

## ΚΛΑΣΕΙΣ & ΠΙΝΑΚΕΣ

### ΠΙΝΑΚΕΣ ΩΣ ΔΕΔΟΜΕΝΟ ΚΛΑΣΗΣ

Οι πίνακες μπορούν να δηλωθούν σαν *δεδομένα-μέλη* μιας κλάσης και να χρησιμοποιηθούν μέσα στις μεθόδους της κλάσης.

Π.χ δήλωση πίνακα ως δεδομένο μιας κλάσης:

```
private:
    int temp;
    float temparr[N];
```

Μεταβλητή ως δεδομένο μέλος

Πίνακας ως δεδομένο μέλος

Χρήση πίνακα μέσα σε μέθοδο με επανάληψη:

```
public: //δημόσια ορατό μέρος κλάσης
    tempMeasure(){ //constructor χωρίς όρισμα
        int i; //τοπική μεταβλητή
        for(i = 0; i < N; i++) //βρόχος N επαναλήψεων
            tempArr[i] = 0; //μηδενισμός στοιχείων πίνακα
    }
```

### ΠΙΝΑΚΕΣ ΑΝΤΙΚΕΙΜΕΝΩΝ

Μπορούμε να έχουμε σε ένα πρόγραμμα και πίνακες που να περιέχουν αντικείμενα. Παρακάτω βλέπουμε έναν πίνακα αντικειμένων *tem* της κλάσης *Temperature*.

Σε κάποιο σημείο του προγράμματος κάποιο αντικείμενο του πίνακα (ή όλα τα αντικείμενα μέσα σε βρόχο for) καλεί τη μέθοδο *readTemp()*.

```
main()
{
    Temperature tem[N]; //Πίνακας αντικειμένων της κλάσης Temperature
    .....
    tem[i].readTemp(); //Κλήση μεθόδου από αντικείμενο
}
```

### ΠΑΡΑΔΕΙΓΜΑΤΑ ΠΙΝΑΚΩΝ ΜΕΣΑ ΣΕ ΚΛΑΣΕΙΣ

Πίνακας ως δεδομένο μιας κλάσης.

```
private:
    char city[30];
    float temperature[N]; //Πίνακας ως δεδομένο μέλος
```

Πίνακας ως μέλος μεθόδου.

```
void readData()
{
    cout<<"Dose A.M.:";
    cin >> arMht;
    cout <<"Dose Vathmos: ";
    for (int i = 0; i < N; i++)
        cin >> vathmos[i]; //Πίνακας ως μέλος μεθόδου
}
```

Πίνακας αντικειμένων μιας κλάσης.

```
int main()
{
    Temperature Tem[N]; //Πίνακας αντικειμένων
    .....
    for (i = 0; i < N; i++){
        Tem[i].setTemp(); //Τα αντικείμενα καλούν τη μέθοδο setTemp()
    }
    .....
}
```

Αρχικοποίηση στοιχείων πίνακα με constructor.

```
class City
{
private:
    char Polh[30]; //Πίνακας χαρακτήρων
    float Temperatures[N]; //Πίνακας αντικειμένων
public:
    City(){ //Constructor χωρίς ορίσματα
        for (int i = 0; i < N; i++)
            Temperatures[i] = 0.0; //Αρχικοποίηση του πίνακα θερμοκρασιών
    }
    ~City(){ //Destructor
    }
}; //End of class
```

Εύρεση Μ.Ο μισθών N υπαλλήλων.

```
//Μέσα σε ένα βρόχο for.....
for (i = 0; i < N; i++)
    //.....τα αντικείμενα επιστρέφουν το μισθό τους
    // τα οποία αθροίζονται σε μία μεταβλητή sum
    sum = sum + erg[i].retMisthos();

avg = sum / N; //Υπολογισμός του Μ.Ο.

cout << "\no Μ.Ο. einai: " << avg << endl; //Εκτύπωση του Μ.Ο.
```

