



Ελληνική Δημοκρατία  
Τεχνολογικό Εκπαιδευτικό  
Ίδρυμα Ηπείρου

# Προγραμματισμός Ι

Ενότητα 10 : Αρχεία

Αλέξανδρος Τζάλλας



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Τμήμα Μηχανικών Πληροφορικής Τ.Ε

## Προγραμματισμός Ι

### Ενότητα 10 : Αρχεία

Αλέξανδρος Τζάλλας

Λέκτορας

Άρτα, 2015



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης





# Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.





# Χρηματοδότηση

- 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.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο ΤΕΙ Ηπείρου**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



# Σκοποί ενότητας

- Να γίνει σαφής ο τρόπος χειρισμού των αρχείων.
- Να περιγραφεί ο τρόπος ανοίγματος ή κλεισίματος, επεξεργασίας και χειρισμού αρχείων κειμένου.
- Να περιγραφεί ο τρόπος χειρισμού αρχείων με τύπο.



# Περιεχόμενα ενότητας

- Εισαγωγικές έννοιες
- Αρχεία Κειμένου.
- Άνοιγμα ή κλείσιμο αρχείων
- Επεξεργασία αρχείων
- Χειρισμός αρχείων
- Αρχεία Με Τύπο



# Εισαγωγικά

- **Αρχείο:** Σύνθετος τύπος δεδομένων με δόμηση.
- Όπως στα σύνολα και στις εγγραφές έτσι και στο αρχείο δίνεται ένα όνομα, η προσπέλαση όμως σε κάθε συνιστώσα γίνεται ξεχωριστά.
- Η προσπέλαση σε μία συνιστώσα γίνεται με το όνομά της ανεξάρτητα από τη θέση της, ενώ σε τύπους με δόμηση η προσπέλαση γίνεται σύμφωνα με τη θέση της στη συλλογή και όχι με το όνομά της.
- Επιτρεπτές πράξεις: Εκχώρηση τιμών, αποθήκευση, ανάκτηση μεμονωμένων συνιστωσών.



# Αρχεία Κειμένου<sub>1/2</sub>

- Αποτελούνται από χαρακτήρες και χωρίζονται σε γραμμές οι οποίες τελειώνουν με το χαρακτήρα μεταφοράς στην επόμενη γραμμή (**CR/LF**).
- Για να χρησιμοποιήσουμε ένα αρχείο κειμένου, πρέπει να του δώσουμε ένα όνομα και να δηλώσουμε τον τύπο του με τη λέξη **Text**.
- Για να χρησιμοποιήσουμε ένα αρχείο για είσοδο ή έξοδο πρέπει να δώσουμε και το πραγματικό όνομα του αρχείου του δίσκου στο οποίο αναφέρεται → διαδικασία **assign**.





# Αρχεία Κειμένου<sub>2/2</sub>

```
var mathites: text;
```

```
...
```

```
begin
```

```
...
```

```
assign (mathites,'C:/students.dat');...
```

- Το όνομα του αρχείου στο δίσκο (**c:students.dat**), στο οποίο αναφέρεται το αρχείο με τη χρήση της διαδικασίας **assign**, χρησιμοποιείται μόνο μία φορά.
- Όλες οι άλλες αναφορές του αρχείου στο πρόγραμμα γίνονται με το όνομά του, όπως αυτό δηλώθηκε στην περιοχή δήλωσης των μεταβλητών (**mathites**).



# Άνοιγμα/κλείσιμο αρχείων<sub>1/3</sub>

- **reset (F1):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή assign.
- *Ενέργεια:* Η εντολή αυτή ανοίγει ένα αρχείο που υπάρχει, με δυνατότητα ανάγνωσης ή εγγραφής σε αυτό.
- Αν το αρχείο είναι τύπου text, τότε το αρχείο μπορεί να χρησιμοποιηθεί μόνο για ανάγνωση.



# Άνοιγμα/κλείσιμο αρχείων<sub>2/3</sub>

- **rewrite (F1):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή `assign`.
- *Ενέργεια:* Η εντολή αυτή δημιουργεί ένα αρχείο για εγγραφή.
- Αν το αρχείο αυτό τότε τα δεδομένα που είναι αποθηκευμένα στο αρχείο καταστρέφονται.



