



Γενικές Οδηγίες και Περιγραφή των Παραδοτέων της Εργασίας

Φάσεις

- **Φάση A: 10% τελικού βαθμού**
 - 16 Νοεμβρίου - 4 Δεκεμβρίου
- **Φάση B: 15% τελικού βαθμού** (ρήτρα 5 για επιτυχία στο μάθημα)
 - 5 Δεκεμβρίου - 22 Δεκεμβρίου (bonus 5%) ή
 - 5 Δεκεμβρίου - 8 Ιανουαρίου 2016 (χωρίς bonus)

Φάση A

Σε αυτή τη φάση πρέπει να γίνει ο σχεδιασμός της εφαρμογής βάσει των ιδεών και των αρχών του αντικειμενοστρεφούς προγραμματισμού που έχετε διδαχτεί. Αποτέλεσμα αυτής της φάσης είναι ο καθορισμός των αντικείμενων, των χαρακτηριστικών και της συμπεριφοράς τους που απαιτούνται για να αναπαραστήσουν τις καταστάσεις και τις λειτουργίες του θέματος της εργασίας, όπως έχουν περιγραφεί στην εκφώνηση.

Παραδοτέα αυτής της φάσης είναι :

- **Γραπτή αναφορά** (όχι σε greeklish) η οποία θα περιγράφει τα παραπάνω στοιχεία και θα παρουσιάζει το σχέδιο υλοποίησης της προγραμματιστικής εργασίας, έτσι ώστε να είναι έτοιμο το πέρασμα στην επόμενη φάση της υλοποίησης. Θα πρέπει να συμπεριλαμβάνονται και **UML class diagrams**.
- **Πηγαίος κώδικας** που περιλαμβάνει τις **διεπαφές** (interfaces) και το περίγραμμα των κλάσεων (class outline) Java του προγράμματος σας, συνοδευόμενες από τα απαραίτητα **javadoc** σχόλια, τα οποία θα καθοδηγήσουν την υλοποίηση της επόμενης φάσης.

Επιγραμματικά, οι σημαντικότερες εργασίες που πρέπει να γίνουν σε αυτή τη φάση είναι:

- Αναγνώριση των κλάσεων και διεπαφών για κάθε μικρή και μεγάλη συνιστώσα του προγράμματος. Αναγνώριση των ευθυνών κάθε κλάσης και των πιθανών σχέσεων της με άλλες.
- Εύρεση των χαρακτηριστικών και των μεθόδων κάθε κλάσης.
- Εύρεση της συμπεριφοράς (behaviour) κάθε κλάσης και διεπαφής, καθώς και της επικοινωνίας μέσω μηνυμάτων (method calls) που χρειάζεται να έχουν μεταξύ τους.
- Οργάνωση των κλάσεων σε ιεραρχίες με στόχο την μέγιστη δυνατή επαναχρησιμοποίηση του κώδικα σας.
- Για κάθε κλάση που υλοποιεί μια διεπαφή δώστε τις υπογραφές (signatures) για όλες τις μεθόδους και τις εκ των προτέρων, εκ των υστέρων και αμετάβλητες συνθήκες (preconditions, postconditions, invariants) που τις διέπουν σε μορφή javadoc σχολίων.

ΣΗΜΑΝΤΙΚΟ: Σημειώστε ότι όσο πληρέστερη και αναλυτικότερη δουλειά κάνετε στην σχεδίαση τόσο πιο σωστή, εύκολη και επιτυχημένη θα είναι η υλοποίηση.

Φάση Β

Σε αυτή τη φάση πρέπει να γίνει η κυρίως υλοποίηση της εφαρμογής, βάσει της σχεδίασης που έχει προηγηθεί (φάση Α). Δεν επιβάλλεται να χρησιμοποιηθεί αυτούσια η σχεδίαση της φάσης Α, καθώς κάποιες σχεδιαστικές επιλογές αποδεικνύεται στην πορεία ότι χρειάζονται αναθεώρηση. Εντούτοις, η τελική βαθμολογία θα εξαρτηθεί και από τη συνέπεια της τελικής υλοποίησης ως προς την αρχική σχεδίαση.

