

Π. Σταθοπούλου

pstath@ece.upatras.gr ή
pstath@upatras.gr

Ομάδα Α' (Φοιτητές με μονό αριθμό Μητρώου)

- Διδασκαλία : Παρασκευή 11πμ-13μμ ΗΛ7
- Φροντιστήριο : Δευτέρα 11πμ-12πμ ΗΛ4

Προηγούμενη Διάλεξη

Βασικές έννοιες Δομών

- * Απλές δομές & Σύνθετες δομές
- * Πίνακες δομών & Δείκτες σε δομές

Διαχείριση Αρχείων Δίσκου

- * Άνοιγμα αρχείου
- * Κλείσιμο αρχείου
- * Ανάγνωση & εγγραφή Δεδομένων

Σήμερα

Αντικειμενοστρεφής Προγραμματισμός

- * Αντικειμενοστρεφείς Γλώσσες Προγραμματισμού
- * Αντικειμενοστρεφείς Δομές
- * Η Γλώσσα Προγραμματισμού **JAVA**

Αντικειμενοστρεφής Προγραμματισμός

Διαδικαστικές & Αντικειμενοστρεφείς Γλώσσες

Η **C** θεωρείται μία γλώσσα **διαδικαστική**

Διαδικαστική γλώσσα

- * είναι η γλώσσα που εκτελεί την μία γραμμή μετά την άλλη.
- * Όμως η ροή του προγράμματος μπορεί να περνά και σε άλλα μέρη του πηγαίου κώδικα.
- * Η μεταβίβαση βασίζεται στην ανακατεύθυνση από την προηγούμενη γραμμή κώδικα.
- * Η σχεδίαση του προγράμματος βασίζεται στη χρήση διαδικασιών ή συναρτήσεων.

Αντικειμενοστρεφής Προγραμματισμός

Διαδικαστικές & Αντικειμενοστρεφείς Γλώσσες

τις τελευταίες δεκαετίες ανάπτυξη

διάφορες Αντικειμενοστρεφείς Γλώσσες
πιο δημοφιλείς

C++ & JAVA

Αντικειμενοστρεφή Γλώσσα = χρήση αντικειμένων

Τι είναι όμως αντικείμενο?

Αντικειμενοστρεφής Προγραμματισμός

Διαδικαστικές & Αντικειμενοστρεφείς Γλώσσες

Αντικείμενο

είναι μία **ανεξάρτητη** και
επαναχρησιμοποιήσιμη ενότητα κώδικα
λογισμικού

που μπορεί να εκτελεί μια συγκεκριμένη εργασία
και να αποθηκεύει τα καθορισμένα δεδομένα που
σχετίζονται με αυτή την εργασία.

Αντικειμενοστρεφής Προγραμματισμός

Διαδικαστικές & Αντικειμενοστρεφείς Γλώσσες

Οι Αντικειμενοστρεφείς Γλώσσες χρησιμοποιούν διαδικασίες που διαθέτουν πρόσθετα χαρακτηριστικά για τον ορισμό και την διαχείριση των αντικειμένων.

Γιατί αναπτύχθηκαν Αντικειμενοστρεφείς Γλώσσες ?

Για να αντιμετωπίσουν την πολυπλοκότητα των προγραμμάτων

Αντικειμενοστρεφής Προγραμματισμός

Αντικειμενοστρεφείς Δομές

Βασικά χαρακτηριστικά για τον ορισμό των αντικειμένων αποτελούν οι παρακάτω δομές:

- * Πολυμορφισμός
- * Συμπύκνωση
- * Κληρονομικότητα

Τέταρτο χαρακτηριστικό που χαρακτηρίζει μια Αντικειμενοστρεφή γλώσσα είναι η

- * Επαναχρησιμοποίηση

Αντικειμενοστρεφής Προγραμματισμός

Πολυμορφισμός

Πολυμορφισμός = πολύ + μορφή

- * Πολυμορφικό πρόγραμμα = πρόγραμμα με πολλές μορφές
- * Πολυμορφικό πρόγραμμα = πρόγραμμα που προσαρμόζεται αυτόματα

Παράδειγμα:

Σχεδιασμός ενός κύκλου

στοιχεία απαραίτητα για το σχεδιασμό

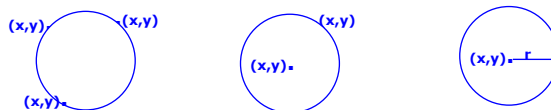
κέντρο & ακτίνα

τρία σημεία στον κύκλο

κέντρο & ένα σημείο

Αντικειμενοστρεφής Προγραμματισμός

Πολυμορφισμός



Τρεις διαφορετικές συναρτήσεις

- * Draw_cicle_with_points (int x1, int y1, int x2, int y2, int x3, int y3);
- * Draw_cicle_with_radius (int ctrX, int ctrY, long radius);
- * Draw_cicle_with_center_and_point(int ctrX, int ctrY, int x1, int y1);

Αντικειμενοστρεφής Προγραμματισμός

Παράδειγμα Πολυμορφισμού

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char* argv[])
{
    int pt_x1 = 0, pt_y1 = 0;
    int pt_x2 = 5, pt_y2 = 5;
    int pt_x3 = 0, pt_y3 = 0;
    long side = 4;

    square( pt_x1, pt_y1, pt_x2, pt_y2); // Call the square function
    printf("\n\n"); //put blank lines between squares

    square( pt_x3, pt_y3, side); //Call the square function
    return 0;
}
```

void square(int topleftX, int topleftY, int bottomleftX, int bottomleftY);

void square(int topleftX, int topleftY, long width);

Κλήσεις συναρτήσεων

Αντικειμενοστρεφής Προγραμματισμός

Παράδειγμα Πολυμορφισμού

```
void square( int topleftX, int topleftY, int bottomleftX, int bottomleftY)
{
    int xctr = 0;
    int yctr = 0;
    for ( xctr = 0; xctr < bottomleftX - topleftX; xctr++)
    {
        printf("\n");
        for ( yctr = 0; yctr < bottomleftY - topleftY; yctr++ )
        {
            printf("*");
        }
    }
}
```

Αντικειμενοστρεφής Προγραμματισμός

Παράδειγμα Πολυμορφισμού

