



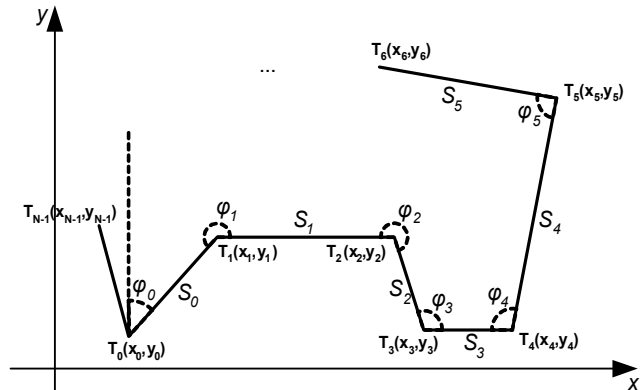
ΑΣΚΗΣΕΙΣ ΓΙΑ ΜΕΛΕΤΗ ΚΑΙ ΕΚΠΟΝΗΣΗ ΣΤΟ ΕΡΓΑΣΤΗΡΙΟ

ΑΣΚΗΣΗ 1

(Προαιρετική επίδειξη στους υπεύθυνους των εργαστηρίων μέχρι και την Πέμπτη 9.3.2006)

Προκειμένου να εκτελεστούν έργα τοπογραφικής αποτύπωσης, δύο ομάδες που λειτουργούν ανεξάρτητα λαμβάνουν μετρήσεις όδευσης περιοχών που ορίζονται από N σημεία T_0 έως T_{N-1} . Το σημείο αρχής λήψης των μετρήσεων (T_0) είναι το ίδιο και για τις δύο ομάδες και έχει συντεταγμένες $T_0(x_0, y_0)$. Από κάθε ομάδα λαμβάνονται N ζεύγη μετρήσεων (s_i, φ_i) όπου: s_i η απόσταση προς το επόμενο σημείο, φ_i είναι η γωνία θάλασης σε βαθμούς, $i=0..N-1$, και N είναι το πλήθος των σημείων που ορίζουν την περιοχή, όπως στο σχήμα.

Μετά την ολοκλήρωση των μετρήσεων οι τιμές που μετρήθηκαν από κάθε ομάδα καταχωρούνται σε Η/Υ με χρήση προγράμματος C++ προκειμένου να κανονικοποιηθούν και να γίνει ο υπολογισμός των συντεταγμένων. Οι αρχικές μετρήσεις κανονικοποιούνται ως ακολούθως:



- Αν οι δύο μετρήσεις του ίδιου μήκους διαφέρουν μεταξύ τους μέχρι 1% λαμβάνεται ως σωστή η μικρότερη από αυτές. Αν διαφέρουν περισσότερο, λαμβάνεται ο μέσος όρος τους.
- Για τις μετρήσεις της ίδιας γωνίας που διαφέρουν μεταξύ τους, λαμβάνεται πάντα ο μέσος όρος.

Μετά από την κανονικοποίηση των αρχικών μετρήσεων, προκύπτει ένα νέο σύνολο μετρήσεων που αποτελείται από N ζεύγη τιμών (S_i, Φ_i), που έχουν υπολογιστεί με τον παραπάνω τρόπο. Οι τιμές αυτές χρησιμοποιούνται στον υπολογισμό της όδευσης, δηλαδή των συντεταγμένων των σημείων $T1(x_1, y_1), T2(x_2, y_2), \dots, TN-1(x_{N-1}, y_{N-1})$. Υπενθυμίζεται ότι ο υπολογισμός αυτός γίνεται σύμφωνα με τους τύπους:

$$a_i = a_{i-1} + \varphi_i - \pi, \quad a_0 = \varphi_0$$
$$x_i = x_{i-1} + s_{i-1} \cdot \sin(a_{i-1}) \quad i = 1..N-1$$
$$y_i = y_{i-1} + s_{i-1} \cdot \cos(a_{i-1})$$