Σε αυτή τη φάση, παραδοτέα είναι :

- ο **πηγαίος κώδικας** που υλοποιεί την εργασία
- αναλυτικές οδηγίες για το πώς μεταγλωττίζεται και πώς τρέχει το πρόγραμμά σας (README, Ant, Maven κλπ)
- αναφορά, στην οποία θα αναλύεται :
 - η τελική σχεδίαση της εφαρμογής,
 - ποιές αλλαγές έγιναν σε σχέση με τη σχεδίαση της Α' φάσης (και γιατί),
 - οι αλγόριθμοι που χρησιμοποιήθηκαν
 - τυχόν διαφοροποιήσεις στους κανόνες σε σχέση με τους κανόνες που δίνονται παραπάνω
 - οι σχεδιαστικές και προγραμματιστικές αποφάσεις που ελήφθησαν και πώς αυτό αντανακλάται στον τελικό χρήστη (π.χ. ευκολία/δυσκολία χειρισμού)
 - τα προβλήματα που αντιμετωπίστηκαν
 - τα JUnit tests που φτιάχτηκαν για τον έλεγχο της ορθότητας
 - γενικά ό,τι άλλο κρίνετε απαραίτητο να αναφερθεί

Βαθμολογία Εργασίας

Για τη βαθμολογία της εργασίας σας θα συνεκτιμηθούν:

- εάν (και πόσο) η σχεδίαση της εφαρμογής εφαρμόζει τις έννοιες και τεχνικές του αντικειμενοστρεφούς προγραμματισμού που διδαχθήκατε στο μάθημα
- εάν (και πόσο) υλοποιήθηκαν οι υποχρεωτικές λειτουργίες της εφαρμογής
- η πληρότητα της τελικής αναφοράς, η οποία θα καταγράφει και θα τεκμηριώνει την σχεδίαση και υλοποίηση της εφαρμογής.

Για διευκρινήσεις σχετικά με την παραπάνω εργασία μπορείτε να στέλνετε μηνύματα με απορίες σας στο σχετικό **forum** στην ιστοσελίδα του moodle. Ερωτήσεις που στέλνονται στην λίστα του μαθήματος hy252-list@csd.uoc.gr **δε θα απαντώνται**.

Καλή Εργασία



Project

Εκπαιδευτικοί Στόχοι

Προδιαγραφή και σχεδίαση συστήματος
Προδιαγραφή Αφαιρετικών Τύπων Δεδομένων (ΑΤΔ) που απαιτούνται για την επιτυχή ολοκλήρωσή του συστήματος
Υλοποίηση εξαρτημάτων (ΑΤΔ) του συστήματος των οποίων η προδιαγραφή δίνεται
Χρήση κληρονομικότητας και πολυμορφισμού
Δημιουργία Γραφικής Διεπαφής
Απεξάρτηση του πυρήνα του συστήματος από τη Γραφική Διεπαφή
Επαναχρησιμοποίηση διεπαφών και κλάσεων
Χρήση JFC (Java Collection Framework)
Τεκμηρίωση, Έλεγχος

Σύντομη Περιγραφή Εργασίας

Στόχος. Στην εργασία αυτή καλείστε να σχεδιάσετε και να υλοποιήσετε μία παραλλαγή του κλασσικού παιχνιδιού **Stratego** με τίτλο **Stratego Ice vs Fire**.

Η εργασία είναι ατομική και απαγορεύεται η χρήση κώδικα που δεν έχετε γράψει οι ίδιοι. Σε περίπτωση εντοπισμού αντιγραφής η εργασία θα μηδενίζεται.

Αν κάποιος δεν γνωρίζει αυτό το παιχνίδι, καλό είναι να δει το παρακάτω video:

https://www.youtube.com/watch?v=o_1-w2Tlqac .

(πατήστε το CC αν θέλετε υπότιτλους στα αγγλικά)

Περιγραφή και Σκοπός του Παιχνιδιού.

Οι δύο παίκτες έχουν ένα στρατό που αποτελείται από 30 πιόνια, συμπεριλαμβανομένης της σημαίας. Ο σκοπός του παιχνιδιού είναι ο παίχτης να αιχμαλωτίσει τη σημαία του αντιπάλου του και να προστατεύσει τη δικιά του. Ο πρώτος από τους 2 παίχτες που τα καταφέρει ανακηρύσσεται νικητής του παιχνιδιού. Ο κάθε παίχτης τοποθετεί τα πιόνια του στις τρεις πρώτες σειρές που έχει μπροστά του, ενώ κάθε πιόνι έχει ένα όνομα και ένα βαθμό κατάταξης. Όσο μεγαλύτερος είναι ο βαθμός τόσο ισχυρότερο είναι το πιόνι. Αν ένα πιόνι υψηλότερου βαθμού επιτεθεί σε ένα πιόνι χαμηλότερου βαθμού, το πιόνι με τον υψηλότερο βαθμό κερδίζει και το άλλο βγαίνει από το παιχνίδι. Η Σημαία μπορεί να κατακτηθεί από οποιοδήποτε πιόνι του αντιπάλου μπορεί να μετακινηθεί.

