

Άσκηση 4

Στην άσκηση αυτή, καλείσθε να υλοποιήσετε ένα πρόγραμμα C που θα παίζει, με αντίπαλο έναν άνθρωπο (τον χρήστη του προγράμματος), το παιχνίδι *Reversi* (ή *Othello*).

Το Reversi παίζεται σε ένα πλαίσιο 8×8 , συνοδευόμενο από ένα σύνολο κυκλικών πλακιδίων, τα οποία είναι μαύρα στη μία όψη τους και λευκά στην άλλη. Οι παίκτες παίζουν εναλλάξ. Ο ένας (μαύρος παίκτης) τοποθετεί στο πλαίσιο πλακίδια με την μαύρη όψη τους προς τα επάνω και ο άλλος (λευκός παίκτης) με τη λευκή προς τα επάνω. Ένας παίκτης επιτρέπεται να τοποθετήσει πλακίδιο του χρώματός του σε κάποια θέση στο πλαίσιο, όταν από τη θέση αυτή και σε τουλάχιστον μία ευθεία γραμμή (οριζόντια, κατακόρυφα ή διαγώνια) υπάρχουν συνεχόμενα πλακίδια του αντίθετου χρώματος και αμέσως μετά πλακίδιο του χρώματος του παίκτη. Στην περίπτωση αυτή, λέμε ότι τα πλακίδια του αντίθετου χρώματος έχουν εγκλωβιστεί, οπότε αντιστρέφονται και αποκτούν πλέον το χρώμα του παίκτη που τα εγκλώβισε. Αυτό εφαρμόζεται σε όλες τις κατευθύνσεις που εγκλωβίζονται πλακίδια του αντίπαλου παίκτη. Το παιχνίδι αρχίζει με την κατάσταση που φαίνεται πιο κάτω στο σχήμα (α) αριστερά. Πρώτος παίζει πάντοτε ο μαύρος παίκτης. Αν, για κάποιο λόγο, ένας παίκτης δεν μπορεί να κάνει νόμιμη κίνηση, τότε χάνει τη σειρά του και παίζει ο άλλος. Το παιχνίδι τελειώνει όταν γεμίσει το πλαίσιο ή όταν κανένας παίκτης δεν μπορεί να κάνει νόμιμη κίνηση. Νικητής είναι ο παίκτης που έχει στο τέλος του παιχνιδιού τα περισσότερα πλακίδια του χρώματός του στο πλαίσιο. Φυσικά, το παιχνίδι μπορεί να τελειώσει και ισόπαλο.

	a	b	c	d	e	f	g	h
1								
2								
3								
4				○	●			
5				●	○			
6								
7								
8								

(α)

	a	b	c	d	e	f	g	h
1								
2								
3								
4				○	●			
5				●	●			
6					●			
7								
8								

(β)

	a	b	c	d	e	f	g	h
1								
2								
3								
4				○	○	○		
5				●	●			
6					●			
7								
8								

(γ)

Αν στην αρχική κατάσταση (α) παραπάνω, ο μαύρος παίκτης παίζει στο **e6**, τότε εγκλωβίζει το λευκό πλακίδιο στο **e5**, οπότε αυτό γίνεται μαύρο και η κατάσταση του παιχνιδιού είναι αυτή στο σχήμα (β). Αν μετά ο λευκός παίκτης παίζει στο **f4**, τότε εγκλωβίζει το μαύρο πλακίδιο στο **e4**, το οποίο γίνεται λευκό και η νέα κατάσταση είναι η (γ).

Το Reversi είναι ένα παιχνίδι στρατηγικής και το να αναπτυχθούν αλγόριθμοι ώστε να μπορεί ένας υπολογιστής να παίζει καλά αυτό το παιχνίδι είναι αντικείμενο ενός κλάδου της Πληροφορικής, που λέγεται *Τεχνητή Νοημοσύνη* (Artificial Intelligence). Περισσότερες πληροφορίες για το παιχνίδι μπορείτε να βρείτε στον σύνδεσμο <http://en.wikipedia.org/wiki/Reversi>. Επίσης, στο διαδίκτυο υπάρχουν ιστοσελίδες στις οποίες μπορείτε να παίζετε on-line το Reversi, έτσι ώστε να εξοικειωθείτε με το παιχνίδι. Για παράδειγμα, δείτε την <http://www.mah-jongg.ch/reversi>.

