



Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Αγρονόμων Τοπογράφων Μηχανικών

Προγραμματιστικές Τεχνικές

Βασίλειος Βεσκούκης

Δρ. Ηλεκτρολόγος Μηχανικός &
Μηχανικός Υπολογιστών ΕΜΠ
v.vescoukis@cs.ntua.gr

Ρωμύλος Κορακίτης

Αστροφυσικός
Αναπλ. Καθηγητής ΕΜΠ
romylos@survey.ntua.gr

Είσοδος / έξοδος δεδομένων

από / προς αρχεία

Αρχεία

Τα *αλληλεπιδραστικά (interactive)* προγράμματα πραγματοποιούν είσοδο δεδομένων από το πληκτρολόγιο και έξοδο προς την οθόνη. Αυτή η ροή δεδομένων απαιτεί παρουσία του χρήστη την ώρα εκτέλεσης (run time).

Η χρήση δεδομένων που βρίσκονται σε αρχεία (files), δηλαδή μόνιμα μέσα αποθηκευσης, προσφέρει σημαντικά πλεονεκτήματα:

- Τα δεδομένα μπορούν να χρησιμοποιηθούν πολλές φορές, από το ίδιο ή άλλο πρόγραμμα, χωρίς να χρειάζεται να πληκτρολογηθούν ξανά.
- Η ενημέρωση ή τροποποίηση των δεδομένων είναι πολύ εύκολη
- Σε πολλές περιπτώσεις, ένα πρόγραμμα δημιουργεί και καταγράφει μεγάλες ποσότητες δεδομένων σε αρχείο χωρίς παρέμβαση του χρήστη.

Ένα πρόγραμμα μπορεί να χρησιμοποιεί αρχεία για δύο σκοπούς:

- 1) **είσοδο δεδομένων** (input) - *αρχείο εισόδου*
- 2) **έξοδο δεδομένων** (output) - *αρχείο εξόδου*

Αρχεία κειμένου (text files)

Ανάλογα με τον τρόπο καταχώρησης των δεδομένων, τα αρχεία διακρίνονται σε:

- **Αρχεία κειμένου (text files)**, στα οποία τα δεδομένα αποτελούν μια σειρά από κωδικοποιημένους αλφαριθμητικούς χαρακτήρες (π.χ. ASCII), οργανωμένα σε εγγραφές (records). Το περιεχόμενο των αρχείων αυτών είναι κατανοητό όταν εμφανιστεί στην οθόνη και μπορεί να εκτυπωθεί.
- **Δυαδικά αρχεία (binary files)**, στα οποία τα δεδομένα καταχωρούνται με την δυαδική αναπαράστασή τους.

Θα ασχοληθούμε με αρχεία κειμένου, που ο χρήστης μπορεί να δημιουργήσει εύκολα με ένα απλό συντάκτη κειμένου (text editor), όπως ο Notepad, ο SourceEdit κ.α.

Προσοχή: σύνθετοι επεξεργαστές κειμένου (word processor), π.χ. Word, ΔΕΝ είναι κατάλληλοι για δημιουργία αρχείων κειμένου, εκτός αν ζητηθεί αποθήκευση του κειμένου σε μορφή απλού αρχείου (.txt)

Σ' ένα αρχείο κειμένου, οι εγγραφές οριοθετούνται από τον χαρακτήρα **end-of-line**, ενώ το τέλος του αρχείου από τον χαρακτήρα **end-of-file**.

Η βιβλιοθήκη `fstream`

Η διαχείριση εισόδου (input) και εξόδου (output) δεδομένων σε αρχεία γίνεται με την χρήση *ρευμάτων* (*streams*).

Γνωρίζουμε ήδη τα καθιερωμένα ρεύματα (standard streams) που περιέχονται στην βιβλιοθήκη `iostream`:

- `cin` (καθιερωμένο ρεύμα εισόδου από το πληκτρολόγιο)
- `cout` (καθιερωμένο ρεύμα εξόδου προς την οθόνη)
- `cerr` (καθιερωμένο ρεύμα εξόδου για μηνύματα σφαλμάτων στην οθόνη)

Αντίστοιχα, η βιβλιοθήκη **`fstream`** περιέχει, μεταξύ άλλων, τις εξής κλάσεις ρευμάτων:

- `ifstream` = input file stream (ρεύμα *εισόδου* από αρχείο)
- `ofstream` = output file stream (ρεύμα *εξόδου* προς αρχείο)
- `fstream` = file stream (ρεύμα αρχείου, χρήσιμο για είσοδο και έξοδο προς το ίδιο αρχείο)

Για την χρήση αρχείων εισόδου / εξόδου είναι απαραίτητη η αναφορά στην βιβλιοθήκη `fstream`:

```
#include <fstream>
```

Ρεύματα εισόδου από αρχεία

Για την είσοδο δεδομένων από ένα αρχείο:

- 1) δηλώνεται ένα ρεύμα τύπου `ifstream`, με κάποιο συγκεκριμένο όνομα
- 2) το ρεύμα αυτό συσχετίζεται με ένα αρχείο (με θέση και όνομα που ακολουθούν τους κανόνες του λειτουργικού συστήματος).
- 3) Η μεταφορά δεδομένων είναι δυνατή μόνο αφού το αρχείο έχει ανοιχτεί

Παράδειγμα:

Δήλωση ρεύματος: `ifstream temps;`

Σύνδεση με αρχείο: `temps.open("temperatures.txt", ios::in);` // ταυτόχρονο
άνοιγμα του αρχείου

Η διαδικασία μπορεί να ενοποιηθεί σε μια μόνο εντολή:

```
ifstream temps ("temperatures.txt", ios::in); // προαιρετικά
```

Η εκχώρηση τιμών, που βρίσκονται στο αρχείο, σε μεταβλητές του προγράμματος γίνεται με τρόπο αντίστοιχο με είσοδο από το πληκτρολόγιο. Η εντολή:

```
temps >> tval;
```

εκχωρεί στην μεταβλητή `tval` την τιμή που βρίσκεται στο αρχείο `temperatures.txt` μέσω του ρεύματος εισόδου `temps`.