Αναλυτική Περιγραφή της Εργασίας

Περιεχόμενα παιχνιδιού

Τα παιχνίδια αποτελούνται από :

- 1 ταμπλό με διαστάσεις 8x10 (8 σειρές-10 στήλες)
- 30 κόκκινα πιόνια (στρατός Volcandria)
- 30 μπλε πιόνια (στρατός Everwinter)

Προετοιμασία του παιχνιδιού

Κάθε ένας από τους 2 παίκτες τοποθετεί τα **30 πιόνια**¹ του στις 3 πρώτες σειρές που βρίσκονται μπροστά του. Κάθε πιόνι μπορεί να τοποθετηθεί σε οποιαδήποτε θέση από αυτές τις 30 όπως φαίνεται και στην Εικόνα 1. Για λόγους απλότητας και ταχύτητας, μπορείτε να τοποθετείτε random τα πιόνια, έτσι ώστε να μπορείτε να ελέγχεται πιο γρήγορα τους κανόνες και να μην καθυστερείτε με την τοποθέτηση τους.

Στο ταμπλό υπάρχουν 8 τετράγωνα (τα κίτρινα) που ονομάζονται απαγορευμένες ζώνες. Δε μπορεί κάποιο πιόνι να παραμείνει σε αυτά τα τετράγωνα.



Εικόνα 1

¹ Στη παραλλαγή αυτή τα πιόνια είναι 30 αντί για 40.

Πιόνια του Παιχνιδιού

Τα συνολικά πιόνια για κάθε παίχτη είναι 30, ενώ τα μοναδικά πιόνια είναι 12, μιας και τα περισσότερα εμφανίζονται πάνω από 1 φορά:

Ονομασία	Κατάταξη	Διαθεσιμότητα
Δράκος (Dragon)	10	1
Μάγος (Mage)	9	1
Ιππότης (Knight)	8	2
Αναβάτης (Beast Rider)	7	3
Μάγισσα (Sorceress)	6	2
Θηρίο Λάβας (Lava Beast) (κόκκινο) / Yeti (μπλε)	5	2
Ξωτικό (Elf)	4	2
Νάνος (Dwarf)	3	5
Ανιχνευτής (Scout)	2	4
Εξολοθρευτής (Slayer)	1 (S)	1
Παγίδα	-	6
Σημαία	-	1

Κλασικοί κανόνες Stratego (90 %)

Στο κλασικό παιχνίδι, τα πιόνια με το μεγαλύτερο βαθμό μπορούν να αιχμαλωτίσουν τα πιόνια με το μικρότερο βαθμό. Τα μοναδικά πιόνια που έχουν κάποιες ειδικές δυνάμεις όσον αφορά είτε την κίνηση τους, είτε την επίθεση τους είναι οι ανιχνευτές, οι νάνοι και οι εξολοθρευτές. Παρόλο που χρησιμοποιούνται οι κλασικοί κανόνες, εσείς θα χρησιμοποιήσετε 30 πιόνια για κάθε παίχτη και το μέγεθος του ταμπλό θα είναι 8 γραμμές x 10 στήλες όπως στην Εικόνα 1.

Οι οδηγίες στα ελληνικά για το κλασικό παιχνίδι βρίσκονται εδώ

<http://www.strategohellas.gr/index.php/info/rules-and-pions>.

Ενέργειες που γίνονται στη σειρά κάθε παίχτη

Στη σειρά του κάθε παίκτης πρέπει:

- Είτε να κινήσει ένα από τα πιόνια του
- Είτε να κάνει επίθεση στον αντίπαλο.
- **Ο ανιχνευτής είναι το μόνο πiónι που μπορεί να κάνει κίνηση και επίθεση στον ίδιο γύρο**
- Σε περίπτωση που ένας παίκτης δε μπορεί να κάνει κάποια κίνηση ή επίθεση, το παιχνίδι τελειώνει και ανακηρύσσεται νικητής ο αντίπαλος

Επίσης κάθε φορά που παίζει ένας παίκτης θα πρέπει να φαίνεται η πίσω όψη της κάρτας του αντιπάλου όπως στις παρακάτω εικόνες. Στην μία περίπτωση παίζει ο παίκτης με τα μπλε και στην άλλη ο παίκτης με τα κόκκινα. **Και τους δύο παίκτες τους χειρίζεται ο χρήστης (δεν ζητείται να παίζει ο χρήστης εναντίον του υπολογιστή).**