Το πρόγραμμα που θα γράψετε (έστω ότι το εκτελέσιμο ονομάζεται “reversi”) θα μπορεί να δέχεται από τη γραμμή εντολής κάποιες επιλογές, για τον καθορισμό αρχικών παραμέτρων, και αφού ξεκινήσει να εκτελείται, θα πρέπει να δέχεται εντολές από ένα προκαθορισμένο ρεπερτόριο που θα καθορίζουν τη λειτουργία του.

Οι επιλογές από τη γραμμή εντολών είναι οι εξής:

- n <size> : Το τυπικό Reversi παίζεται σε πλαίσιο 8×8 . Το πρόγραμμά σας θα πρέπει να είναι σε θέση να παίζει το παιχνίδι για οποιοδήποτε **άρτιο** μέγεθος της διάστασης του πλαισίου (<size>) μεταξύ 4 και 26 (συμπεριλαμβανομένων), το οποίο καθορίζεται από την επιλογή αυτή. Αν δεν δοθεί η επιλογή, να θεωρείται ότι <size> = 8.
- d <difficulty> : Ορίζεται το επίπεδο δυσκολίας στο οποίο θα παίζεται το παιχνίδι από το πρόγραμμα που θα γράψετε. Αν δεν δοθεί η επιλογή, να θεωρείται <difficulty> = 1, που είναι το χαμηλότερο επίπεδο δυσκολίας. Περισσότερα για το επίπεδο δυσκολίας, στη συνέχεια.
- l : Όταν είναι η σειρά του χρήστη να παίζει, να του υποδεικνύει το πρόγραμμα ποιες είναι οι δυνατές κινήσεις που μπορεί να κάνει.
- w : Ο χρήστης θα είναι ο λευκός παίκτης, οπότε παίζει δεύτερος. Αν δεν δοθεί η επιλογή, ο χρήστης θα είναι ο μαύρος παίκτης, οπότε παίζει πρώτος.

Αφού ξεκινήσει να εκτελείται το πρόγραμμα που θα γράψετε, να δέχεται εντολές που καθορίζουν παραμέτρους, αλλά και τη λειτουργία του παιχνιδιού. Οι εντολές αυτές είναι:

- newgame** [<size>] : Αρχίζει εξ αρχής νέο παιχνίδι. Αν δοθεί η προαιρετική παράμετρος <size>, τότε αυτή θα είναι η διάσταση του πλαισίου (αλλιώς 8).
- play** <move> : Η εντολή είναι αποδεκτή όταν είναι η σειρά του χρήστη να παίζει. Το <move> είναι η κίνηση που επιθυμεί να κάνει ο παίκτης, στη μορφή <letter><number>, όπου <letter> είναι η στήλη (a, b, c, ...) και <number> είναι η γραμμή (1, 2, 3, ...).
- cont** : Η εντολή είναι αποδεκτή όταν είναι η σειρά του προγράμματος να παίζει, οπότε γίνεται η κίνηση που αυτό αποφασίζει.
- undo** : Ο παίκτης αναιρεί την προηγούμενη κίνησή του. Αν έχει γίνει ήδη και η κίνηση του προγράμματος, αναιρείται και αυτή.
- suggest** : Ο παίκτης ζητά από το πρόγραμμα να του υποδείξει την κίνηση που θα ήταν καλό να κάνει. Η εντολή είναι αποδεκτή όταν είναι η σειρά του παίκτη να παίζει.
- selectcolor** <color> : Ο χρήστης επιλέγει να είναι του χρώματος <color>, όπου το <color> είναι black ή white.
- showlegal** <option> : Ο χρήστης επιλέγει αν επιθυμεί (<option> = on) ή όχι (<option> = off) να του υποδεικνύει το πρόγραμμα ποιες είναι οι διαθέσιμες νόμιμες κινήσεις του, όταν είναι η σειρά του να παίζει.
- level** [<difficulty>] : Ο χρήστης επιλέγει σαν επίπεδο δυσκολίας το <difficulty>. Αν δεν δοθεί <difficulty>, το πρόγραμμα εκτυπώνει ποιο είναι τρέχον επίπεδο δυσκολίας του παιχνιδιού.
- save** <statefile> : Αποθηκεύει την τρέχουσα κατάσταση του παιχνιδιού στο αρχείο <statefile>.
- load** <statefile> : Φορτώνει από το αρχείο <statefile> μία αποθηκευμένη κατάσταση παιχνιδιού. Περισσότερα για τη μορφή των αρχείων καταστάσεων, στη συνέχεια.
- showstate** : Εκτυπώνεται η τρέχουσα κατάσταση του παιχνιδιού.