1. Να καθορίσετε τα δεδομένα εισόδου και εξόδου του προβλήματος
2. Να καταστρώσετε έναν αλγόριθμο για την επίλυση του προβλήματος
3. Να κατασκευάσετε το αντίστοιχο πρόγραμμα σε C++, χρησιμοποιώντας συναρτήσεις
 - για να διαβάσετε τα δεδομένα εισόδου
 - για να μετατρέψετε βαθμούς σε ακτίνια
 - για να υπολογίσετε τις κανονικοποιημένες μετρήσεις
 - για να κάνετε τους υπολογισμούς της όδευσης
 - για να εκτυπώσετε τα αποτελέσματα

ΑΣΚΗΣΗ 2. Συνάρτηση για τον υπολογισμό μήκους ευθύγραμμου τμήματος

(Μελέτη μέχρι την Πέμπτη 16.3.2006)

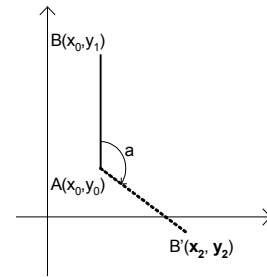
Να κατασκευάσετε τη συνάρτηση `LineLen(float x1, float y1, float x2, float y2)` η οποία θα δέχεται ως ορίσματα τις καρτεσιανές συντεταγμένες δύο σημείων $A(x_1, y_1)$ και $B(x_2, y_2)$, θα υπολογίζει το μήκος του ευθύγραμμου τμήματος AB και θα το επιστρέφει (ως πραγματικό αριθμό απλής ακρίβειας) στην καλούσα συνάρτηση. Ο υπολογισμός του μήκους να γίνεται με χρήση του Πυθαγορείου θεωρήματος, εκτός από την περίπτωση που το ευθύγραμμο τμήμα AB είναι παράλληλο με έναν από τους καρτεσιανούς άξονες, οπότε ο υπολογισμός του μήκους να γίνεται με μια απλή αφαίρεση.

Να επιδείξετε τη χρήση της `LineLen` καλώντας την από ένα κύριο πρόγραμμα το οποίο διαβάζει από το πληκτρολόγιο τις συντεταγμένες δύο σημείων, καλεί τη συνάρτηση `LineLen` με αυτές ως παραμέτρους και τυπώνει το αποτέλεσμα που επιστρέφει η συνάρτηση. Η διαδικασία "ανάγνωση συντεταγμένων - κλήση συνάρτησης - εκτύπωση μήκους" να επαναλαμβάνεται μέχρις ότου δοθούν σημεία που ταυτίζονται.

ΑΣΚΗΣΗ 3. Συνάρτηση για την περιστροφή ευθύγραμμου τμήματος κατά γωνία a

(Μελέτη μέχρι την Πέμπτη 16.3.2006)

Δίνεται ευθύγραμμο τμήμα AB, παράλληλο στον άξονα yy' με συντεταγμένες A(x₀, y₀) και B(x₁, y₁), όπου x₀=x₁ (όπως στο σχήμα). Κατασκευάστε τη συνάρτηση `rotateAB` σε C++, η οποία υπολογίζει τις καρτεσιανές συντεταγμένες του άκρου B'(x₂, y₂) του ευθύγραμμου τμήματος AB όταν αυτό υποστεί δεξιόστροφη περιστροφή κατά γωνία a (εκπεφρασμένη σε ακτίνια) όπως στο σχήμα.



Η συνάρτηση θα δέχεται ως παραμέτρους τις τιμές x₀, y₀, y₁ και a (σε μονάδες μέτρησης μήκους και ακτίνια) και θα μεταβάλλει τις τιμές των κατ' αναφορά παραμέτρων x₂ και y₂.