# Άνοιγμα/κλείσιμο αρχείων<sub>3/3</sub>

- **close (F1):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή assign.
- *Ενέργεια:* Η εντολή αυτή κλείνει το αρχείο F1 και ενημερώνει τον κατάλογο (directory).
- Μετά τη χρήση του αρχείου και πριν τελειώσει το πρόγραμμα, χρησιμοποιείται η εντολή close για το κλείσιμο του αρχείου.
- Εάν δε χρησιμοποιηθεί και έχουμε ανοίξει αρχείο για εγγραφή, τότε είναι πιθανό να χάσουμε ορισμένα από τα τελευταία δεδομένα που γράφτηκαν.



# Άνοιγμα/κλείσιμο αρχείων<sub>4/4</sub>

- **append (F1):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή assign.
- *Ενέργεια:* Ανοίγει ένα αρχείο τύπου κειμένου για εγγραφή και συγκεκριμένα για πρόσθεση νέων δεδομένων στο τέλος του αρχείου και τα νέα δεδομένα δε σβήνουν τα παλιά.



# Επεξεργασία αρχείων<sub>1/5</sub>

- **read/ readln (F1, M1, M2, ..., Mn):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή assign και M1, M2,...,Mn είναι τα ονόματα των μεταβλητών.
- *Ενέργεια:* Η εντολή read διαβάζει δεδομένα από το αρχείο F1 και τα αποθηκεύει στις μεταβλητές M1, M2,...,Mn.
- Η εντολή readln, η οποία χρησιμοποιείται μόνο σε αρχεία κειμένου, διαβάζει δεδομένα με τον ίδιο τρόπο αλλά μετά την εκτέλεσή της ο δείκτης του αρχείου μετακινείται προς την επόμενη γραμμή του κειμένου του αρχείου.



# Επεξεργασία αρχείων<sub>2/5</sub>

- **eof (F1):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή assign.
- *Ενέργεια:* Η συνάρτηση είναι boolean και επιστρέφει την τιμή true, αν βρεθεί η ένδειξη τέλους του αρχείου (τέλος δεδομένων), αλλιώς, επιστρέφει την τιμή false.



# Επεξεργασία αρχείων<sub>3/5</sub>

- **seekeof (F1):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή assign.
- *Ενέργεια:* Η συνάρτηση είναι boolean και επιστρέφει την τιμή true, αν βρεθεί η ένδειξη τέλους του αρχείου (τέλος δεδομένων), αλλιώς, επιστρέφει την τιμή false.
- Είναι αντίστοιχη της συνάρτησης eof με τη διαφορά ότι παραλείπει όλα τα κενά και τους χαρακτήρες τέλους γραμμής.





# Επεξεργασία αρχείων<sub>4/5</sub>

- **eoln (F1):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή assign.
- *Ενέργεια:* Η συνάρτηση δηλώνει αν υπάρχουν ακόμη χαρακτήρες στην τρέχουσα γραμμή.
- Η συνάρτηση επιστρέφει την τιμή true, αν διαβάζοντας βρει το τέλος της γραμμής ή το τέλος του αρχείου, αλλιώς επιστρέφει την τιμή false.



# Επεξεργασία αρχείων<sub>5/5</sub>

- **loresult:**
- *Ενέργεια:* Η συνάρτηση επιστρέφει μία ακέραια τιμή που δείχνει την κατάσταση της τελευταίας εντολής εισόδου/εξόδου/ Η συνάρτηση είναι χωρίς παράμετρο.
- Αν η τελευταία λειτουργία εισόδου/εξόδου τελειώσει επιτυχώς, τότε η συνάρτηση επιστρέφει την τιμή 0 (μηδέν).
- Για να λειτουργήσει η συνάρτηση πρέπει να έχει δηλωθεί προς τον compiler της Pascal η δήλωση `{SI-}` που δίνει στο χρήστη τη δυνατότητα ελέγχου των λαθών που έχουν σχέση με είσοδο και έξοδο.



