

Προγραμματισμός II
2η Άσκηση

Διδάσκων:
Χ. Τρυφωνόπουλος

Παράδοση μέχρι Τρίτη 22/05/2012 ώρα 23.59
Εξέταση Πέμπτη 24/05/2012

ΣΗΜΑΝΤΙΚΕΣ ΣΗΜΕΙΩΣΕΙΣ:

1. Στο αρχείο που γράφετε τον κώδικα για κάθε άσκηση πρέπει **ΟΠΩΣΔΗΠΟΤΕ** να βάλετε σε σχόλια τα ονόματα, τους Α.Μ., και τα username/email των μελών της ομάδας (ομάδες **αυστηρά 2 ατόμων**). Όλα τα σχόλια και τα μηνύματα του προγράμματός σας πρέπει να είναι με λατινικούς χαρακτήρες (και **όχι στα ελληνικά**). Ακολουθεί παράδειγμα:

/* Nikos Papadopoulos

AM: 2888

cst2888@uop.gr

Christos Tryfonopoulos

AM: 2929

cst02929@uop.gr

*/

2. Αφού έχετε ολοκληρώσει την άσκηση που θέλετε να παραδώσετε την υποβάλετε στο eclass στο υποσύστημα «Εργασίες φοιτητών». Η υποβολή πρέπει να γίνει **ΠΡΙΝ** την ημερομηνία παράδοσης. Παραδίδετε **ΜΟΝΟ** τα αρχεία με τον κώδικα (με κατάληξη .c και .h) σε ένα συμπιεσμένο αρχείο και **ΟΧΙ** τα εκτελέσιμα μετά την μεταγλώττιση. **Προσοχή:** τα προγράμματα που θα παραδώσετε θα πρέπει να κάνουν compile και να τρέχουν στα μηχανήματα UNIX του τμήματος. Ασκήσεις οι οποίες δεν κάνουν compile ή δεν τρέχουν στα μηχανήματα UNIX του τμήματος **θα μηδενίζονται**.
3. Περιπτώσεις αντιγραφής **θα μηδενίζονται** και οι εμπλεκόμενοι **δεν θα έχουν δικαίωμα παράδοσης άλλων ασκήσεων**. Η ημερομηνία παράδοσης είναι αυστηρή, και η παράδοση γίνεται μόνο μέσω του eclass και όχι με email στον διδάσκοντα ή στην βοηθό του μαθήματος. Ασκήσεις που παραδίδονται μετά τη λήξη της προθεσμίας **δε γίνονται δεκτές**.

ΕΚΦΩΝΗΣΗ ΑΣΚΗΣΗΣ

Στην άσκηση αυτή καλείστε να κατασκευάσετε ένα πρόγραμμα που θα επιτρέπει το **διαδραστικό παίξιμο** του παιχνιδιού ντάμα (draughts στα αγγλικά). Η

αμερικάνικη έκδοσή του λέγεται checkers κι έχει ελαφρώς διαφορετικούς κανόνες.

Περίληπτικά οι κανόνες του παιχνιδιού είναι οι παρακάτω. Η ντάμα είναι ένα απλό παιχνίδι στρατηγικής που παίζεται σε ένα ταμπλό 8x8 από 2 παίκτες, όπου κάθε παίκτης ελέγχει 12 πιόνια. Τα πιόνια κινούνται μόνο διαγώνια μπροστά σε κενό τετράγωνο. Ένα πιόνι «τρώει» ένα αντίπαλο πιόνι όταν βρίσκονται σε δύο γειτονικά διαγώνια τετράγωνα κι αν ακριβώς πίσω διαγώνια από το αντίπαλο πιόνι υπάρχει κενό τετράγωνο. Το «φάγωμα» αντίπαλων πιονιών μπορεί να συμβεί για περισσότερα από ένα αντίπαλα πιόνια στην ίδια κίνηση, ακόμα και αν αυτά δεν είναι στην ευθεία, με την προϋπόθεση ότι το πιόνι που τα τρώει προχωρά μόνο προς τα διαγώνια εμπρός. Το φάγωμα ενός πιονιού είναι υποχρεωτικό, δηλαδή ο παίκτης δεν πρέπει να εκτελέσει άλλη κίνηση αν μπορεί να «φάει» ένα αντίπαλο πιόνι. Αν ένα πιόνι φτάσει στην απέναντι άκρη του ταμπλό γίνεται βασιλιάς. Αυτό σημαίνει πως το συγκεκριμένο πιόνι μπορεί να κινείται και διαγώνια πίσω ένα τετράγωνο τη φορά. Όταν ένας παίκτης δεν έχει κινήσεις, χάνει το παιχνίδι, το οποίο κερδίζει ο αντίπαλός του. Ένας παίκτης μπορεί να ξεμείνει από κινήσεις είτε αν χάσει όλα του τα πιόνια, είτε αν ο αντίπαλός του τα μπλοκάρει. Περισσότερες λεπτομέρειες για τους κανόνες του παιχνιδιού δείτε στην σελίδα της Wikipedia <http://en.wikipedia.org/wiki/English draughts>. Το πρόγραμμα που καλείστε να υλοποιήσετε θα πρέπει να έχει τη λειτουργικότητα που περιγράφεται παρακάτω.