Δηλαδή, η επικεφαλίδα ορισμού της είναι η ακόλουθη:

```
void rotateAB(float x0, float y0, float y1, float angle, float& x2, float& y2)
```

Μπορείτε να χρησιμοποιήσετε τους τύπους:

$x_2 = x_0 + s \cdot \sin(a)$, $y_2 = y_0 + s \cdot \cos(a)$ όπου s είναι το μήκος του AB το οποίο να υπολογίζεται με κλήση της συνάρτησης `LineLen(x0, y0, x0, y1)`. Οι τριγωνομετρικές συναρτήσεις `cos` και `sin` περιέχονται στη βιβλιοθήκη `cmath` της C++.

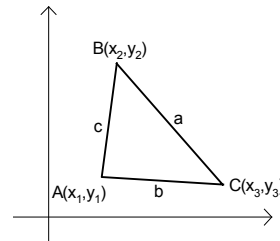
Να επιδείξετε τη χρήση της `rotateAB` καλώντας την από ένα κύριο πρόγραμμα το οποίο διαβάζει από το πληκτρολόγιο τις συντεταγμένες x₀, y₀, y₁ και τη γωνία a, και καλεί τη συνάρτηση `rotateAB` με παραμέτρους τις τιμές των x₀, y₀, y₁, a, καθώς και τις (κατ' αναφορά) μεταβλητές x₂ και y₂, και τυπώνει τις τιμές x₂ και y₂ που μεταβάλλει η συνάρτηση. Η διαδικασία "ανάγνωση δεδομένων - κλήση συνάρτησης - εκτύπωση τιμών x₂ και y₂" να επαναλαμβάνεται μέχρις ότου δοθεί μηδενική γωνία περιστροφής a.

ΑΣΚΗΣΗ 4. Συνάρτηση για τον υπολογισμό εμβαδού τριγώνου με τον τύπο του Ήρωνα

(Μελέτη μέχρι την Πέμπτη 16.3.2006)

Δίνονται οι συντεταγμένες των κορυφών τριγώνου ABC, όπως στο διπλανό σχήμα. Να κατασκευάσετε τη συνάρτηση `tarea` η οποία θα δέχεται ως παραμέτρους τις τιμές των συντεταγμένων x₁, y₁, κλπ και θα επιστρέφει στην καλούσα συνάρτηση (ως πραγματικό αριθμό απλής ακρίβειας) το εμβαδόν του τριγώνου ABC, το οποίο θα υπολογίζει χρησιμοποιώντας τον τύπο του Ήρωνα:

$$A = \sqrt{\tau \cdot (\tau - a) \cdot (\tau - b) \cdot (\tau - c)}, \quad \tau = \frac{a + b + c}{2}$$



a, b και c είναι τα μήκη των πλευρών, τα οποία πρέπει να υπολογίσετε ως μήκη των ευθυγράμμων τμημάτων BC, AC και AB αντίστοιχα, καλώντας υποχρεωτικά τη συνάρτηση `LineLen` που έχετε κατασκευάσει στην άσκηση 2.

Να επιδείξετε τη χρήση της `tarea` καλώντας την από ένα κύριο πρόγραμμα το οποίο διαβάζει από το πληκτρολόγιο τις συντεταγμένες x₁, y₁, x₂, y₂, x₃, y₃, καλεί τη συνάρτηση `tarea` με αυτές ως παραμέτρους και τυπώνει το αποτέλεσμα που επιστρέφει η συνάρτηση. Η διαδικασία "ανάγνωση συντεταγμένων - κλήση συνάρτησης - εκτύπωση εμβαδού" θα πρέπει να επαναλαμβάνεται μέχρις ότου το εμβαδόν να είναι μηδέν.

ΑΣΚΗΣΗ 5. Συνάρτηση για τη μετατροπή βαθμών σε ακτίνια

(Μελέτη μέχρι την Πέμπτη 16.3.2006)

Να κατασκευάσετε τη συνάρτηση `float grad_to_rads(float g)` η οποία μετατρέπει γωνίες εκπεφρασμένες σε βαθμούς (δίνονται ως όρισμα g της συνάρτησης), σε ακτίνια (αποτέλεσμα που επιστρέφει η συνάρτηση). Να επιδείξετε τη χρήση της καλώντας τη από ένα κύριο πρόγραμμα.

