;
        return max;
    }
    private static double minValue(double[] array) {
        double min = array[0];
        for (double item: array)
            if(item < min) min = item;
        return min;
    }
    private static void myCopy(double[] grades, double[] grades_backup) {
        for (int i=0; i < grades.length; i++)
            grades_backup[i] = grades[i];
    }
    private static void printArray(double[] array) {
        for (int i=0; i < array.length; i++)
            System.out.print(i==array.length-1 ?
                                array[i]+" \n" : array[i]+" ");
    }
}

```



Η απάντηση να δοθεί παραπάνω. Επίσης συμπληρώστε και το αντίστοιχο java file.

## Άσκηση 2.6 (15 μονάδες)

Δίνεται το πρόγραμμα ex26:

```
class Shape {
    void draw() {}
    void erase() {}
}

class Circle extends Shape {

    void draw() {
        System.out.println("Circle.draw()");
    }
    void erase() {
        System.out.println("Circle.erase()");
    }
}

class Square extends Shape {

    void draw() {
        System.out.println("Square.draw()");
    }
    void erase() {
        System.out.println("Square.erase()");
    }
}

class Triangle extends Shape {

    void draw() {
        System.out.println("Triangle.draw()");
    }

    void erase() {
        System.out.println("Triangle.erase()");
    }
}
```

```

public class ex26 {

    public static Shape randShape() {

        switch((int)(Math.random() * 3)) {
            default:
            case 0: return new Circle();
            case 1: return new Square();
            case 2: return new Triangle();
        }
    }

    public static void main(String[] args) {
        Shape[] shapes = new Shape[10];
        for (int i=0; i<10; i++)
            shapes[i] = randShape();
        for (Shape s: shapes)
            s.draw();
    }
}

```

Στο κύριο πρόγραμμα συμπληρώστε τα κενά (μην προσθέσετε νέες γραμμές-εντολές), ειδικότερα:

1. Ορίστε έναν πίνακα 10 θέσεων. Σε κάθε θέση του πίνακα πρέπει να μπορεί να αποθηκεύεται ένα σχήμα δηλαδή ένα αντικείμενο τύπου **Circle** ή **Square** ή **Triangle**.
2. Γεμίστε τον πίνακα με σχήματα δηλαδή αντικείμενα που έχουν δημιουργηθεί με τυχαίο τρόπο μέσω της μεθόδου **randShape()**.
3. Εμφανίστε τα σχήματα του πίνακα, δηλαδή για κάθε σχήμα του πίνακα καλέστε τη μέθοδο **draw()** (πολυμορφισμός).

**Η απάντηση να δοθεί παραπάνω. Επίσης συμπληρώστε και το αντίστοιχο java file.**

## Άσκηση 2.7 (15 μονάδες)

Δίνεται το πρόγραμμα **ex27**:

```

abstract class Animal {
    String name;
    static int animals = 0;
    public Animal(String n) {
        name = n;
        animals++;
    }
    public abstract void speak();
    public static int numberOfAnimals() {
        return animals;
    }
} // Animal

```

```

class Dog extends Animal {
    String sound = "woof";
    static int dogs = 0;

    // Fill your code here
    public Dog(String n){
        super(n); //Animal.animals increases, as well
        dogs++;
    }
    public void speak(){
        System.out.print(name + ": " + sound + "\n");
    }
    public static int numberOfDogs() {
        return dogs;
    }
}

class Cat extends Animal {
    String sound = "miaou";
    static int cats = 0;

    // Fill your code here
    public Cat(String n){
        super(n); //Animal.animals increases, as well
        cats++;
    }
    public void speak(){
        System.out.print(name + ": " + sound + "\n");
    }
    public static int numberOfCats() {
        return cats;
    }
}

public class ex27 {

    public static void main(String[] args) {
        Animal[] animal ={new Cat("stella"),new Cat("ziggy"),new Dog("azor")};
        System.out.println("cats: "+Cat.numberOfCats());
        System.out.println("dogs: "+Dog.numberOfDogs());
        System.out.println("animals: "+Animal.numberOfAnimals());
        for ( int i = 0; i < animal.length; i++ )
            animal[i].speak();
    }
}

```

Το πρόγραμμα ex27 δημιουργεί δύο αντικείμενα τύπου Cat και ένα τύπου Dog.  
Στη συνέχεια εμφανίζει το πλήθος των αντικειμένων που δημιούργησε, ως εξής:

```
cats: 2
dogs: 1
animals: 3
```

Τέλος εμφανίζει την ιδιότητα sound κάθε αντικειμένου.