Εκτέλεση από τη γραμμή εντολών

Ο χρήστης θα μπορεί να τρέξει το πρόγραμμα από τη γραμμή εντολών με τους τρόπους που περιγράφονται παρακάτω. Λύσεις που δεν θα υλοποιούν αυτή τη λειτουργικότητα θα έχουν **ποινή -10%**.

myCheckers -n	τρέχει το πρόγραμμα και επιτρέπει στον χρήστη να ξεκινήσει να παίξει ένα καινούριο παιχνίδι ντάμας
myCheckers -l christos.chk	τρέχει το πρόγραμμα και επιτρέπει στον χρήστη να φορτώσει και να συνεχίσει το παιχνίδι μιας αποθηκευμένης παρτίδας από ένα αρχείο που δίνεται από τον χρήστη (στο συγκεκριμένο παράδειγμα το christos.chk)
myCheckers -h	τυπώνει βοήθεια στον χρήστη που του εξηγεί με τι παραμέτρους μπορεί να τρέξει το πρόγραμμα

Εκκίνηση και παίξιμο του παιχνιδιού

Αν ο χρήστης επέλεξε να παίξει ένα καινούριο παιχνίδι ντάμας (τρέχοντας το παιχνίδι με την παράμετρο -n) τότε το προγράμματά σας θα πρέπει να κάνει τα εξής:

1. Θα πρέπει να δημιουργεί ένα καινούριο ταμπλό παιχνιδιού αρχικοποιώντας έναν **πίνακα διαστάσεων 8x8** όπου επάνω του θα τοποθετούνται τα πιόνια των δύο παικτών στις αρχικές τους θέσεις. Οι γραμμές του πίνακα θα είναι αριθμημένες από το 1-8 και οι στήλες του από το a-h, όπως φαίνεται παρακάτω. Για τα δύο είδη πιονιών μπορείτε να χρησιμοποιείτε όποιον χαρακτήρα θέλετε (στο παράδειγμα χρησιμοποιούνται τα O και X), ενώ για το κενό τετράγωνο χρησιμοποιείτε την τελεία. Υλοποιήσεις που δεν θα ακολουθούν την παρακάτω αναπαράσταση θα έχουν **ποινή -10%**.

	a	b	c	d	e	f	g	h	
1		X	.	X	.	X	.	X	1
2		.	X	.	X	.	X	.	2
3		X	.	X	.	X	.	X	3
4		4
5		5
6		.	O	.	O	.	O	.	6
7		O	.	O	.	O	.	O	7
8		.	O	.	O	.	O	.	8

Σχήμα 1. Αρχικό ταμπλό του παιχνιδιού

	a	b	c	d	e	f	g	h	
1		X	.	X	.	X	.	.	1
2		.	X	.	.	.	X	.	2
3		X	.	X	.	X	.	X	3
4		.	X	.	X	.	.	O	4
5		O	.	5
6		.	O	.	O	.	O	.	6
7		O	.	O	.	O	.	O	7
8		.	O	.	O	.	O	.	8

Σχήμα 2. Ταμπλό του παιχνιδιού μετά από μερικές κινήσεις

2. Η ντάμα είναι παιχνίδι δύο παικτών, οπότε θα πρέπει να ζητάτε κάθε φορά από τον παίκτη που είναι η σειρά του να παίζει να δώσει την κίνηση που θέλει να κάνει με τη μορφή εντολής από το πληκτρολόγιο. Η εντολή θα δίνει τις **συντεταγμένες από το πιόνι που θα μετακινηθεί** και τις **συντεταγμένες του κενού τετραγώνου** που θα τοποθετηθεί **χωρισμένα με μία παύλα**. Ο τρόπος που θα φαίνεται αυτό στον χρήστη φαίνεται παρακάτω. Υλοποιήσεις που θα ζητούν την κίνηση διαφορετικά θα έχουν **ποινή -10%**.