Τμήμα μίας ενδεικτικής εκτέλεσης του προγράμματος φαίνεται στη συνέχεια:

```
% ./reversi -d 3 -l
```

```
> newgame
```

```
  a  b  c  d  e  f  g  h
+---+---+---+---+---+---+---+---+
1 |  |  |  |  |  |  |  |  |
+---+---+---+---+---+---+---+---+
2 |  |  |  |  |  |  |  |  |
+---+---+---+---+---+---+---+---+
3 |  |  |  | * |  |  |  |  |
+---+---+---+---+---+---+---+---+
4 |  |  | * | W | B |  |  |  |
+---+---+---+---+---+---+---+---+
5 |  |  |  | B | W | * |  |  |
+---+---+---+---+---+---+---+---+
6 |  |  |  |  | * |  |  |  |
+---+---+---+---+---+---+---+---+
7 |  |  |  |  |  |  |  |  |
+---+---+---+---+---+---+---+---+
8 |  |  |  |  |  |  |  |  |
+---+---+---+---+---+---+---+---+
```

```
Black player (human) plays now
```

```
White: 2 - Black: 2
```

```
> play f5
```

```
  a  b  c  d  e  f  g  h
+---+---+---+---+---+---+---+---+
1 |  |  |  |  |  |  |  |  |
+---+---+---+---+---+---+---+---+
2 |  |  |  |  |  |  |  |  |
+---+---+---+---+---+---+---+---+
3 |  |  |  |  |  |  |  |  |
+---+---+---+---+---+---+---+---+
4 |  |  |  | W | B |  |  |  |
+---+---+---+---+---+---+---+---+
5 |  |  |  | B | B | B |  |  |
+---+---+---+---+---+---+---+---+
6 |  |  |  |  |  |  |  |  |
+---+---+---+---+---+---+---+---+
7 |  |  |  |  |  |  |  |  |
+---+---+---+---+---+---+---+---+
8 |  |  |  |  |  |  |  |  |
+---+---+---+---+---+---+---+---+
```

```
Move played: f5
```

```
White player (computer) plays now
```

```
White: 1 - Black: 4
```

> cont

	a	b	c	d	e	f	g	h
1								
2								
3				*				
4				*	W	B		
5				*	W	B	B	
6				*	W			
7				*				
8								

Move played: d6

Black player (human) plays now

White: 3 - Black: 3

> play c6

	a	b	c	d	e	f	g	h
1								
2								
3								
4					W	B		
5					B	B	B	
6				B	W			
7								
8								

Move played: c6

White player (computer) plays now

White: 2 - Black: 5

> save file.rev

>

Άλλη μία ενδεικτική εκτέλεση του προγράμματος, για παιχνίδι σε πλαίσιο 4 × 4, είναι η εξής:

```
% ./reversi -n 4 -w
> showlegal on
> level 2
> newgame
  a  b  c  d
+---+---+---+---+
1 |  |  |  |  |
+---+---+---+---+
2 |  | W | B |  |
+---+---+---+---+
3 |  | B | W |  |
+---+---+---+---+
4 |  |  |  |  |
+---+---+---+---+
Black player (computer) plays now
```

```
White: 2 - Black: 2
> cont
  a  b  c  d
+---+---+---+---+
1 |  |  |  |  |
+---+---+---+---+
2 |  | W | B | * |
+---+---+---+---+
3 |  | B | B |  |
+---+---+---+---+
4 |  | * | B | * |
+---+---+---+---+
Move played: c4
White player (human) plays now
```

```
White: 1 - Black: 4
> play d4
  a  b  c  d
+---+---+---+---+
1 |  |  |  |  |
+---+---+---+---+
2 |  | W | B |  |
+---+---+---+---+
3 |  | B | W |  |
+---+---+---+---+
4 |  |  | B | W |
+---+---+---+---+
Move played: d4
Black player (computer) plays now
```

```
White: 3 - Black: 3
```