```
void square( int topleftX, int topleftY, long
width )
{
  int xctr = 0;
  int yctr = 0;
  for ( xctr = 0; xctr < width; xctr++)
  {
    printf("\n");
    for ( yctr = 0; yctr < width; yctr++ )
    {
      printf("*");
    }
  }
}
```

Υπερφόρτωση
συνάρτησης ή μεθόδου

Αντικειμενοστρεφής Προγραμματισμός

Συμπύκνωση

**2^ο χαρακτηριστικό μιας αντικειμενοστρεφούς
γλώσσας προγραμματισμού.**

- * **Η συμπύκνωση οδηγεί στην δημιουργία
αντικειμένων που είναι «αυτάρκη».**
- * **Η συμπύκνωση παρέχει την λειτουργικότητα
«μαύρου κουτιού»**

Τι σημαίνουν όλα αυτά ?

Αντικειμενοστρεφής Προγραμματισμός

Συμπύκνωση

**αντικείμενα «αυτάρκη» & λειτουργικότητα
«μαύρου κουτιού»**

Σημαίνει ότι

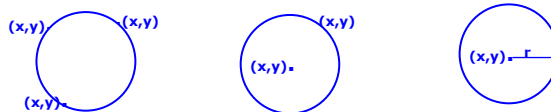
**Ο χρήστης χρησιμοποιεί τον κώδικα χωρίς να
γνωρίζει πως λειτουργεί.**

μοναδικές γνώσεις

- * κλήση της διαδικασίας &
- * η επιστροφή των αποτελεσμάτων

Αντικειμενοστρεφής Προγραμματισμός

Παράδειγμα Συμπύκνωσης

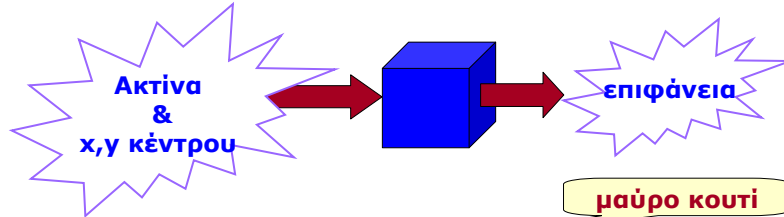


**Για την προβολή των παραπάνω κύκλων μας
ενδιαφέρει μόνο η κλήση των αντίστοιχων
ρουτινών.**

Αντικειμενοστρεφής Προγραμματισμός

Παράδειγμα Συμπύκνωσης

Area = Calculate_circle_area(center_x, center_y, radius)

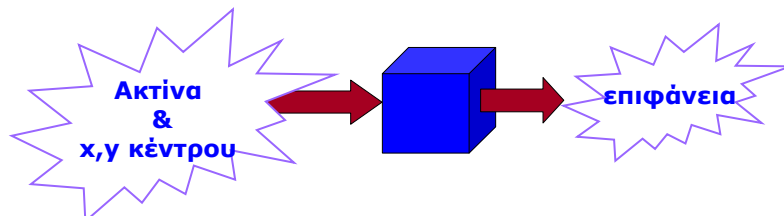


Calculate_circle_area(center_x, center_y, radius)

Αντικειμενοστρεφής Προγραμματισμός

Συμπύκνωση δεδομένων

Area = Calculate_circle_area()



Calculate_circle_area() = μαύρο κουτί

Αντικειμενοστρεφής Προγραμματισμός

Κληρονομικότητα

3^ο χαρακτηριστικό μιας αντικειμενοστρεφούς γλώσσας προγραμματισμού.

- * Η κληρονομικότητα είναι δυνατότητα δημιουργίας νέων αντικειμένων που επεκτείνουν τα χαρακτηριστικά των υπαρκτών αντικειμένων.

Αντικειμενοστρεφής Προγραμματισμός

Κληρονομικότητα (inheritance)

Έστω το αντικείμενο του τετραγώνου square με τις ακόλουθες πληροφορίες:

- * πάνω αριστερή γωνία
- * μήκος πλευράς
- * τρόπο σχεδιασμού του τετραγώνου
- * συνάρτηση για τον υπολογισμό του εμβαδού.

Χρησιμοποιώντας την κληρονομικότητα επεκτείνουμε το τετράγωνο σε ένα άλλο αντικείμενο τον **κύβο**.

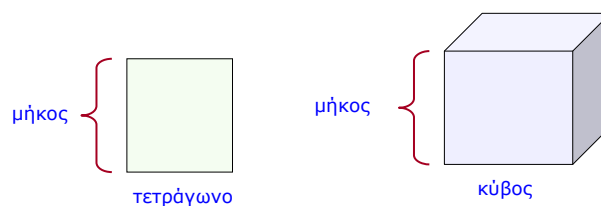
Αντικειμενοστρεφής Προγραμματισμός

Κληρονομικότητα (inheritance)

- * Το αντικείμενο του κύβου κληρονομεί τα χαρακτηριστικά του τετραγώνου **square**.
- * Όλα τα χαρακτηριστικά του αντικειμένου **square** αποτελούν μέρος του **cube**.
- * Το αντικείμενο **cube** τροποποιεί την συνάρτηση **area** για να υπολογίσει το εμβαδόν του.
- * Ο χρήστης όταν διαχειρίζεται το αντικείμενο **cube** δε χρειάζεται να γνωρίζει ότι εμπλέκεται το αντικείμενο **square**.

Αντικειμενοστρεφής Προγραμματισμός

Κληρονομικότητα (inheritance)



Επιφάνεια τετραγώνου = μήκος * μήκος

Επιφάνεια κύβου = Επιφάνεια τετραγώνου * μήκος

Αντικειμενοστρεφής Προγραμματισμός

Η σχέση των C++ & C

* Η C++ είναι **υπερσύνολο** της C και δημιουργήθηκε για να χρησιμοποιηθεί με αντικειμενοστρεφείς δομές.

* Λέξεις-κλειδιά όπως:

Class, template

προστέθηκαν για να επιτύχουν την δημιουργία αντικειμένων.

* όπως και C, η C++ χρησιμοποιήθηκε περισσότερο για την δημιουργία εκτελέσιμων προγραμμάτων.

(εκτελέσιμα, βιβλιοθήκες, δυναμικά συνδεδεμένες βιβλιοθήκες & στοιχεία ελέγχου)

Αντικειμενοστρεφής Προγραμματισμός

Η σχέση των C++ & C

