

**Ανοικτά Ακαδημαϊκά Μαθήματα**

**Τεχνολογικό Εκπαιδευτικό Ίδρυμα Αθήνας**

**Αντικειμενοστρεφής Προγραμματισμός (Ε)**

**Ενότητα 1:** Βασικά χαρακτηριστικά της C++

Κλειώ Σγουροπούλου

Τμήμα Μηχανικών Πληροφορικής ΤΕ

|  |  |
| --- | --- |
| Το περιεχόμενο του μαθήματος διατίθεται με άδεια Creative Commons εκτός και αν αναφέρεται διαφορετικά | Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους. |

# Περιεχόμενα

[Αντικειμενοστρεφής Προγραμματισμός- Εργαστήριο 1- Λύσεις 3](#_Toc402941463)

[Άσκηση 1 3](#_Toc402941464)

[Άσκηση 2 4](#_Toc402941465)

[Άσκηση 3 6](#_Toc402941466)

# Αντικειμενοστρεφής Προγραμματισμός- Εργαστήριο 1- Λύσεις

## Άσκηση 1

**#include <iostream>**

**#include <string>**

**using namespace** std;

**class** Vehicle

{

**public**:

string company;

string model;

**int** year;

string plate;

**int** type; //1:car, 2:truck, 3:motorcycle

**void** display()

{

cout << **"Vehicle details: "** << endl;

cout << **"Company: "** << company << endl;

cout << **"Model: "** << model << endl;

cout << **"Year: "** << year << endl;

cout << **"Licence plate no.: "** << plate << endl;

**if**(type == 1)

cout << **"Type: Car"** << endl;

**else if**(type == 2)

cout << **"Type: Truck"** << endl;

**else if**(type == 3)

cout << **"Type: Motorcycle"** << endl;

**else**

cout << **"Error! Unknown type!"** << endl;

}

};

**int** main(**int** argc, char\* argv[])

{

Vehicle v;

v.company = **"Toyota"**; v.model =

**"Avensis"**; v.year =

2007; v.plate =

**"IAA2333";**

//Σκεφτείτε τι θα συνέβαινε αν δίναμε type = 5!

//Για αυτό τον λόγο είναι απαραίτητες οι accessor functions

v.type = 1; // Το σωστό θα ήταν: v.setType(5) --> θα μας ειδοποιούσε για τον

λάθος τύπο

v.display();

system**("pause"**);

**return** 0;

}

## 

## Άσκηση 2

Παρατηρήστε την χρησιμότητα των accessor functions, καθώς κάνουν έλεγχο για τις τιμές που θα

δώσει ο χρήστης!

**#include <iostream>**

**#include <string>**

**using namespace** std;

**class** Vehicle

{

string company; **//τώρα είναι private**

string model;

**int** year;

string plate;

**int** type; **//1:car, 2:truck, 3:motorcycle**

**public**:

**void** setCompany(string \_company)

{

company = \_company;

}

**void** setModel(string \_model)

{

model = \_model;

}

**void** setYear(int \_year)

{

**//και εδώ μπορούμε να κάνουμε έλεγχο ορθότητας της ημερομηνίας**

**if**(\_year > 1900 && \_year < 2012)

year = \_year;

**else**

{

cout << **"Error in vehicle year! Please try again"** << endl;

year = 2012;

}

}

**void** setPlate(string \_plate)

{

plate = \_plate;

}

**void** setType(int \_type)

{

**if**(\_type == 1 || \_type == 2 || \_type == 3)

{

type = \_type;

}

**else**

{

}

**//Εδώ μπορούμε εισαγάγουμε μια σωστή default τιμή.**

**//Αλλιώς μπορούμε να θέσουμε type = 0**

**//Η πιο σωστή αντιμετώπιση θα ήταν με exception**

**//(θα το δούμε σε επόμενο μάθημα)**

cout << **"Error in vehicle type! Please try again"** << endl;

type = 1;

}

## Άσκηση 3

**#include <iostream>**

**#include <string>**

**using namespace** std;

**class** Vehicle

{

string company; **//τώρα είναι private**

string model;

int year;

string plate;

int type; **//1:car, 2:truck, 3:motorcycle**

**public**:

**void** setCompany(string \_company)

{

company = \_company;

}

**void** setModel(string \_model)

{

model = \_model;