> cont

```
      a   b   c   d
+----+----+----+----+
1 |   |   |   |   |
+----+----+----+----+
2 |   | W | B | * |
+----+----+----+----+
3 |   | B | B | B |
+----+----+----+----+
4 |   | * | B | W |
+----+----+----+----+
```

Move played: d3

White player (human) plays now

White: 2 - Black: 5

> play b4

```
      a   b   c   d
+----+----+----+----+
1 |   |   |   |   |
+----+----+----+----+
2 |   | W | B |   |
+----+----+----+----+
3 |   | W | B | B |
+----+----+----+----+
4 |   | W | W | W |
+----+----+----+----+
```

Move played: b4

Black player (computer) plays now

White: 5 - Black: 3

> cont

```
      a   b   c   d
+----+----+----+----+
1 |   |   | * |   |
+----+----+----+----+
2 | * | W | B | * |
+----+----+----+----+
3 |   | B | B | B |
+----+----+----+----+
4 | B | W | W | W |
+----+----+----+----+
```

Move played: a4

White player (human) plays now

White: 4 - Black: 5

> play d2

```
  a  b  c  d
+---+---+---+---+
1 |  |  |  |  |
+---+---+---+---+
2 |  | W | W | W |
+---+---+---+---+
3 |  | B | W | W |
+---+---+---+---+
4 | B | W | W | W |
+---+---+---+---+
```

Move played: d2

Black player (computer) plays now

White: 8 - Black: 2

> cont

```
  a  b  c  d
+---+---+---+---+
1 |  | * | * | B |
+---+---+---+---+
2 | * | W | B | W |
+---+---+---+---+
3 | * | B | W | W |
+---+---+---+---+
4 | B | W | W | W |
+---+---+---+---+
```

Move played: d1

White player (human) plays now

White: 7 - Black: 4

> play b1

```
  a  b  c  d
+---+---+---+---+
1 |  | W |  | B |
+---+---+---+---+
2 | * | W | W | W |
+---+---+---+---+
3 | * | B | W | W |
+---+---+---+---+
4 | B | W | W | W |
+---+---+---+---+
```

Move played: b1

White player (human) plays now

White: 9 - Black: 3

> cont

It's your turn to play

```

> play a2
  a  b  c  d
+---+---+---+---+
1 |   | W |   | B |
+---+---+---+---+
2 | W | W | W | W |
+---+---+---+---+
3 |   | W | W | W |
+---+---+---+---+
4 | B | W | W | W |
+---+---+---+---+
Move played: a2

```

White: 11 - Black: 2

END OF GAME

```

> quit
%
```

Όσον αφορά τη μορφή των αρχείων καταστάσεων, που χειρίζονται οι εντολές `save` και `load`, αυτή πρέπει να είναι η εξής: Αρχικά, μέσα στο αρχείο υπάρχει (σε δυαδική μορφή) ένας ακέραιος που είναι η διάσταση του πλαισίου (έστω n) και μετά n^2 bytes, τα οποία κωδικοποιούν τα περιεχόμενα των θέσεων του πλαισίου, κατά γραμμές. Για τις θέσεις με μαύρο πλακίδιο υπάρχει το 'b', για εκείνες με λευκό πλακίδιο το 'w' και για τις κενές θέσεις το 'n' (ASCII κωδικοί). Το αρχείο στο οποίο αποθηκεύτηκε κάποια κατάσταση του παιχνιδιού, κατά την πρώτη από τις δύο προηγούμενες ενδεικτικές εκτελέσεις, είναι το <http://www.di.uoa.gr/~ip/hwfiles/file.rev>. Σημειώνεται ότι το αρχείο δημιουργήθηκε σε μηχάνημα Sun.