```
class square {  
public:  
    square();  
    square(int);  
    int length;  
    long area();  
    int draw();  
};
```

```
class cube: public  
square {  
public:  
    cube( int );  
    long area();  
};
```

```
long square::area( void )  
{  
    return((long) length * length);  
}
```

Αντικειμενοστρεφής Προγραμματισμός

Η γλώσσα προγραμματισμού **JAVA**

- * Σχεδιάστηκε αρχικά για να προγραμματίσει μία συγκεκριμένη συσκευή
- * Δημιουργοί της **JAVA**, η εταιρεία Sun Microsystems, οραματίστηκε την διαχείριση των ηλεκτρικών οικιακών συσκευών.
- * Χαρακτηριστικά μιας τέτοιας γλώσσας:
 - * Ουδέτερη αρχιτεκτονικά
 - * Ισχυρή
 - * Αντικειμενοστρεφής
 - * Ασφαλής
 - * Απλή χωρίς περιορισμούς

Αντικειμενοστρεφής Προγραμματισμός

Η σχέση της **JAVA** με τις **C++ & C**

- * Η **JAVA** είναι η C++ με κάποια ακόμη χαρακτηριστικά
- * Η C++ είναι ισχυρή και **ευέλικτη** αλλά συνδυάζει **πολυπλοκότητα** και ελλοχεύει και **κινδύνους**
- * Η **JAVA** λοιπόν έχει σχεδιασθεί με τα παραπάνω μειονεκτήματα να τα ξεπερνά.
- * Η βασικότερη διαφορά **JAVA & C++** είναι:
 - * Η **C++** χρησιμοποιεί ενώ
 - * Η **JAVA** απαιτεί **αντικειμενοστραφή προγραμματισμό**

Αντικειμενοστρεφής Προγραμματισμός

Η Ανεξαρτησία της πλατφόρμας της JAVA

Τι σημαίνει ανεξαρτησία πλατφόρμας ?

- * Τα προγράμματα εκτελούνται σε όλες τις υπολογιστικές μηχανές χωρίς να απαιτούνται τροποποιήσεις.

Που οφείλεται η συγκεκριμένη ανεξαρτησία ?

- * Η ανεξαρτησία της πλατφόρμας **JAVA** οφείλεται στην μεταγλώττιση ειδικού τύπου

Μεταγλώττιση **bytecode**

- * Κάθε μηχανή ή πλατφόρμα διαθέτει δικό της διερμηνευτή **-Java Virtual Machine (JVM)**.

Αντικειμενοστρεφής Προγραμματισμός

Τα Πακέτα της JAVA

- * Η **JAVA** όπως όλες οι αντικειμενοστρεφείς γλώσσες συνδυάζει τα χαρακτηριστικά:

Κληρονομικότητας & επαναχρησιμοποίησης

- * επιπλέον χαρακτηριστικό

Πακέτα = Βιβλιοθήκες κλάσεων

Τα Πακέτα ισοδυναμούν με

Application Programming Interface – API

σε άλλες γλώσσες

Αντικειμενοστρεφής Προγραμματισμός

Applets & εφαρμογές JAVA

Applets είναι ειδικά προγράμματα

- * σχεδιασμένα για διανομή μέσω του internet και
- * για εκτέλεση σε ένα πρόγραμμα περιήγησης

Δεν υπάρχουν μεγάλες διαφορές μεταξύ

Applets & εφαρμογών JAVA

Βιβλιοθήκες κλάσεων της **JAVA**



Κάτι ανάλογο Βιβλιοθήκες C

Αντικειμενοστρεφής Προγραμματισμός

1^ο Παράδειγμα προγράμματος JAVA

```
public class HelloWorld {  
  
    public static void main(String args[]) {  
        Say("Hello, world!");  
    }  
    private static void Say(String message) {  
        System.out.println(message);  
    }  
}
```