}

**void** setYear(**int** \_year)

{

**//και εδώ μπορούμε να κάνουμε έλεγχο ορθότητας της ημερομηνίας**

**if**(\_year > 1900 && \_year < 2012)

year = \_year;

**else**

{

cout << **"Error in vehicle year! Please try again"** << endl;

year = 2012;

}

}

**void** setPlate(string \_plate)

{

plate = \_plate;

}

**void** setType(**int** \_type)

{

**if**(\_type == 1 || \_type == 2 || \_type == 3)

{

type = \_type;

}

**else**

{

**//Εδώ μπορούμε εισαγάγουμε μια σωστή default τιμή.**

**//Αλλιώς μπορούμε να θέσουμε type = 0**

**//Η πιο σωστή αντιμετώπιση θα ήταν με exception**

**//(θα το δούμε σε επόμενο μάθημα)**

cout << **"Error in vehicle type! Please try again"** << endl;

type = 1;

}

}

void setValues(string \_company, string \_model, int \_year, string \_plate, int

\_type)

{

company = \_company;

model = \_model;

plate = \_plate;

**//Year**

**if**(\_year > 1900 && \_year < 2012)

year = \_year;

**else**

{

cout << **"Error in vehicle year! Please try again"** << endl;

year = 2012;

}

**//Car type**

**if**(\_type == 1 || \_type == 2 || \_type == 3)

{

type = \_type;

}

**else**

{

cout << **"Error in vehicle type! Please try again"** << endl; type = 1;

}

}

int getVehicleAge(int current\_year)

{

**if**(current\_year < year)

{

cout << **"Error! Car date is later than the year you provided "**

<< endl;

**return** 0; **//κάτι πρέπει να επιστρέψουμε εδώ!**

}

**Else**

**return** current\_year - year;

}

**void** display()

{

cout << **"Vehicle details: "** << endl;

cout << **"Company: "** << company << endl;

cout << **"Model: "** << model << endl;

cout << **"Year: "** << year << endl;

cout << **"Licence plate no.: "** << plate << endl;

**if**(type == 1)

cout << **"Type: Car"** << endl;

**else if**(type == 2)

cout << **"Type: Truck"** << endl;

**else if**(type == 3)

cout << **"Type: Motorcycle"** << endl;

**else**

cout << **"Error! Unknown type!"** << endl;

}

};

**int** main(**int** argc, char\* argv[])

{

Vehicle v;

v.setValues(**"Toyota"**, **"Avensis",**2007, **"IAA2333"**, 1);

cout << **"Vehicle is: "** << v.getVehicleAge(2012) << **" year old."**<< endl;

system(**"pause"**);

**return** 0;

}

|  |
| --- |
| **Ανοικτά Ακαδημαϊκά Μαθήματα**  **Τεχνολογικό Εκπαιδευτικό Ίδρυμα Αθήνας** |
| **Τέλος Ενότητας** |
| **Χρηματοδότηση**   * Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα. * Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο ΤΕΙ Αθήνας**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού. * Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους. |

**Σημειώματα**

**Σημείωμα Αναφοράς**

Copyright ΤΕΙ Αθήνας, Κλειώ Σγουροπούλου, 2014. Κλειώ Σγουροπούλου. «Αντικειμενοστρεφής Προγραμματισμός». Ενότητα 1: Βασικά χαρακτηριστικά της C++ Λύσεις ασκήσεων». Έκδοση: 1.0. Αθήνα 2014. Διαθέσιμο από τη δικτυακή διεύθυνση: [ocp.teiath.gr](https://ocp.teiath.gr/).

**Σημείωμα Αδειοδότησης**

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».

[](file:///C:\Users\pantelis\Downloads\%5b1%5d%20http:\creativecommons.org\licenses\by-nc-sa\4.0\)

[1] http://creativecommons.org/licenses/by-nc-sa/4.0/

Ως **Μη Εμπορική** ορίζεται η χρήση:

* που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
* που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
* που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

**Διατήρηση Σημειωμάτων**

* Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:
* Το Σημείωμα Αναφοράς
* Το Σημείωμα Αδειοδότησης
* Τη δήλωση Διατήρησης Σημειωμάτων
* Το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει) μαζί με τους συνοδευόμενους υπερσυνδέσμους.