Επίπεδο δυσκολίας - Ο αλγόριθμος minimax

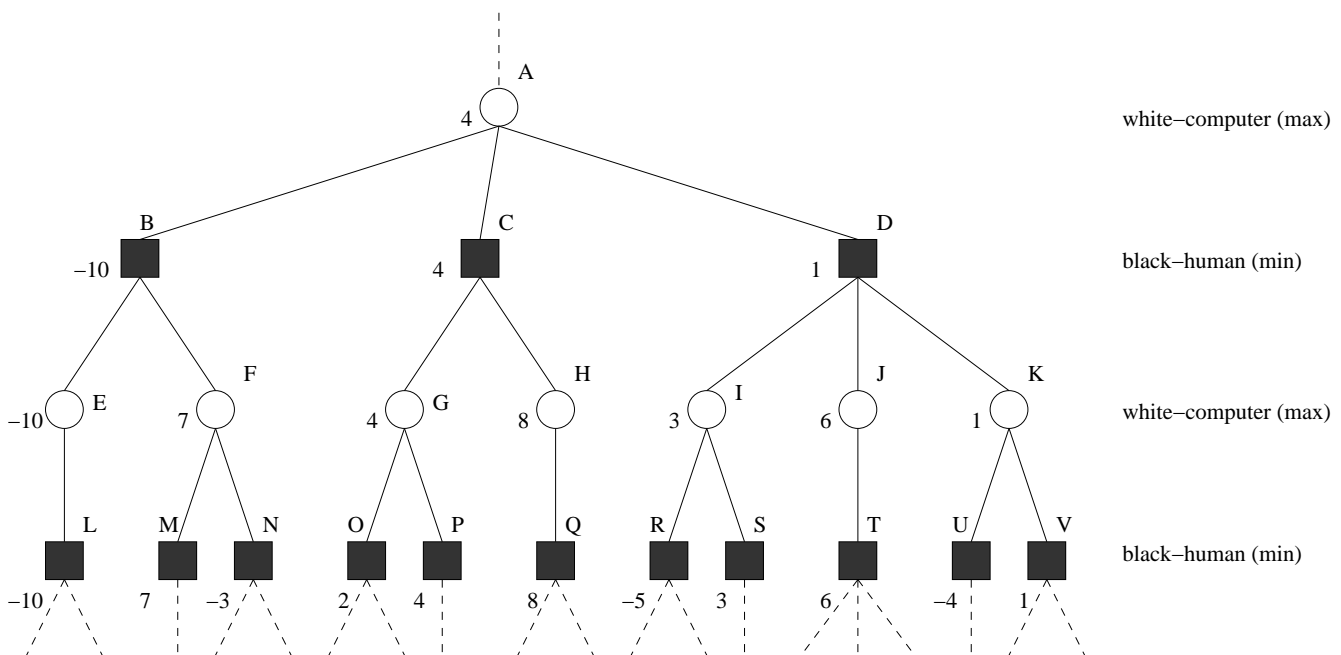
Ένα θέμα που θα πρέπει να συζητηθεί εδώ σε κάποιο βαθμό είναι αυτό του επιπέδου δυσκολίας στο οποίο θα καλείται να παίξει το πρόγραμμα που θα γράψετε.

Έστω ότι το πρόγραμμα βρίσκεται στο σημείο που πρέπει να αποφασίσει τι κίνηση να κάνει. Είναι δεδομένη η κατάσταση του παιχνιδιού εκείνη τη στιγμή, και δεδομένου του χρώματος (μαύρο/λευκό) του παίκτη-υπολογιστή, μπορεί το πρόγραμμα να βρει ποιες είναι οι νόμιμες κινήσεις που μπορεί να κάνει. Πώς όμως θα αποφασίσει ποια από αυτές να επιλέξει; Προφανώς πρέπει να είναι εκείνη που θα το οδηγήσει στην ευνοϊκότερη δυνατή κατάσταση για το χρώμα του. Οπότε χρειάζεται ένα μέτρο του πόσο καλή είναι μία κατάσταση παιχνιδιού για κάποιο παίκτη δεδομένου χρώματος. Ένα πολύ απλό τέτοιο μέτρο είναι η διαφορά του πλήθους των πλακιδίων του χρώματος του παίκτη μείον το πλήθος των πλακιδίων του αντίπαλου χρώματος. Δεν είναι το μοναδικό μέτρο, ούτε το καλύτερο δυνατό, αλλά είναι αντικείμενο της άσκησης και το να σκεφτείτε ή να βρείτε και άλλα χρήσιμα μέτρα της ποιότητας μίας κατάστασης, τα οποία να συνδυάσετε κατάλληλα ώστε να προκύψει μία όσο το δυνατόν περισσότερο αξιόπιστη συνισταμένη.