> player X give your move: a3-b4

> player O give your move: b6-a5

Αν ο χρήστης δώσει τις τιμές σε άλλη μορφή ή δώσει μη έγκυρες τιμές, το πρόγραμμά σας θα πρέπει να του εμφανίζει κατάλληλο μήνυμα και να του ζητάει την εισαγωγή νέας κίνησης. Θα πρέπει σε κάθε περίπτωση να ελέγχετε την είσοδο από τον χρήστη με κατάλληλες συναρτήσεις διαχείρισης χαρακτήρων και αλφαριθμητικών.

3. Σε κάθε κίνηση του παίκτη θα πρέπει να γίνεται **έλεγχος αν υπάρχει κάποιο αναγκαστικό «φάγωμα»** σε πιόνι του αντιπάλου, και αν υπάρχει θα πρέπει να επισημαίνεται στον παίκτη και να μην επιτρέπεται άλλη κίνηση. Σε περίπτωση που ένα πιόνι μπορεί να φάει με διαδοχικές κινήσεις πολλά αντίπαλα πιόνια, τότε ο χρήστης θα πρέπει να κάνει την πρώτη κίνηση και μετά **με αναδρομικό τρόπο** το πρόγραμμά σας θα πρέπει να εκτελεί τις υπόλοιπες κινήσεις έως ότου δεν μπορούν να φαγωθούν άλλα πιόνια, ή το πιόνι φτάσει στην απέναντι άκρη του ταμπλό. Υλοποιήσεις χωρίς αυτήν τη λειτουργικότητα ή χωρίς την αναδρομή θα έχουν **ποινή -25%**.

4. Αν ο χρήστης εισάγει ως **εντολή κίνησης το z9-z9** σημαίνει ότι επιθυμεί να τερματίσει την τρέχουσα παρτίδα, οπότε θα πρέπει να τον ρωτήσετε αν θέλει να αποθηκεύσει την παρτίδα ή να παίξει μία νέα. Στην περίπτωση που θέλει να αποθηκεύσει την παρτίδα ζητάτε από τον χρήστη το **όνομα του αρχείου κειμένου** στο οποίο θέλει να το αποθηκεύσει. Τα αρχεία αποθήκευσης πρέπει πάντα να έχουν προέκταση .chk, την οποία θα προσθέτετε εσείς στο τέλος του ονόματος που δίνει ο χρήστης. Λύσεις που δεν θα υλοποιούν αυτή τη λειτουργικότητα θα έχουν **ποινή -20%**.
5. Σε κάθε κίνηση παίκτη το πρόγραμμά σας θα πρέπει να ελέγχει αν ο παίκτης μπορεί να κινήσει τα πιόνια του (ή αν του έχουν μείνει πιόνια), και αν όχι να τερματίζει το παιχνίδι ανακοινώνοντας τον παίκτη που κέρδισε.

Αναιρώντας κινήσεις

Το πρόγραμμά σας θα πρέπει να δίνει τη δυνατότητα να αναιρεθούν κινήσεις των παικτών. Αν ο παίκτης δώσει την **εντολή u9-u9**, τότε σημαίνει ότι θέλει να αναίρεσει την τελευταία του κίνηση. Οι χρήστες **μπορούν να αναιρέσουν όλες κινήσεις επιθυμούν**, ακόμα και αν οι αναίρεσεις αυτές επιστρέψουν το παιχνίδι στην αρχή του.

Για να υλοποιήσετε αυτήν τη λειτουργία θα χρειαστείτε **μία στοίβα** (θυμηθείτε ότι η στοίβα είναι η απλούστερη μορφή απλά συνδεδεμένης λίστας: ένα στοιχείο εισάγεται μόνο στην αρχή της στοίβας, και εξάγεται μόνο από την αρχή της) στην οποία **θα εισάγετε ένα στοιχείο κάθε φορά που εκτελείται μία κίνηση** από έναν παίκτη, και **θα εξάγετε ένα στοιχείο κάθε φορά που ο παίκτης αναίρει μία κίνησή του**. Για να μη μπλέξετε με συμβολισμούς κινήσεων, μπορείτε να αποθηκεύετε στη στοίβα ολόκληρο το ταμπλό μετά από κάθε κίνηση. Έτσι, η αναίρεση κινήσεων γίνεται ευκολότερη καθώς δεν θα χρειάζεται να ανασυνθέτετε το ταμπλό από τις κινήσεις των παικτών, απλά θα πρέπει να τυπώνετε το ταμπλό που είναι αποθηκευμένο στο στοιχείο της στοίβας που εξάγετε. Λύσεις που δεν θα υλοποιούν την αναίρεση κινήσεων θα έχουν **ποινή -25%**.