# Χειρισμός αρχείων<sub>1/3</sub>

- **eof (F1):** Όπου F1 είναι μεταβλητή τύπου αρχείου, και F2 το όνομα του αρχείου στο Λειτουργικό Σύστημα.
- *Ενέργεια:* Η εντολή αυτή αντιστοιχίζει στη μεταβλητή F1 το όνομα του αρχείου F2, έτσι όπως έχει οριστεί στο Λειτουργικό Σύστημα.
- **erase (F1):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή assign.
- *Ενέργεια:* Η εντολή διαγράφει από το δίσκο το αρχείο που αντιστοιχεί στην μεταβλητή F1.



# Χειρισμός αρχείων<sub>2/3</sub>

- **assign (F1, F2):** Όπου F1 είναι μεταβλητή τύπου αρχείου, και F2 το όνομα του αρχείου στο Λειτουργικό Σύστημα.
- *Ενέργεια:* Η εντολή αυτή αντιστοιχίζει στη μεταβλητή F1 το όνομα του αρχείου F2, έτσι όπως έχει οριστεί στο Λειτουργικό Σύστημα.
- **erase (F1):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή assign.
- *Ενέργεια:* Η εντολή διαγράφει από το δίσκο το αρχείο που αντιστοιχεί στην μεταβλητή F1.



# Χειρισμός αρχείων<sub>3/3</sub>

- **assign (F1, F2):** Όπου F1 είναι μεταβλητή τύπου αρχείου, και F2 το όνομα του αρχείου στο Λειτουργικό Σύστημα.
- *Ενέργεια:* Η εντολή αυτή αντιστοιχίζει στη μεταβλητή F1 το όνομα του αρχείου F2, έτσι όπως έχει οριστεί στο Λειτουργικό Σύστημα.
- **erase (F1):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή assign.
- *Ενέργεια:* Η εντολή διαγράφει από το δίσκο το αρχείο που αντιστοιχεί στην μεταβλητή F1.



# Παράδειγμα 1

Εγγραφή σε αρχείο  
κειμένου

```
Program paradeigma_14;  
Var fname, txt : string;  
Userfile : text;  
Begin  
Write('dose onoma arxeiou  
(px.c:\test.txt) :');  
Readln(fname);  
Assign(userfile, fname);  
Rewrite(userfile);  
Writeln('grapse to kimeno : ');  
Readln(txt);  
Writeln(userfile,txt);  
Close(userfile);  
End.
```



# Παράδειγμα 2

Ανάγνωση από  
αρχείο κειμένου

```

Program paradeigma_15;
Var userfile : text;
Filename, tfile : string;
Begin
Write('dose onoma arxeiou
(px.c:\test.txt) :');
Readln(filename);
Assign(userfile, filename);
Reset(userfile);
Repeat
Readln(userfile,tfile);
Writeln(tfile);
Until eof(userfile);
Close(userfile);
Readln;
End.
    
```



# Παράδειγμα 3

Διαγραφή Αρχείου  
( Προσοχή Κίνδυνος !!! )

```

Program paradeigma_16;
Var ufile : text;
Filename : string;
Begin
Write('dose onoma arxeiou
(px.c:\test.txt) :');
Readln(filename);
Assign(ufile,filename);
Erase(ufile);
End.
    
```

Επεξήγηση Προγράμματος  
**Erase(UFile)**; Άνοιγμα του αρχείου  
FileName για διαγραφή.





# Παράδειγμα 4<sub>1/2</sub>

- Να γράψετε μια διαδικασία (procedure) στην Pascal που θα ζητά τις πληροφορίες: όνομα, επίθετο, email, τηλέφωνο και να τις αποθηκεύει σε αρχείο κειμένου (Append).
- Στη συνέχεια θα γίνεται ερώτηση αν θέλουμε να συνεχίσουμε να προσθέτουμε δεδομένα στο αρχείο. Αν η απάντηση είναι θετική θα ξανακαλείται η διαδικασία, ενώ αν είναι αρνητική θα τερματίζεται το πρόγραμμα.
- Τα δεδομένα να αποθηκεύονται στο αρχείο στην εξής μορφή →