Επιστρέφοντας τώρα στο πρόβλημα της επιλογής από το πρόγραμμα της καταλληλότερης κίνησης μεταξύ όλων των δυνατών, μία, σχετικά απλοϊκή, προσέγγιση, είναι να επιλεγεί εκείνη η κίνηση που οδηγεί σε κατάσταση με την καλύτερη ποιότητα, για τον παίκτη-υπολογιστή. Θα μπορούσαμε τότε

να πούμε ότι το πρόγραμμα παίζει σε επίπεδο δυσκολίας 1. Η προσέγγιση αυτή χαρακτηρίζεται σαν απλοϊκή, διότι δίνει κάποιο πρόσκαιρο πλεονέκτημα στο πρόγραμμα, όμως μπορεί να οδηγήσει σε κατάσταση όπου ο παίκτης-άνθρωπος να έχει διαθέσιμη κάποια πολύ καίρια κίνηση, η οποία να ανατρέπει υπέρ του την ισορροπία του παιχνιδιού. Μήπως θα έπρεπε το πρόγραμμα να προβλέψει κάτι τέτοιο και να μην κάνει τελικά τη λάθος κίνηση; Με άλλα λόγια, δεν θα μπορούσε, αντί να λαμβάνει υπόψη του το τι μπορεί να συμβεί μετά από μία κίνηση στην τρέχουσα κατάσταση, να το κάνει αυτό για τις δύο επόμενες κινήσεις (επίπεδο 2); Και γιατί να κάνει πρόβλεψη μόνο για δύο κινήσεις και όχι για τρεις; Καταλαβαίνετε ότι αυτή η λογική γενικεύεται, ώστε να μπορεί το πρόγραμμα να εξερευνά τα πιθανά εναλλακτικά για τις επόμενες κινήσεις, οπότε τότε λέμε ότι παίζει σε επίπεδο δυσκολίας d.

Η μέθοδος που περιγράφηκε πιο πάνω σε γενικές γραμμές είναι ο αλγόριθμος minimax που έχει αναπτυχθεί για παιχνίδια δύο παιχτών, στο πλαίσιο της Τεχνητής Νοημοσύνης. Για να καταλάβετε καλύτερα τη λογική αυτού του αλγορίθμου, δείτε το επόμενο σχήμα που παριστάνει τμήμα ενός δέντρου παιχνιδιού (game tree).



Έστω ότι κάποια στιγμή το παιχνίδι βρίσκεται στην κατάσταση A, στην οποία είναι η σειρά του παίκτη-υπολογιστή να παίζει με τα λευκά πλακίδια, και έστω ότι οι δυνατές κινήσεις που μπορεί να κάνει θα οδηγούσαν το παιχνίδι σε κάποια από τις καταστάσεις B, C ή D. Αν το πρόγραμμα έπαιζε σε επίπεδο δυσκολίας 1, απλά θα αποτιμούσε την ποιότητα αυτών των τριών καταστάσεων και θα έκανε την κίνηση που θα ήταν η πιο συμφέρουσα γι' αυτό. Έστω όμως ότι έχει καθορισθεί το πρόγραμμα να παίζει σε επίπεδο δυσκολίας 3. Τότε θα πρέπει να πάρει την απόφασή του προβλέποντας τι μπορεί να συμβεί μετά από τρεις κινήσεις. Οπότε, για κάθε πιθανή επόμενη κατάσταση από τις B, C ή D, στις οποίες είναι σειρά του παίκτη-άνθρώπου να παίζει με τα μαύρα πλακίδια, πρέπει το πρόγραμμα να βρει τις επόμενες πιθανές καταστάσεις, στις οποίες θα είναι η σειρά του παίκτη-υπολογιστή να παίζει και, τέλος, για κάθε μία από αυτές, να βρει τις πιθανές επόμενες, στις οποίες θα είναι η σειρά του παίκτη-άνθρώπου να παίζει. Αυτές οι καταστάσεις φαίνονται στο τελευταίο επίπεδο του σχήματος.