Ρεύματα εξόδου προς αρχεία

Εντελώς αντίστοιχα με τα ρεύματα εισόδου, ορίζονται και χρησιμοποιούνται τα ρεύματα εξόδου προς αρχεία.

Παράδειγμα:

Δήλωση ρεύματος: `ofstream results;`

Σύνδεση με αρχείο: `results.open("areas.txt", ios::out);`

Η διαδικασία μπορεί να ενοποιηθεί σε μια μόνο εντολή:

```
ofstream results ("areas.txt", ios::out); // προαιρετικά
```

Προσοχή: αν το αρχείο που συνδέεται *δεν υπάρχει*, τότε δημιουργείται με την εντολή αυτή. Αν το αρχείο υπάρχει ήδη, τα περιεχόμενά του *διαγράφονται* και ετοιμάζεται να δεχθεί τις νέες εγγραφές.

Η καταγραφή τιμών μεταβλητών του προγράμματος στο αρχείο γίνεται με τρόπο αντίστοιχο με την έξοδο προς την οθόνη. Η εντολή:

```
results << "Area of triangle = " << Et << endl;
```

κατγράφει στο αρχείο `areas.txt` το αλφαριθμητικό και την τιμή της μεταβλητής `Et` μέσω του ρεύματος εξόδου `results`.

Προσάρτηση εγγραφών σε αρχείο

Σε πολλές περιπτώσεις είναι αναγκαίο να γίνουν εγγραφές σε ένα αρχείο χωρίς να διαγραφούν τα περιεχόμενα που ήδη έχει.

Για το σκοπό αυτό το άνοιγμα του αρχείου γίνεται με την μεθοδο της προσάρτησης (append mode):

```
ofstream newvals ("data.txt", ios::app); // υποχρεωτικά!
```

Προφανώς, στην περίπτωση αυτή το αρχείο πρέπει να υπάρχει ήδη. Οι νέες εγγραφές προστίθενται στο τέλος του αρχείου (μετά τις υπάρχουσες).

Συναρτήσεις που σχετίζονται με τα ρεύματα προς αρχεία

Πρίν από κάθε χρήση ρεύματος εισόδου ή εξόδου είναι χρήσιμο να γίνεται έλεγχος για την επιτυχία της σύνδεσης με το αρχείο. Το όνομα του ρεύματος (π.χ. `streamName` μπορεί να χρησιμοποιηθεί ως *λογική έκφραση* για τον έλεγχο αυτό: αν ο έλεγχος: `if (streamName)` είναι αληθής (`true`), τότε η σύνδεση με το αρχείο που έχει συσχετιστεί είναι επιτυχής.

Επιπλέον, διάφορες συναρτήσεις εξυπηρετούν ειδικές ανάγκες. Παραδείγματα:

`streamName.is_open()` - επιστρέφει τιμή `true` αν το ρεύμα είναι ανοικτό (ενεργό)

`streamName.fail()` - επιστρέφει τιμή `true` αν το ρεύμα βρεθεί σε κατάσταση σφάλματος (π.χ. ανεπιτυχής σύνδεση με αρχείο ή ανάγνωση δεδομένων διαφορετικού τύπου από τον αναμενόμενο κλπ.). Στην κατάσταση αυτή ΔΕΝ μπορεί να γίνει οποιαδήποτε μεταφορά δεδομένων από το ρεύμα μέχρι να εκτελεστεί η συνάρτηση `streamName.clear()`.

`streamName.eof()` - επιστρέφει τιμή `true` αν το ρεύμα (εισόδου) προσπαθήσει να διαβάσει μετά το τέλος του αρχείου εισόδου.

`streamName.close()` - διακόπτει την σύνδεση του ρεύματος με το αρχείο που είχε συσχετισθεί, με ταυτόχρονο κλείσιμο του αρχείου. Τώρα, το ίδιο ρεύμα μπορεί να συνδεθεί με άλλο αρχείο, με την συνάρτηση:

`streamName.open("fileName", ios::mode)`

Παραδείγματα

Στο διαδικτυακό τόπο του μαθήματος βρίσκονται μερικά αρχεία με παραδείγματα.

Το αρχείο `pt_lect2_ex1.cpp` είναι μια απλή επίδειξη των βασικών μεθόδων χρήσης αρχείων εισόδου / εξόδου.

Το αρχείο `pt_lect2_ex2.cpp` περιλαμβάνει μια πιο σύνθετη εφαρμογή διαχείρισης γεωμετρικών οντοτήτων, που χρειάζεται το αρχείο εισόδου `quad_vertex.txt` (βρίσκεται επίσης στον διαδικτυακό τόπο). Το αρχείο αυτό περιέχει τις ορθογώνιες συντεταγμένες (x_i, y_i) των 4 κορυφών ενός πλήθους τετραπλεύρων (8 συντεταγμένες ανά γραμμή)

Η εφαρμογή διαβάζει τα δεδομένα, υπολογίζει τα μήκη των πλευρών, την περίμετρο και το εμβαδόν κάθε τετραπλεύρου και, τέλος, γράφει όλα τα αποτελέσματα στο αρχείο `quad_results.txt`

Παράδειγμα 2