ΑΣΚΗΣΗ 6. Συνάρτηση για τη μετατροπή ακτίνων σε μοίρες

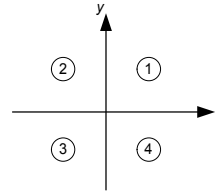
(Μελέτη μέχρι την Πέμπτη 16.3.2006)

Να κατασκευάσετε τη συνάρτηση `void rads_to_deg(float rads, float& deg, float& min, float& sec)` η οποία μετατρέπει γωνίες εκπεφρασμένες σε ακτίνια (όρισμα rads της συνάρτησης), σε εξηκονταδικές μοίρες (ορίσματα deg, min, sec, που μεταβάλλει η συνάρτηση). Να επιδείξετε τη χρήση της καλώντας τη από ένα κύριο πρόγραμμα.

ΑΣΚΗΣΗ 7. Συνάρτηση ελέγχου θέσης σημείου

(Μελέτη μέχρι την Πέμπτη 16.3.2006)

Θεωρώντας ότι τα τεταρτημόρια του καρτεσιανού επιπέδου αριθμούνται όπως στο διπλανό σχήμα, να κατασκευάσετε τη συνάρτηση `sector` η οποία δέχεται ως παραμέτρους τις συντεταγμένες x και y ενός σημείου (πραγματικοί αριθμοί) και επιστρέφει έναν χαρακτήρα ως εξής:



- 1 αν το σημείο ανήκει στο τεταρτημόριο 1
- 2 αν ανήκει στο 2ο τεταρτημόριο
- 3 αν ανήκει στο 3ο τεταρτημόριο
- 4 αν ανήκει στο 4ο τεταρτημόριο

X (κεφαλαίο x) αν ανήκει στα θετικά του άξονα xx'
 x (πεζό x) αν ανήκει στα αρνητικά του άξονα xx'
Y (κεφαλαίο y) αν ανήκει στα θετικά του άξονα yy'
 y (πεζό y) αν ανήκει στα αρνητικά του άξονα yy'
0 (μηδέν) αν πρόκειται για την αρχή των αξόνων

ΥΠΟΔΕΙΞΕΙΣ ΑΣΚΗΣΕΩΝ 2-7

1. Για κάθε άσκηση καλό είναι να καταστρώσετε τον αλγόριθμο που θα ακολουθήσετε.
2. Καμία από τις συναρτήσεις που καλείστε να κατασκευάσετε δεν θα πρέπει να πραγματοποιεί είσοδο δεδομένων από το πληκτρολόγιο ή έξοδο προς την οθόνη. Είσοδος και έξοδος θα πρέπει να γίνεται μόνο από το κύριο πρόγραμμα που καλεί τις συναρτήσεις.
3. Τοποθετείστε όλες τις συναρτήσεις που κατασκευάσατε στο αρχείο **"myfunctions.cpp"** και κάντε το **#include** στα αρχεία που περιέχουν τα κύρια προγράμματα που τις χρησιμοποιούν.

ΑΣΚΗΣΗ 8. Εμβέλεια μεταβλητών

(Μελέτη μέχρι την Πέμπτη 16.3.2006)

Να τρέξετε το διπλανό πρόγραμμα με το χέρι και να καταγράψετε τι περιμένετε να εκτυπώνει.

Στη συνέχεια να το πληκτρολογήσετε και να το τρέξετε στον υπολογιστή.

Αν παρατηρήσετε διαφορά μεταξύ της αναμενόμενης και της πραγματικής εξόδου του προγράμματος, να προσπαθήσετε να την εξηγήσετε.