*Onoma : Toto*

*Epitheto : Blabla*

*email: mymail@teiep.com*

*Telefono : 2109999999*



# Παράδειγμα 4<sub>2/2</sub>

```

Program paradeigma_17;
Procedure insert();
Var
Userfile : text;
Name, surname, mail, phone : string;
Yn : char;
Begin
Assign(userfile, 'c:\phonebook.txt');
{to arxeio c:\phonebook.txt prepei na yparxei gia na mporesei na ginei
Append !!!}
Append(userfile);
Write('onoma: ');
Readln(name);
Write('epitheto: ');
Readln(surname);
Write('email:');
Readln(mail);
Write('tilefono: ');
Readln(phone);
Writeln(userfile, 'onoma: ' + name);
Writeln(userfile, 'epitheto: ' + surname);
Writeln(userfile, 'email:' + mail);
Writeln(userfile, 'tilefono: ' + phone);
Writeln(userfile, '');
Close(userfile);
Write('thelete na sinexisete (y/n) :');
Readln(yn);
If(yn='y') then
Begin
Clrscr; {katharismos tis othonis}
Insert();
End
Else
Begin
Write('bye bye ...');
Exit; {exodos apo to programma}

```



# Αρχεία Με Τύπο<sub>1/7</sub>

- Τα **αρχεία με τύπο (typed files)** είναι αρχεία, όπου οι αριθμητικές τιμές απεικονίζονται σε αριθμητική μορφή και όχι σε μορφή χαρακτήρων, όπως στα αρχεία κειμένου.
- Οι συνιστώσες του αρχείου μπορεί να είναι κάθε τύπου, απλού, σύνθετου, με δόμηση κτλ. με εξαίρεση τον τύπο αρχείου.
- Τα αρχεία αποτελούνται από εγγραφές ενός συγκεκριμένου τύπου πχ ακεραίου, πραγματικού κτλ.



# Αρχεία Με Τύπο<sub>2/7</sub>

- Η δόμηση των αρχείων ορισμένου τύπου προσδιορίζεται από τον τύπο των δεδομένων τα οποία θα αποθηκευτούν σε αυτά.
- Για να χρησιμοποιήσουμε ένα αρχείο, πρέπει να του δώσουμε ένα όνομα και να δηλώσουμε ότι είναι **αρχείο (file)**, και να δηλώσουμε τον τύπο των δεδομένων που αποθηκεύονται σε αυτό.
- Τα δεδομένα είναι οργανωμένα σε **εγγραφές (records)** και κάθε δεδομένο αποτελεί μία εγγραφή.
- Το μήκος μίας εγγραφής είναι ο αριθμός των bytes τα οποία απαιτούνται για την αποθήκευση του συγκεκριμένου τύπου.



# Αρχεία Με Τύπο<sub>3/7</sub>

- Οι εντολές **reset**, **rewrite**, **read**, **write**, **append**, **assign**, **close**, **erase**, **rename** και **eof** που είναι διαθέσιμες από τη γλώσσα Pascal για τα αρχεία κειμένου ισχύουν με την ίδια σημασία και για τα αρχεία με τύπο.
- Οι εντολές **read** και **write** είναι απλοποιημένες, αφού μία μόνο συνιστώσα διαβάζεται ή γράφεται στο αρχείο κάθε φορά που εκτελούνται.
- Τα αρχεία με τύπο ονομάζονται και **δυναδικά (binary)** γιατί η εσωτερική απεικόνιση των δεδομένων στη μνήμη είναι ίδια με την εξωτερική απεικόνιση των δεδομένων στο αρχείο.



# Αρχεία Με Τύπο<sub>4/7</sub>

- Τα δυαδικά αρχεία διαθέτουν έναν αριθμό σειράς για κάθε συνιστώσα (εγγραφή) αρχίζοντας από το μηδέν.
- Εκτός από τη σειριακή προσπέλαση που μπορεί να γίνει με την εντολή read, η οποία κάθε φορά διαβάζει την επόμενη εγγραφή, μία τυχαία εγγραφή μπορεί να βρεθεί αν δοθεί ο αριθμός της σειράς της.
- Η άμεση ή τυχαία προσπέλαση επιτυγχάνεται με έναν ειδικό δείκτη, το **δείκτη εγγραφής αρχείου (file pointer)**.