Η αναίρεση κινήσεων από τους παίκτες θα καταγράφεται μόνο για την τρέχουσα παρτίδα και δεν χρειάζεται να αποθηκεύεται. Επομένως αν ένας χρήστης φορτώσει μια ημιτελή παρτίδα το ιστορικό των κινήσεων της παρτίδας θα είναι κενό και θα ξεκινά να καταγράφεται με την πρώτη κίνηση συνέχισης της παρτίδας.

Hints and Tips

- Στο αρχείο που θα αποθηκεύετε το ταμπλό του παιχνιδιού μπορεί να χρειαστεί να αποθηκεύσετε και άλλες πληροφορίες (όπως για παράδειγμα ποιος παίκτης έχει σειρά να παίξει, τα ονόματα των παικτών, κλπ). Σκεφτείτε τι χρειάζεται να αποθηκεύσετε και αποφασίστε μια κατάλληλη αναπαράσταση για το αρχείο σας.

- Δομήστε τον κώδικά σας χρησιμοποιώντας κατάλληλες συναρτήσεις. Ενδεικτικά αναφέρονται μερικές από τις συναρτήσεις που πρέπει να ορίσετε:
 - `isValidMove`: ελέγχει αν η κίνηση του παίκτη είναι έγκυρη και επιστρέφει ανάλογα 1 (TRUE) ή 0 (FALSE).
 - `emptyBlock`: ελέγχει αν η θέση που θα μετακινηθεί το πούλι είναι κενή. Χρησιμοποιείται από την `isValidMove` και επιστρέφει ανάλογα 1 (TRUE) ή 0 (FALSE).
 - `pushStack`: γεμίζει τις τιμές ενός στοιχείου της στοίβας και το εισάγει στην κορυφή της στοίβας. Αν είχαμε λίστα, η συνάρτηση θα ονομαζόταν `insert`, απλά συνηθίζεται στις στοίβες η συνάρτηση να λέγεται `push`.
 - `popStack`: διαγράφει το κορυφαίο στοιχείο της στοίβας. Αν είχαμε λίστα, η συνάρτηση θα ονομαζόταν `delete`, απλά συνηθίζεται στις στοίβες η συνάρτηση να λέγεται `pop`.
 - `saveToFile`: ανοίγει το αρχείο κειμένου που υπέδειξε ο χρήστης, αποθηκεύει το ταμπλό της τρέχουσας παρτίδας στο αρχείο αυτό, και κλείνει το αρχείο.
 - `loadFromFile`: ανοίγει το αρχείο κειμένου που υπέδειξε ο χρήστης, διαβάζει το ταμπλό της παρτίδας που έχει αποθηκευτεί στο αρχείο αυτό, το φορτώνει σε ένα ταμπλό που βρίσκεται στην κύρια μνήμη, και κλείνει το αρχείο.
 - `printBoard`: τυπώνει το ταμπλό του παιχνιδιού στην οθόνη.
- Δεν χρειάζεστε μόνο αυτές τις συναρτήσεις! Χωρίστε το πρόβλημα σε υποπροβλήματα και δημιουργείτε τις συναρτήσεις που θα λύνουν κάθε υποπρόβλημα. Δομήστε τον κώδικά σας με συναρτήσεις και **μην γράψετε όλη την υλοποίηση στη main.**

Βαθμοί μπόνους (μέχρι 10%)

Ενσωμάτωση επιπλέον χαρακτηριστικών στο παιχνίδι μπορεί να πάρει μέχρι 10% bonus στη βαθμολογία (σε συνεννόηση με τον διδάσκοντα), ανάλογα με τη δυσκολία της υλοποίησης. Ενδεικτικά αναφέρονται:

- αποθήκευση και του ιστορικού των κινήσεων στο αρχείο (εκτός από το ταμπλό της παρτίδας) και ανάκτησή του κατά τη φόρτωση της παρτίδας
- παίξιμο του παιχνιδιού και με τον υπολογιστή, υλοποιώντας κάποιον απλό τρόπο για τις κινήσεις που κάνει ο Η/Υ (π.χ. `random`)

Οι ομάδες που θα υλοποιήσουν κάποια επέκταση θα πρέπει προφανώς να έχουν υλοποιήσει και όλη την βασική λειτουργικότητα που περιγράφεται παραπάνω για να πάρουν τους επιπλέον βαθμούς.

Καλή δουλειά!