```
stella: miaou
ziggy: miaou
azor: woof
```

Διορθώστε και συμπληρώστε τις κλάσεις **Animal**, **Dog** και **Cat** έτσι ώστε το πρόγραμμα να μην έχει συντακτικά σφάλματα και όταν εκτελείται να λαμβάνουμε τα παραπάνω αποτελέσματα **ΧΩΡΙΣ ΝΑ ΑΛΛΑΞΕΤΕ το κύριο πρόγραμμα.**

Η απάντηση να δοθεί παραπάνω. Επίσης συμπληρώστε και το αντίστοιχο java file.

## Άσκηση 2.8 (15 μονάδες)

Δίνεται το πρόγραμμα ex28:

```
class product {
    private String name;
    private String price;

    // Fill your code here

    public product(String name, String price){
        this.name = name;
        this.price = price;
    }

    public String getName(){
        return name;
    }
    public String getPrice(){
        return price;
    }
    public void setPrice(String newPrice){
        this.price = newPrice;
    }
}

class book extends product {
    private String author;
    private String publisher;
```

```

// Fill your code here

public book(String name, String price, String publisher, String author){
    super(name,price);
    this.publisher = publisher;
    this.author = author;
}
public String getPublisher(){
    return publisher;
}
public String getAuthor(){
    return author;
}
}

class cd extends product {
    private String artist;

    // Fill your code here

    public cd(String name, String price, String artist){
        super(name,price);
        this.artist = artist;
    }
    public String getArtist(){
        return artist;
    }
}

```

```

public class ex28 {
    public static void main (String args[]) {
        System.out.println("Welcome to our bookstore!");
        book b = new book("Effective Java", "30", "Addison Wesley",
                           "Joshua Bloch");
        System.out.println("Titlos bibliou: " + b.getName() + "\nTimi: "
                           + b.getPrice() + " \nEkdoseis:" + b.getPublisher()
                           + "\nAuthor " + b.getAuthor());

        cd c = new cd ( "the wall", "20.99", "Pink Floyd");

        System.out.println("Titlos cd : " + c.getName() + "\nTimi: "
                           + c.getPrice()+ "\nArtist: " + c.getArtist());

        c.setPrice("9,99");
        System.out.println("Titlos cd : " + c.getName()
                           + "\nTimi me ektptwsi: " +c.getPrice()
                           + "\nArtist: " + c.getArtist());
    }
}

```

Συμπληρώστε τις κλάσεις **product**, **book** και **cd** έτσι ώστε να λαμβάνουμε τα αναμενόμενα αποτελέσματα **ΧΩΡΙΣ ΝΑ ΑΛΛΑΞΕΤΕ το κύριο πρόγραμμα**.

Με βάση το main που δίνεται τα αναμενόμενα αποτελέσματα είναι:

```

Welcome to our bookstore!
Titlos bibliou: Effective Java
Timi: 30
Ekdoseis: Addison Wesley
Author Joshua Bloch
Titlos cd : the wall
Timi: 20.99
Artist: Pink Floyd
Titlos cd : the wall
Timi me ektptwsi: 9,99
Artist: Pink Floyd

```

Η απάντηση να δοθεί παραπάνω. Επίσης συμπληρώστε και το αντίστοιχο java file.

## Άσκηση 2.9 (15 μονάδες)

Σε αυτή την άσκηση θα πρέπει να συμπληρώσετε ένα πρόγραμμα που δημιουργεί αληθοφανείς αριθμούς πιστωτικών καρτών **Visa**, **MasterCard**, **American Express** και **Diners**. Το πρόγραμμα ρωτάει το χρήστη για τον τύπο της κάρτας που θέλει να δημιουργήσει και στη συνέχεια δημιουργεί έναν αριθμό πιστωτικής κάρτας.

Οι αριθμοί πιστωτικών καρτών έχουν διαφορές και ομοιότητες μεταξύ τους. Για τον σχεδιασμό του προγράμματος έχουν ληφθεί υπόψη οι παρακάτω προδιαγραφές για κάθε τύπο κάρτας\*.

Είδος κάρτας	Αριθμός ψηφίων	Αρχικά ψηφία	Αλγόριθμος επιβεβαίωσης
Visa	13 ή 16	4	Luhn formula
MasterCard	16	52	Luhn formula
American Express	15	3	Luhn formula
Diners	14	36	Luhn formula

(\*) ο πίνακας περιέχει απλουστεύσεις και είναι σωστός για τα πλαίσια τις άσκησης.

Δίνεται το πρόγραμμα `ex29`, που δημιουργεί ένα αντικείμενο τύπου `Visa` ή `MasterCard` ή `American Express` ή `Diners` με βάση την επιλογή του χρήστη. Παράγει τον αριθμό της κάρτας ψηφίο-ψηφίο ως ένα πίνακα ψηφίων (ακεραίων) `int[] number` με βάση τις παραπάνω προδιαγραφές.

Για παράδειγμα το πρώτο ψηφίο μιας κάρτας τύπου `Visa` είναι το 4, ενώ κάθε επόμενο ψηφίο δημιουργείται με τυχαίο τρόπο, με κλήση της συνάρτησης `nextInt()` της κλάσης `Java.util.Random`<sup>1</sup>.

Παράδειγμα εκτέλεσης του προγράμματος:

Which card type do you need? (1 = Visa, 2 = MasterCard, 3 = American Express, 4 = Diners) 1

How many digits for the Visa card? (13 or 16) 13

Visa Number: 4331256508156

Συμπληρώστε τον κατασκευαστή των κλάσεων `Visa`, `MasterCard`, `American Express` και `Diners`. **Κάθε κατασκευαστής δημιουργεί και ενημερώνει τον πίνακα ακεραίων `number` (της `Card`) έτσι ώστε να περιέχει τα ψηφία του αριθμού της κάρτας.** Γράψτε τον κατάλληλο αριθμό εντολών, με βάση μόνο τα κενά και προσπαθήστε να μην προσθέσετε επιπλέον εντολές-γραμμές. Μην τροποποιήσετε τον υπόλοιπο (έτοιμο) κώδικα που σας δίνεται.

### <sup>1</sup> `Java.util.Random.nextInt()`

**Declaration :** `public int nextInt(int n)`

**n :** This is the bound on the random number to be returned. Must be positive.

**Return Value :** Returns a random number between 0 (inclusive) and n (exclusive).