## ΠΑΡΑΔΕΙΓΜΑΤΑ ΔΗΛΩΣΗΣ ΚΑΙ ΧΡΗΣΗΣ ΠΙΝΑΚΩΝ

### Παράδειγμα 1: ΠΙΝΑΚΑΣ ΩΣ ΔΕΔΟΜΕΝΟ ΜΕΛΟΣ ΚΛΑΣΗΣ

Στο παρακάτω πρόγραμμα ορίζεται μία κλάση CityTemp με δεδομένα το όνομα της πόλης και έναν πίνακα που αποθηκεύονται 10 θερμοκρασίες.

```
#include <iostream>
#include <string.h>
#define N 5 //Αναγνωριστικό
using namespace std;

class City{ //Δήλωση της κλάσης

private:
    char nameOfCity[20];
    float tempOfCity[N]; //Πίνακας ως δεδομένο μέλος
public:
    //Constructor χωρίς ορίσματα
    City(){
        strcpy(nameOfCity, "+No Name+");
        for (int i = 0; i < N; i++){
            //Αρχικοποίηση του πίνακα στην τιμή 0
            tempOfCity[i] = 0;
        }
    }

    //Destructor
    ~City(){
    }

    //Μέθοδος για την εισαγωγή τιμών στα δεδομένα
    void readData(){
        cout << "Give a City name:";    cin >> nameOfCity;
        cout << "Give Temperatures:" << endl;
        for (int i = 0; i < N; i++){
            cout << i+1 << ": ";
            cin >> tempOfCity[i];
        }
        cout << endl;
    }

    //Μέθοδος για την εμφάνιση του ονόματος της πόλης
    void printNameOfCity(){
        cout << nameOfCity << endl;
    }

    //Μέθοδος για την εμφάνιση των θερμοκρασιών
    void GetTemp(){
        for (int i = 0; i < N; i++)
            cout << i+1 << " Temp: " << tempOfCity[i] << " oC" << endl;
    }
};

int main(){

    City firstCity, secondCity; //Κατασκευή αντικειμένων
    int printData; //Βοηθητική μεταβλητή
    char choice; //Βοηθητική μεταβλητή

    cout << "====DATA ENTRY====" << endl;
```

```

cout << endl;

cout << "====(1st City)====" << endl;
firstCity.readData(); //Εισαγωγή στοιχείων πρώτης πόλης

cout << "====(2nd City)====" << endl;
secondCity.readData(); //Εισαγωγή στοιχείων δεύτερης πόλης

do{
    cout << "====City Informations====" << endl;
    cout << "Select City Press 1->(1st City), 2->(2nd City): ";
    cin >> printData;

    if(printData == 1){
        firstCity.printNameOfCity(); //Εκτύπωση στοιχείων πρώτης πόλης
        firstCity.GetTemp();
    }
    else{
        secondCity.printNameOfCity(); //Εκτύπωση στοιχείων δεύτερης πόλης
        secondCity.GetTemp();
    }
    //Επανάληψη εκτυπώσεων
    cout << "Continue Y->(Yes), N->(No): "; cin >> choice;

}while(choice == 'Y' || choice == 'y');

cout << "\nEND OF PROGRAM!!!" << endl;

return 0;
}

```

## Παράδειγμα 2: ΠΙΝΑΚΑΣ ΑΝΤΙΚΕΙΜΕΝΩΝ ΣΤΟ ΚΥΡΙΩΣ ΠΡΟΓΡΑΜΜΑ

Στο παρακάτω πρόγραμμα καταγράφονται τα στοιχεία 5 εργαζομένων. Στη συνέχεια γίνεται αναζήτηση συγκεκριμένου εργαζόμενου με την πληκτρολόγηση του κωδικού του. Αν ο εργαζόμενος βρεθεί εκτυπώνονται τα στοιχεία του αλλιώς αν δεν υπάρχει το πρόγραμμα εμφανίζει μήνυμα λάθους.

```

#include <iostream>
#include <string.h>
#define Multitude 5 //Αναγνωριστικό
using namespace std;

class Employees{ //Δημιουργία κλάσης

private:
    int code;
    char name[20];
    float salary;
public:
    //Constructor χωρίς ορίσματα
    Employees(){
        code = 0;
        strcpy(name, "NoName");
        salary = 0;
    }

    //Destructor
    ~Employees(){}

    //Μέθοδος για την εισαγωγή στοιχείων
    void readInformations(){
        cout << "Give Code: "; cin >> code;
        cout << "Give Name: "; cin >> name;
    }
};

```

```

        cout << "Give Salary: "; cin >> salary;
    }

    //Μέθοδος για την εκτύπωση στοιχείων
    void printInformations(){
        cout << "Employee Code: " << code << endl;
        cout << "Employee Name: " << name << endl;
        cout << "Employee Salary: " << salary << endl;
    }

    //Μέθοδος για την επιστροφή κωδικού
    int getCode(){
        return code;
    }
}; //Τέλος κλάσης

int main(){
    //Δημιουργία πίνακα 5 αντικειμένων
    Employees emp[Multitude];

    //Βοηθητικές μεταβλητές
    int i,j, _code, choice;
    bool found = false;

    // Βρόχος για την εισαγωγή στοιχείων
    cout <<"Give informations for " << Multitude <<" Employees " <<endl;
    for (i = 0; i < Multitude; i++){
        cout << endl << "Employee " << i+1 << ":" <<endl;
        //Κλήση της μεθόδου readInformations από κάθε αντικείμενο
        emp[i].readInformations();
    }

    do{
        //Έναρξη αναζήτησης
        cout << "====Start Searching====";
        cout << "\nGive Code: ";
        cin >> _code;

        //Βοηθητικές μεταβλητές
        j = 0;
        found = false;

        //Βρόχος αναζήτησης
        while (j < Multitude && found == false){
            //Σύγκριση με κωδικό
            if (emp[j].getCode() == _code) //Εάν βρεθεί ο κωδικό
                found = true;
            else //Αλλιώς
                j++; //Προχώρα στον επόμενο
        }

        if (found == true) // Ο εργαζόμενος βρέθηκε
            emp[j].printInformations(); //εκτύπωσε τα στοιχεία του
        else
            //Ο εργαζόμενος δεν υπάρχει
            cout << "The employee does not exist!!!" << endl << endl;

        //Επανάληψη αναζήτησης
        cout << "To try again press 1->YES, 2->NO: "; cin >> choice;
    }while(choice == 1);
}

```

```
    cout << "Bye Bye!!!" << endl;  
    return 0;  
} //Τέλος main
```