Μην κάνετε "ζαβολιές"!

```
#include <iostream>
using namespace std;

int a,b,c,d;

void f(int a, int& b) {
    int c;
    b=a+b; a=d+3; c=a+1;
    cout<<a<<" "<<b<<" "<<c<<" "<<d<<endl; }

int main() {
    a=4; b=3; c=2; d=1;
    cout<<a<<" "<<b<<" "<<c<<" "<<d<<endl;
    f(b,c);
    cout<<a<<" "<<b<<" "<<c<<" "<<d<<endl;
    f(a,d);
    cout<<a<<" "<<b<<" "<<c<<" "<<d<<endl; }
```

ΑΣΚΗΣΕΙΣ 9-19. Ασκήσεις του βιβλίου

(Μελέτη μέχρι την Πέμπτη 16.3.2006)

Διαβάστε τις παραγράφους 3.1-3.9 και ασχοληθείτε με τις ακόλουθες ασκήσεις:

3.16, 3.17, 3.18, 3.20, 3.21, 3.22, 3.23, 3.26, 3.35, 3.41, και 3.53.

ΥΠΟΔΕΙΞΗ ΑΣΚΗΣΕΩΝ 20-24

Για τις ασκήσεις που ακολουθούν θα χρειαστείτε το αρχείο **myfunctions.cpp** που αναφέρεται στην προηγούμενη υπόδειξη. Το αρχείο θα χρησιμοποιείται με τη δήλωση **#include "myfunctions.cpp"** και θα πρέπει να περιέχει τουλάχιστον τις παρακάτω συναρτήσεις:

`float LineLen(float x1, float y1, float x2, float y2)`: υπολογισμός μήκους

`float tarea(float x1, float y1, float x2, float y2, float x3, float y3)`: εμβαδό τριγώνου

`float grad_to_rads(float g)`: βαθμοί σε ακτίνια

`void rads_to_deg(float rads, float& deg, float& min, float& sec)`: ακτίνια σε μοίρες

ΑΣΚΗΣΗ 20. Εγγραφή τιμών σε αρχείο κειμένου

(Μελέτη μέχρι την Πέμπτη 30.3.2006)

Κατασκευάστε τη συνάρτηση `savegrads()` η οποία διαβάζει από το πληκτρολόγιο μετρήσεις γωνιών σε βαθμούς και τις αποθηκεύει στο αρχείο `"grads.txt"`. Σε κάθε γραμμή του αρχείου να γράφεται μόνο μία μέτρηση. Η ανάγνωση και η αποθήκευση των τιμών να επαναλαμβάνεται μέχρις ότου ο χρήστης δώσει την τιμή μηδέν η οποία να μην αποθηκεύεται στο αρχείο. Σε περίπτωση που η εγγραφή στο αρχείο είναι επιτυχής η συνάρτηση `savegrads()` πρέπει να επιστρέφει την τιμή 0, διαφορετικά να επιστρέφει την τιμή -1.

Ακολουθώντας, κατασκευάστε ένα κύριο πρόγραμμα που καλεί τη συνάρτηση `savegrads()` για να την χρησιμοποιήσετε. Ανοίξτε το αρχείο `measurements1.txt` για να επιβεβαιώσετε τη λειτουργία του προγράμματός σας. Να χρησιμοποιήσετε τις συναρτήσεις της βιβλιοθήκης `fstream`.

ΑΣΚΗΣΗ 21. Ανάγνωση από αρχείο κειμένου

(Μελέτη μέχρι την Πέμπτη 16.3.2006)

Κατασκευάστε τη συνάρτηση `processgrads()`, η οποία να ανοίγει το αρχείο `grads.txt` που κατασκευάσατε στην άσκηση 20, να διαβάζει τις μετρήσεις σε βαθμούς, να μετατρέπει κάθε μέτρηση σε ακτίνια και σε μοίρες, και να εμφανίζει στην οθόνη την τιμή σε βαθμούς, ακτίνια και μοίρες, μορφοποιημένη όπως στο παρακάτω παράδειγμα. Θα χρειαστεί να καλέσετε τις συναρτήσεις της βιβλιοθήκης `myfunctions.cpp`.

Περιεχόμενα αρχείου <code>grads.txt</code>	Εκτύπωση προγράμματος στην οθόνη		
200	grad	rad	deg
100	200	3.14	180 0' 0"
50	100	1.57	90 0' 0"
	50	0.78	45 0' 0"

Τροποποιήστε το πρόγραμμά σας ώστε εκτός από την οθόνη, τα ίδια αποτελέσματα να γράφονται στο αρχείο `angles.txt`.