```

import java.util.Random;
import java.util.Scanner;

class Card {

    Random rg;
    int[] number; //The card number.

    public Card() {
        rg = new Random();
    }

    public long getNumber() {
        String temp = "";
        for(int i = 0; i < number.length; i++) temp += number[i];
        return Long.parseLong(temp);
    }

    //Validates a credit card number using Luhn's formula.
    public static boolean validate(int[] number) {
        int sum = 0;
        int length = number.length;
        for (int i = 0; i < length; i++) {
            int digit = number[length - i - 1];
            if (i % 2 == 1) digit *= 2;
            sum += digit > 9 ? digit - 9 : digit;
        }
        return sum % 10 == 0;
    }
}

//-----
class Visa extends Card {
    public Visa(int digits) {
        /* superclass's constructor is indirectly activated, thus rg is instantiated */
        number = new int[digits]; //number: is inherited from superclass Card
        number[0] = 4;
        for (int i=1; i<digits; i++)
            number[i] = rg.nextInt(9);
    }
}

```



```
// -----  
class MasterCard extends Card {  
    public MasterCard() {  
        number = new int[16];  
        number[0] = 5;  
        number[1] = 2;  
        for (int i=2; i<16; i++)  
            number[i] = rg.nextInt(9);  
    }  
}  
  
//-----  
  
class Diners extends Card {  
    public Diners() {  
        number = new int[14];  
        number[0] = 3;  
        number[1] = 6;  
        for (int i=2; i<14; i++)  
            number[i] = rg.nextInt(9);  
    }  
}  
  
//-----  
  
class AmericanExpress extends Card {  
    public AmericanExpress() {  
        number = new int[15];  
        number[0] = 3;  
        for (int i=1; i<15; i++)  
            number[i] = rg.nextInt(9);  
    }  
}
```

```

public class ex29 {

    public static void main (String args[]){

        Scanner in = new Scanner(System.in);

        System.out.println ("Which card type do you need? " +
            "(1 = Visa, 2 = MasterCard, " +
            "3 = American Express, 4 = Diners)");

        int type = in.nextInt();

        if(type == 1) {
            System.out.println ("How many digits for the Visa card? " +
                " (13 or 16)");
            int digits = in.nextInt();
            Visa v = new Visa(digits);
            while(!Card.validate(v.number)) v = new Visa(digits);
            System.out.println("Visa Number: " + v.getNumber());
        }
        else if(type == 2) {
            MasterCard mc = new MasterCard();
            while(!Card.validate(mc.number)) mc = new MasterCard();
            System.out.println("MasterCard Number: " + mc.getNumber());
        }
        else if(type == 3) {
            AmericanExpress ae = new AmericanExpress();
            while(!Card.validate(ae.number)) ae = new AmericanExpress();
            System.out.println("American Express Number: "
                + ae.getNumber());
        }
        else if(type == 4) {
            Diners d = new Diners();
            while(!Card.validate(d.number)) d = new Diners();
            System.out.println("Diners Number: " + d.getNumber());
        }
        else {
            System.out.println("Error! Card type not recognised!");
        }
    }
}

// main

}

// ex9

```

## Γενικές Παρατηρήσεις:

---

1. Για να πετύχετε το βέλτιστο αποτέλεσμα στις απαντήσεις σας, σε όποιες ασκήσεις ζητείται, να συμπληρώσετε μόνο τα κενά χωρίς να προσθέσετε επιπλέον εντολές-γραμμές.
2. Μη χρησιμοποιείτε ελληνικούς χαρακτήρες στα προγράμματά σας (java files).
3. Κάθε πρόγραμμα (java file) πρέπει να μεταγλωττίζεται και να εκτελείται στη γραμμή εντολών (όχι μέσω κάποιου IDE).
4. Η άσκηση είναι ατομική και πρέπει να γίνει από όλους τους φοιτητές, είτε επιθυμείτε να κρατήσετε εργασίες από προηγούμενη χρονιά είτε όχι.
5. Η βαθμολογία της 2<sup>ης</sup> άσκησης αντιστοιχεί στο 5% του τελικού βαθμού του μαθήματος.
6. Παρακαλούμε να υποβάλετε όλα τα αρχεία (**doc file και java files**) αφού πρώτα τα **συμπίεσετε σε ένα αρχείο** με όνομα τον αριθμό του φοιτητικού σας μητρώου.