Αυτό που πρέπει να κάνει το πρόγραμμα είναι να αποτιμήσει τις καταστάσεις του τελευταίου επιπέδου και να διαδώσει προς τα επάνω τις αποτιμήσεις με την εξής λογική. Σε κάθε κατάσταση που είναι η σειρά του παίκτη-υπολογιστή να παίζει με τα άσπρα πλακίδια, η αποτίμησή της είναι το μέγιστο των αποτιμήσεων των καταστάσεων που μπορεί να προκύψουν από αυτήν (γι' αυτό, ο παίκτης-υπολογιστής ονομάζεται max). Ο λόγος είναι ότι αν βρεθεί το παιχνίδι σε μία τέτοια κατάσταση και είναι η σειρά του παίκτη-υπολογιστή να παίζει, το πρόγραμμα θα πρέπει να επιλέξει την πιο συμφέρουσα κίνηση.

Αντίστροφα, σε κάθε κατάσταση που είναι η σειρά του παίκτη-ανθρώπου να παίξει με τα μαύρα πλακίδια, η αποτίμησή της είναι το ελάχιστο των αποτιμήσεων των καταστάσεων που μπορεί να προκύψουν από αυτήν (γί αυτό, ο παίκτης-άνθρωπος ονομάζεται *min*). Η λογική είναι ότι ο παίκτης-άνθρωπος θα κάνει την λιγότερο συμφέρουσα κίνηση για τον παίκτη-υπολογιστή.¹

Στο προηγούμενο σχήμα, οι αριθμοί στους κόμβους του τελευταίου επιπέδου είναι οι αποτιμήσεις των αντίστοιχων καταστάσεων, όπως αυτές υπολογίζονται από τη συνάρτηση που έχει επιλεγεί για να εκτιμά την ποιότητα των καταστάσεων από την πλευρά του παίκτη-υπολογιστή (π.χ. διαφορά πλήθους λευκών μείον πλήθος μαύρων πλακιδίων). Οι αριθμοί στους ενδιάμεσους κόμβους είναι οι αποτιμήσεις των αντίστοιχων καταστάσεων, πάντα από την πλευρά του παίκτη-υπολογιστή, όπως αυτές διαδίδονται από το τελευταίο επίπεδο προς τα ανώτερα. Τελικά, για το δεδομένο παράδειγμα, όταν η κατάσταση παιγνιδιού είναι η *A*, η κίνηση που συμφέρει το πρόγραμμα να κάνει είναι αυτή που το οδηγεί στην κατάσταση *C*.

Μπορείτε να δείτε μία πολύ ενδιαφέρουσα οπτικοποιημένη εφαρμογή του αλγορίθμου *minimax* στο σύνδεσμο <http://upload.wikimedia.org/wikipedia/en/e/e1/Plminmax.gif>. Επίσης, για πληροφόρησή σας, ως αναφερθεί ότι υπάρχει και ένας αλγόριθμος για παιγνίδια δύο παικτών, που αποτελεί μία βελτιωμένη εκδοχή του *minimax*, ο αλγόριθμος των *άλφα-βήτα αποκοπών* (*alpha-beta pruning*).² Σημειώνεται ότι δεν είναι μέσα στις απαιτήσεις της άσκησης να υλοποιήσετε τον αλγόριθμό αυτό, αλλά αναφέρεται γιατί μπορεί να σας χρησιμεύσει αν επιθυμείτε να συμμετάσχει η άσκησή σας με αξιώσεις στο κύπελλο *Reversi* που περιγράφεται στο τέλος της εκφώνησης.

Παραδοτέο

Το πρόγραμμα που θα γράψετε θα πρέπει να είναι δομημένο σε ένα σύνολο από **τουλάχιστον τρία πηγαία αρχεία C** (με κατάληξη *.c*) και **τουλάχιστον δύο αρχεία επικεφαλίδας** (με κατάληξη *.h*). Για να παραδώσετε τη δουλειά σας, θα πρέπει να κάνετε τα εξής:

- Τοποθετήστε όλα τα αρχεία (πηγαία και αρχεία επικεφαλίδας) μέσα σ' ένα κατάλογο που θα δημιουργήσετε, έστω με όνομα *reversi*, σε κάποιο σύστημα *Unix*. Επίσης, τοποθετήστε στον κατάλογο αυτό και ένα αρχείο με όνομα *README*, στο οποίο να δίνετε οδηγίες για τη μεταγλώττιση των αρχείων και την κατασκευή του τελικού εκτελέσιμου, καθώς και ό,τι άλλο κρίνετε σκόπιμο να επισημάνετε. Προαιρετικά, μπορείτε να παραδώσετε και ένα αρχείο *Makefile* που να αναλαμβάνει όλη τη διαδικασία της κατασκευής του τελικού εκτελέσιμου μέσω της εντολής *"make"* (δώστε *"man make"* για περισσότερες λεπτομέρειες).
- Όντας στον κατάλογο που περιέχει τον κατάλογο *reversi*, δημιουργήστε ένα επιπεδοποιημένο *tar* αρχείο (έστω με όνομα *reversi.tar*) που περιέχει τον κατάλογο *reversi* και όλα του τα περιεχόμενα. Αυτό γίνεται με την εντολή *"tar cvf reversi.tar reversi"*.³
- Συμπιέστε το αρχείο *reversi.tar*, ώστε να δημιουργηθεί το αρχείο *reversi.tar.gz*. Αυτό γίνεται με την εντολή *"gzip reversi.tar"*.⁴
- Το αρχείο *reversi.tar.gz* είναι που θα πρέπει να υποβάλετε, με διαδικασία που θα ανακοινωθεί σύντομα.

¹Στους χαρακτηρισμούς των δύο παικτών, ως *max* και *min*, οφείλεται και το όνομα του αλγορίθμου (*minimax*).

²Για περισσότερες πληροφορίες, στο http://en.wikipedia.org/wiki/Alpha-beta_pruning.

³Αν θέλετε να ανακτήσετε την δενδρική δομή που έχει φυλαχθεί σ' ένα επιπεδοποιημένο *tar* αρχείο *file.tar*, αυτό μπορεί να γίνει με την εντολή *"tar xvf file.tar"*.

⁴Αν θέλετε να αποσυμπιέσετε ένα αρχείο *file.gz* που έχει συμπιεσθεί με την εντολή *gzip*, αυτό μπορεί να γίνει με την εντολή *"gzip -d file.gz"*.

Ομαδική άσκηση

Η άσκηση αυτή μπορεί να παραδοθεί και από **ομάδες των δύο ατόμων**. Στην περίπτωση αυτή, θα παραδοθεί μόνο από το ένα μέλος της ομάδας, αλλά μέσα στο αρχείο README θα αναφέρονται σαφώς τα στοιχεία των δύο μελών. Ο στόχος της διαδικασίας αυτής είναι να ενισχυθεί η ιδέα της **ισότιμης** συνεργασίας σε μία ομάδα για την επίτευξη ενός στόχου. Αν τα μέλη της ομάδας έχουν υλοποιήσει διαφορετικά τμήματα της άσκησης, θα πρέπει στο αρχείο README να αναφέρεται ρητά τι έχει υλοποιήσει κάθε μέλος, έτσι ώστε στην προφορική εξέταση που θα ακολουθήσει, να μην υπάρχει η απαίτηση να έχει κάποιο μέλος της ομάδας πλήρη γνώση του πώς έχουν υλοποιηθεί τα τμήματα στα οποία εκείνο δεν έχει εμπλακεί.

Κύπελλο Reversi (BONUS βαθμολογία)

Από τις ασκήσεις που θα υποβληθούν, θα προεπιλεγούν οι 16 (το μέγιστο) καλύτερες για να συμμετάσχουν σε κύπελλο Reversi που θα διεξαχθεί, με σύστημα knock out, την ημέρα της προφορικής εξέτασης των ασκήσεων, μετά το τέλος της. Για τις τρεις καλύτερες ασκήσεις, θα υπάρξει επιβράβευση στη βαθμολογία τους κατά 100%, 70% και 40%, κατά σειρά. Η νικήτρια άσκηση θα αναμετρηθεί και με το πρόγραμμα που έχει υλοποιηθεί από συνεργάτη του μαθήματος, και στην περίπτωση που το κερδίσει, θα υπάρξει επιπλέον bonus 50%. Περισσότερες λεπτομέρειες για το κύπελλο θα ανακοινωθούν εν καιρώ στο φόρουμ του μαθήματος. Την εποπτεία της διοργάνωσης του κυπέλλου θα αναλάβει επιτροπή από συνεργάτες του μαθήματος.