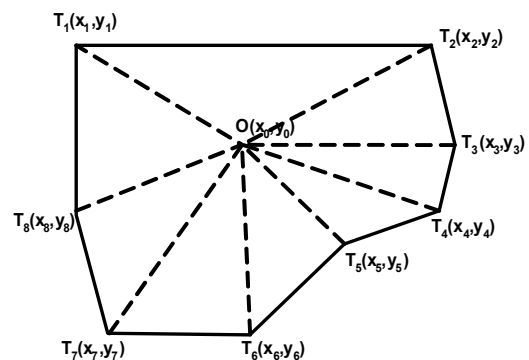
ΑΣΚΗΣΗ 22. Υπολογισμός εμβαδού πολυγώνου

(Μελέτη μέχρι την Πέμπτη 16.3.2006)

Σε ένα αρχείο `"polygon.txt"` βρίσκονται αποθηκευμένες οι συντεταγμένες ενός εσωτερικού σημείου και των N κορυφών ενός πολυγώνου ως εξής:

1. Στην πρώτη γραμμή βρίσκονται οι συντεταγμένες x_0, y_0 ενός σημείου O μέσα στο πολύγωνο
2. Σε κάθε μία από τις επόμενες γραμμές του αρχείου βρίσκονται οι συντεταγμένες x_k, y_k της k -κορυφής του πολυγώνου T_k . Το πλήθος των κορυφών δεν είναι εκ των προτέρων γνωστό και καθορίζεται από το πλήθος των γραμμών του αρχείου που έπονται της πρώτης.

Το εμβαδόν του πολυγώνου μπορεί να υπολογιστεί ως άθροισμα των εμβαδών των τριγώνων $OT_1T_2, OT_2T_3, OT_3T_4, \dots$ κλπ που σχηματίζονται, όπως στο διπλανό σχήμα.



Για παράδειγμα, το αρχείο `polygon.txt` μπορεί να περιέχει τα παρακάτω στοιχεία:

```
1.0      1.0
3.0      2.0
2.0      -2.0
-2.0     -2.0
-3.0     2.0
```

Κατασκευάστε τη συνάρτηση `parea()` η οποία ανοίγει το αρχείο `polygon.txt`, διαβάζει τα περιεχόμενά του και επιστρέφει το εμβαδόν του πολυγώνου. Σε περίπτωση που το αρχείο δεν μπορεί να ανοίξει ή να διαβαστεί, η συνάρτηση να επιστρέφει -1. Για τον υπολογισμό των εμβαδών των τριγώνων να χρησιμοποιήσετε τη συνάρτηση `tarea()` από τη βιβλιοθήκη `myfunctions.cpp`. Να ελέγξετε και να επιδείξετε τη λειτουργία της συνάρτησης καλώντας την μέσα από ένα κύριο πρόγραμμα το οποίο εμφανίζει στην οθόνη την τιμή του εμβαδού που υπολογίζει η συνάρτηση. Για διευκόλυνσή σας, μπορείτε να κάνετε χρήση του αρχείου `polygon.txt` που μπορείτε να κατεβάσετε από το site του μαθήματος.

Σημείωση: Το πρόγραμμά σας **δεν ζητείται** να ελέγχει αν το σημείο O που περιέχεται στο αρχείο είναι πράγματι εντός του πολυγώνου. Μπορείτε να κάνετε τον έλεγχο προαιρετικά!

ΑΣΚΗΣΗ 23. Εμβαδό πολυγώνου (2)

(Μελέτη μέχρι την Πέμπτη 30.3.2006)

A. Τροποποιήστε το πρόγραμμά της άσκησης 22 ώστε πρώτα να αποθηκεύει τις συντεταγμένες των κορυφών του πολυγώνου σε δύο μονοδιάστατους πίνακες πραγματικών αριθμών X και Y και να κάνει τον υπολογισμό του εμβαδού του πολυγώνου διατρέχοντας τα στοιχεία των πινάκων.