# Αρχεία Με Τύπο<sub>5/7</sub>

- Οι ενέργειες είναι:
  1. Ενημερώνεται ο δείκτης αρχείου με τον αριθμό της εγγραφής του αρχείου που θέλουμε να διαβάσουμε ή να γράψουμε.
  2. Χρησιμοποιείται η αντίστοιχη εντολή ανάγνωσης ή εγγραφής, η οποία εκτελείται πάνω στην τρέχουσα εγγραφή.



# Αρχεία Με Τύπο<sub>6/7</sub>

- **seek (F1, R):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή assign, και R η θέση της προς αναζήτηση εγγραφής στο αρχείο.
- *Ενέργεια:* Η εντολή, η οποία είναι διαδικασία, τοποθετεί το δείκτη ανάγνωσης/εγγραφής στο record με αριθμό R για μελλοντικό read ή write.
- Η εντολή δε λειτουργεί σε αρχεία τύπου TEXT.
- Η πρώτη εγγραφή του αρχείου έχει αριθμό μηδέν (0).





# Αρχεία Με Τύπο 7/7

- **filepos (F1):** Όπου F1 είναι μεταβλητή τύπου αρχείου, όπως έχει καθοριστεί από την εντολή `assign`.
- *Ενέργεια:* Η συνάρτηση επιστρέφει τη θέση του αρχείου όπου θα διαβαστεί ή θα γραφεί η επόμενη εγγραφή.
- Στην αρχή του αρχείου η συνάρτηση επιστρέφει την τιμή μηδέν (0), στο τέλος του αρχείου επιστρέφει την ίδια τιμή με τη `filesize`.



# Παράδειγμα

```
Program paradeigma_18;
Var a,i:integer;
Var fp1: file of integer;
Begin
  Assign(fp1,'tempo.dat');
  Rewrite(fp1);
  For i:=1 to 5 do
    Begin
      Write('dose enan akeraio: ');
      Readln(a);
      Write(fp1,a);
    End;
  Close(fp1);
  Reset(fp1);
  Writeln('ta dedomena tou arxeiou einai:');
  While not eof(fp1) do
    Begin
      Read(fp1,a);
      Writeln(a);
    End;
  Writeln;
  Close(fp1);
End.
```



# Βιβλιογραφία

Βλαχάβας Ι. (1994). Η γλώσσα προγραμματισμού Pascal. Εκδόσεις Γαρταγάνης Διονύσιος.

Κάβουρας Ι.Κ. (1999). Δομημένος Προγραμματισμός με Pascal. Εκδόσεις Κλειδάριθμος.

Αλεβίζου Θ., & Καμπουρέλης Α. (1995). Μαθήματα Προγραμματισμού: Εισαγωγή με τη Γλώσσα Pascal. Εκδόσεις Παπασωτηρίου.

Cooper D. (1993). Oh! Pascal!, An Introduction to Computing, του. Εκδόσεις Norton.

Larry R.N. (1998). Advanced Programming in Pascal with Data Structures. Εκδόσεις Macmillan USA.

Τσελίκης Γ.Σ., Τσελίκας Ν.Δ. (2012). C: από τη Θεωρία στην Εφαρμογή (Β' Έκδοση). Εκδόσεις Παπασωτηρίου.

Aho A.V., Hopcroft J.E., & Ullman J.D. (1974). The design and analysis of computer algorithms. Εκδόσεις Addison Wesley.

Abelson H., Sussman G.J., Sussman J. (1985). Structure and Interpretation of Computer Programs, MIT Press, McGraw Hill Book Company.



# Σημείωμα Αναφοράς

Copyright Τεχνολογικό Ίδρυμα Ηπείρου. Αλέξανδρος Τζάλλας.  
Προγραμματισμός Ι.

Έκδοση: 1.0 Άρτα, 2015. Διαθέσιμο από τη δικτυακή  
διεύθυνση:

<http://eclass.teiep.gr/OpenClass/courses/COMP111/>



# Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά Δημιουργού-Μη Εμπορική Χρήση-Όχι Παράγωγα Έργα 4.0 Διεθνές [1] ή μεταγενέστερη. 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».



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

[1] <http://creativecommons.org/licenses/by-nc-nd/4.0/deed.el>



# Τέλος Ενότητας

Επεξεργασία: Ευάγγελος Καρβούνης  
Άρτα, 2015



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



# Τέλος Ενότητας

## Αρχεία



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

