



Πανεπιστήμιο Δυτικής Αττικής  
Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

```
bpy.context.scene.objects.active = modifier_ob  
print("Selected" + str(modifier_ob)) # modifier ob selected
```

# Αντικειμενοστρεφής Προγραμματισμός (C++)

**Ενότητα 5: Είσοδος/Έξοδος**

**Καθηγήτρια Κλειώ Σγουροπούλου**

# Αρχεία εισόδου/εξόδου 1 / 3

Η C++ υποστηρίζει αρχεία εισόδου/εξόδου μέσω των κλάσεων.

- **ofstream**: Υποστήριξη λειτουργιών εγγραφής
- **ifstream**: Υποστήριξη λειτουργιών ανάγνωσης
- **fstream**: Υποστήριξη λειτουργιών εγγραφής/ανάγνωσης
- Άνοιγμα αρχείου μέσω δημιουργίας **αντικειμένου ρεύματος (stream object)** και χρήσης μεθόδου **open()**

```
void open (const char * filename, openmode mode) ;
```

- Κλείσιμο αρχείου μέσω συνάρτησης close()

# Αρχεία εισόδου/εξόδου 2/3

- Mode: συνδυασμός των ακόλουθων flags (μέσω τελεστή OR - |)
  - **ios::in** Αρχείο για ανάγνωση
  - **ios::out** Αρχείο για εγγραφή
  - **ios::ate** Αρχική θέση: τέλος αρχείου
  - **ios::app** Κάθε έξοδος προστίθεται στο τέλος του αρχείου
  - **ios::trunc** Αν το αρχείο προϋπάρχει, σβήνεται
  - **ios::binary** Δυαδικά αρχεία
- Παράδειγμα  
**ofstream file;**  
**file.open ("example.bin", ios::out | ios::app | ios::binary);**

# Αρχεία εισόδου/εξόδου 3/3

- Default Mode:

Class	Default <i>mode</i> to parameter
Ofstream	ios::out   ios::trunc
Ifstream	ios::in
Fstream	ios::in   ios::out

- Constructor των κλάσεων ρευμάτων:

```
ofstream file ("example.bin", ios::out | ios::app |  
ios::binary) ;
```

- Έλεγχος αν ένα αρχείο έχει αρχικοποιηθεί μέσω συνάρτησης

```
is_open() :
```

```
bool is_ospen() ;
```

# Αρχεία κειμένου 1 / 2

```
// writing on a text file
#include <fstream.h>

int main () {
    ofstream examplefile ("example.txt");
    if (examplefile.is_open())
    {
        examplefile << "This is a line.\n";
        examplefile << "This is another line.\n";
        examplefile.close();
    }
    return 0;
}
```

*file example.txt*

This is a line.

This is another  
line.

# Αρχεία κειμένου 2/2

```
// reading a text file
#include <iostream.h>
#include <fstream.h>
#include <stdlib.h>

int main () {
    char buffer[256];
    ifstream examplefile ("example.txt");
    if (! examplefile.is_open())
    { cout << "Error opening file"; exit (1); }

    while (! examplefile.eof() )
    {
        examplefile.getline (buffer,100);
        cout << buffer << endl;
    }
    return 0;
}
```

```
This is a line.
This is another
line.
```

# Μέθοδοι κατάστασης ρευμάτων

Μέθοδοι για την επαλήθευση της κατάστασης των ρευμάτων:

- **eof()**: true - τέλος αρχείου
- **bad()**: true – σφάλμα κατά τη λειτουργία εγγραφής ή ανάγνωσης
- **fail()**: όπως η **bad()** και επιπλέον λάθη format
  - ▣ π.χ. προσπάθεια ανάγνωσης integer και λαμβανεται char
- **good()**: γενική, false στις καταστάσεις όπου οποιαδήποτε από τις προηγούμενες θα επέστρεφε true

# Δείκτες ρευμάτων `get` και `put` 1/3

Όλα τα ρεύματα εισόδου/εξόδου διαθέτουν τουλάχιστον ένα δείκτη:

- `istream`: Δείκτη `get`, δείχνει στο επόμενο προς ανάγνωση στοιχείο
- `ostream`: Δείκτη `put`, δείχνει τη θέση στην οποία θα εγγραφεί το επόμενο στοιχείο
- `fstream`: Δείκτες `get` και `put`



# Δείκτες ρευμάτων **get** και **put** 2/3

Ο χειρισμός των παραπάνω δεικτών γίνεται με τη βοήθεια των ακόλουθων μεθόδων:

- **tellg()** , **tellp()**: επιστρέφουν **pos\_type** - τρέχουσα θέση του δείκτη
- **seekg()** , **seekp()**: αλλάζουν την τρέχουσα θέση
  - ▣ **seekg** ( **pos\_type** *position* );  
**seekp** ( **pos\_type** *position* );
  - ▣ **seekg** ( **off\_type** *offset*, **seekdir** *direction* );  
**seekp** ( **off\_type** *offset*, **seekdir** *direction* );

ios: :beg	offset specified from the beginning of the stream
ios: :cur	offset specified from the current position of the stream pointer
ios: :end	offset specified from the end of the stream

# Δείκτες ρευμάτων `get` και `put` 3/3

```
/ obtaining file size
#include <iostream.h>
#include <fstream.h>

const char * filename = "example.txt";

int main () {
    long l,m;
    ifstream file (filename, ios::in|ios::binary);
    l = file.tellg();
    file.seekg (0, ios::end);
    m = file.tellg();
    file.close();
    cout << "size of " << filename;
    cout << " is " << (m-1) << " bytes.\n";
    return 0;
}
```

size of  
example.txt is  
40 bytes.

# Δυαδικά αρχεία 1 / 2

Χρήση μεθόδων `write()` και `read()`:

```
write ( char * buffer, streamsize size );
```

```
read ( char * buffer, streamsize size );
```

```
// reading binary file
#include <iostream.h>
#include <fstream.h>

const char * filename = "example.txt";

int main () {
    char * buffer;
    long size;
    ifstream file (filename,
ios::in|ios::binary|ios::ate);
    size = file.tellg();
```

# Διαδικά αρχεία 2/2

The complete  
file is in  
buffer

```
file.seekg (0, ios::beg);  
buffer = new char [size];  
file.read (buffer, size);  
file.close();  
cout << "the complete file is in a buffer";  
delete[] buffer;  
return 0;  
}
```

# ΤΕΛΟΣ ΕΝΟΤΗΤΑΣ



# Σημειώματα



# Σημείωμα Αναφοράς

Copyright Πανεπιστήμιο Δυτικής Αττικής, Κλειώ Σγουροπούλου 2020. Κλειώ Σγουροπούλου. «Αντικειμενοστραφής Προγραμματισμός-Θ. Ενότητα 1: Περιήγηση στη C++». Έκδοση: 1.0. Αθήνα 2020. Διαθέσιμο από τη δικτυακή διεύθυνση: [eclass.uniwa.gr](http://eclass.uniwa.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. Οι όροι χρήσης των έργων τρίτων επεξηγούνται στη διαφάνεια «Επεξήγηση όρων χρήσης έργων τρίτων».

Τα έργα για τα οποία έχει ζητηθεί άδεια αναφέρονται στο «Σημείωμα Χρήσης Έργων Τρίτων».



[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.



# Επεξήγηση όρων χρήσης έργων τρίτων

©	Δεν επιτρέπεται η επαναχρησιμοποίηση του έργου, παρά μόνο εάν ζητηθεί εκ νέου άδεια από το δημιουργό.
διαθέσιμο με άδεια CC-BY	Επιτρέπεται η επαναχρησιμοποίηση του έργου και η δημιουργία παραγώγων αυτού με απλή αναφορά του δημιουργού.
διαθέσιμο με άδεια CC-BY-SA	Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού, και διάθεση του έργου ή του παράγωγου αυτού με την ίδια άδεια.
διαθέσιμο με άδεια CC-BY-ND	Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού. Δεν επιτρέπεται η δημιουργία παραγώγων του έργου.
διαθέσιμο με άδεια CC-BY-NC	Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού. Δεν επιτρέπεται η εμπορική χρήση του έργου.
διαθέσιμο με άδεια CC-BY-NC-SA	Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού και διάθεση του έργου ή του παράγωγου αυτού με την ίδια άδεια. Δεν επιτρέπεται η εμπορική χρήση του έργου.
διαθέσιμο με άδεια CC-BY-NC-ND	Επιτρέπεται η επαναχρησιμοποίηση του έργου με αναφορά του δημιουργού. Δεν επιτρέπεται η εμπορική χρήση του έργου και η δημιουργία παραγώγων του.
διαθέσιμο με άδεια CC0 Public Domain	Επιτρέπεται η επαναχρησιμοποίηση του έργου, η δημιουργία παραγώγων αυτού και η εμπορική του χρήση, χωρίς αναφορά του δημιουργού.
διαθέσιμο ως κοινό κτήμα	Επιτρέπεται η επαναχρησιμοποίηση του έργου, η δημιουργία παραγώγων αυτού και η εμπορική του χρήση, χωρίς αναφορά του δημιουργού.
χωρίς σήμανση	Συνήθως δεν επιτρέπεται η επαναχρησιμοποίηση του έργου.

# 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.