B. Δώστε δεδομένα ελέγχου για το πρόγραμμά σας, δηλαδή τιμές σε συντεταγμένες X και Y οι οποίες να ορίζουν πολύγωνο με εμβαδόν που υπολογίζεται χωρίς υπολογιστή χειρός (calculator). Στη συνέχεια, τρέξτε το πρόγραμμά σας και συγκρίνετε το αποτέλεσμα του με το εκ των προτέρων γνωστό σε σας εμβαδόν.

ΑΣΚΗΣΗ 24. Ανάγνωση σήματος

(Μελέτη μέχρι την Πέμπτη 30.3.2006)

Σήμα είναι μια αλληλουχία τιμών που παριστάνουν ένα μετρήσιμο φυσικό φαινόμενο. Θεωρήστε ότι στο αρχείο signal1D.txt περιέχεται άγνωστος αριθμός ακέραιων τιμών που αποτελούν ένα σήμα. Κατασκευάστε τη συνάρτηση getsignal() που ανοίγει το αρχείο signal1D.txt και τοποθετεί τις τιμές του σήματος σε ένα πίνακα ακεραίων, ο οποίος έχει προκαθορισμένη μέγιστη διάσταση. Αν το πλήθος των τιμών του αρχείου είναι μεγαλύτερο από την προκαθορισμένη μέγιστη διάσταση, η συνάρτηση getsignal() να επιστρέφει την τιμή -1, διαφορετικά να επιστρέφει το πλήθος των τιμών του σήματος.

ΑΣΚΗΣΗ 25. Δομή struct για τον ορισμό πολυγώνου

(Μελέτη μέχρι την Πέμπτη 4.5.2006)

Στην άσκηση 22 ασχοληθήκαμε με τον υπολογισμό του εμβαδού ενός πολυγώνου τις συντεταγμένες των κορυφών του οποίου διαβάζαμε από αρχείο. Μια καλύτερη ιδέα είναι να ορίσουμε το struct `polygon` που αντιστοιχεί σε πολύγωνο και να αποθηκεύουμε τις συντεταγμένες που διαβάζουμε από το αρχείο σε μεταβλητές τύπου `polygon`. Το struct αυτό πρέπει να περιέχει πεδία:

- α) για το πλήθος των κορυφών του πολυγώνου (ακέραιος αριθμός που δεν μπορεί να είναι μεγαλύτερος από έναν αριθμό MAXK που δηλώνεται ως σταθερά στην αρχή του προγράμματος).
- β) για τις συντεταγμένες x και y των κορυφών του πολυγώνου (δύο μονοδιάστατοι πίνακες πραγματικών αριθμών με διάσταση MAXK)
- γ) το εμβαδό και την περίμετρο του πολυγώνου (πραγματικοί αριθμοί)
- δ) τις συντεταγμένες ενός εσωτερικού σημείου του πολυγώνου (πραγματικοί αριθμοί)

Το μέγιστο πλήθος των κορυφών MAXK να είναι 100 και να ορίζεται με σταθερά (const) ώστε να μπορεί να μεταβληθεί εύκολα. Τι έχετε να παρατηρήσετε για την ανεξαρτησία των δεδομένων που περιλαμβάνονται στο struct polygon;

ΑΣΚΗΣΗ 26. Ανάγνωση συντεταγμένων πολυγώνου από αρχείο κειμένου

(Μελέτη μέχρι την Πέμπτη 4.5.2006)

Κατασκευάστε τη συνάρτηση `init_polygon(polygon p)` η οποία δέχεται ως όρισμα μια μεταβλητή p τύπου `polygon` και διαβάζει από το αρχείο `polygon.txt` τις συντεταγμένες του εσωτερικού σημείου και των κορυφών του πολυγώνου, τις οποίες και τοποθετεί στα αντίστοιχα πεδία του struct p. Προσέξτε τη σωστή ενημέρωση του πεδίου που αντιστοιχεί στο πλήθος των κορυφών του πολυγώνου. Να κάνετε την παραδοχή ότι στο αρχείο `polygon.txt` δεν περιέχονται περισσότερες από MAXK κορυφές. Στη συνέχεια κατασκευάστε τη συνάρτηση `polygon_data(polygon1 p)` η οποία δέχεται επίσης ως όρισμα μια μεταβλητή p τύπου `polygon` και εμφανίζει στην οθόνη τις συντεταγμένες των κορυφών του πολυγώνου.

Υπόδειξη: Χρησιμοποιήστε ως αφετηρία τη συνάρτηση `parea()` της άσκησης 22 και τροποποιήστε τη ώστε να μην πραγματοποιεί υπολογισμό εμβαδού και μόνο να αποθηκεύει σε μεταβλητή μνήμης τύπου struct p τα δεδομένα που διαβάζει από το αρχείο.

Προαιρετικό: τροποποιήστε τη συνάρτηση `init_polygon` ώστε να δέχεται ένα επιπλέον όρισμα που αντιστοιχεί στο όνομα του αρχείου απ' όπου θα διαβαστούν τα στοιχεία του πολυγώνου.

ΑΣΚΗΣΗ 27. Ορισμός της κλάσης "πολύγωνο"

(Μελέτη μέχρι την Πέμπτη 18.5.2006)

Δώστε τον πλήρη ορισμό μιας κλάσης `polygonclass` η οποία αντιστοιχεί στο πολύγωνο και επεκτείνει το struct που ορίσατε στην άσκηση 25. Εκτός από τα πεδία που αναφέρονται στην 25, θα πρέπει να ορίσετε μεθόδους για τη δημιουργία ενός πολυγώνου με ανάγνωση από αρχείο (constructor), για εκτύπωση των συντεταγμένων των κορυφών, για τον υπολογισμό του εμβαδού, καθώς και για τον υπολογισμό της περιμέτρου. Να κατασκευάσετε ένα κύριο πρόγραμμα που δημιουργεί το αντικείμενο `myP` του τύπου `polygon`, το αρχικοποιεί, υπολογίζει εμβαδό και περίμετρο, και τα εκτυπώνει στην οθόνη. Για την υλοποίηση των μεθόδων της κλάσης αρκεί να τροποποιήσετε τις συναρτήσεις που κατασκευάσατε στην άσκηση 26 και στην άσκηση 22.

ΑΣΚΗΣΗ 28. Επέκταση-βελτίωση της κλάσης "πολύγωνο"

(Μελέτη μέχρι το τέλος του εξαμήνου)

A. Να δηλώσετε όλα τα πεδία της κλάσης `polygonclass` ως `private` και να ορίσετε μεθόδους για την ορατότητά τους από έξω. Τροποποιήστε ανάλογα το κύριο πρόγραμμα που κατασκευάσατε στην άσκηση 3.4 ώστε να επιδείξετε τη λειτουργία της ορατότητας.

B. Να προσθέσετε στην κλάση `polygonclass` μια μέθοδο `move_polygon(float Dx, float Dy)` που μεταφέρει το πολύγωνο στο επίπεδο κατά `Dx` και `Dy` και να την επιδείξετε με κατάλληλη τροποποίηση του κυρίως προγράμματος.

Γ. Να προσθέσετε στην κλάση `polygonclass` μια μέθοδο `move_koryfh(int koryfh, float Dx, float Dy)` που μετακινεί μόνο την κορυφή `koryfh` κατά `Dx` και `Dy`, και πραγματοποιεί όλους τους διορθωτικούς υπολογισμούς. Να την επιδείξετε με κατάλληλη τροποποίηση του κυρίως προγράμματος.

ΑΣΚΗΣΕΙΣ 29-32. Ασκήσεις από το βιβλίο

(Μελέτη μέχρι το τέλος του εξαμήνου)

Παράγραφος 6.5 (παράδειγμα), 6.7, 6.12, 6.13