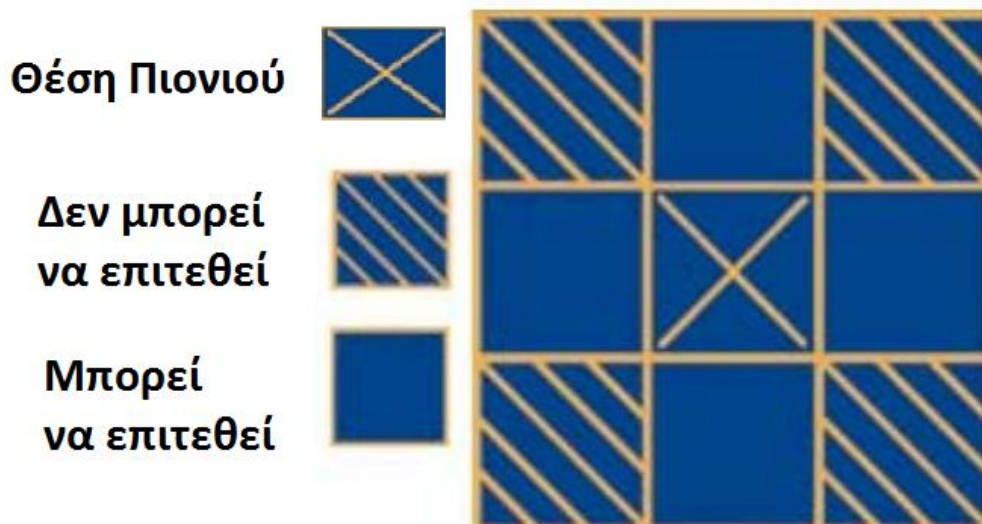
Κανόνες Κίνησης

Όλα τα πιόνια μπορούν να μετακινηθούν, εκτός από τη σημαία και τις παγίδες, τα οποία μένουν στην ίδια θέση που τοποθετούνται από την αρχή του παιχνιδιού. Τα υπόλοιπα πιόνια κινούνται σύμφωνα με τους παρακάτω κανόνες:

- Σε κάθε γύρο μόνο ένα πiónι μπορεί να μετακινηθεί
- Κάθε πiónι μετακινείται ένα τετράγωνο κάθε φορά
 - Τα πιόνια μετακινούνται μπροστά, πίσω, δεξιά ή αριστερά αλλά όχι διαγώνια
 - Εξαιρείται ο ανιχνευτής ο οποίος μπορεί να μετακινηθεί όσες κενές θέσεις θέλει μπροστά πίσω δεξιά ή αριστερά. Απλώς δε μπορεί να περάσει από τις απαγορευμένες ζώνες και να μετακινηθεί διαγώνια
- Δε γίνεται 2 πιόνια να βρίσκονται στην ίδια θέση.
- Δε γίνεται ένα πiónι να περάσει από τις απαγορευμένες ζώνες

Κανόνες Επίθεσης

Όλα τα πιόνια μπορούν να κάνουν επίθεση, εκτός από τη Σημαία ή την Παγίδα. Ένα πiónι μπορεί να επιτεθεί σε οποιοδήποτε πiónι αντιπάλου που βρίσκεται 1 τετράγωνο μπροστά, πίσω, δεξιά ή αριστερά όπως δείχνει η παρακάτω εικόνα



Όταν ένα πiónι επιτίθεται, τότε το πiónι με το χαμηλότερο βαθμό (κατάταξη) από τα 2 αφαιρείται από το ταμπλό.

- Αν το πiónι που αφαιρείται είναι του αμυνόμενου, τότε το πiónι του επιτιθέμενου παίρνει τη θέση του.
- Αν το πiónι που αφαιρείται είναι του επιτιθέμενου, τότε το πiónι του αμυνόμενου μένει στην ίδια θέση
- Αν τα πiónια έχουν τον ίδιο βαθμό, τότε αφαιρούνται και τα δύο

Ο Ανιχνευτής μπορεί να επιτεθεί από όποια απόσταση θέλει, αρκεί τα τετράγωνα μεταξύ αυτού και του αμυνόμενου πιονιού να είναι ελεύθερα (άδεια).

Κατάταξη κάθε πιονιού για την επίθεση

Οι κανόνες που ισχύουν για την επίθεση είναι οι εξής:

- Ο δράκος (10) κερδίζει το Μάγο (9) και όλα τα χαμηλότερα σε βαθμό πiónια. Ο Μάγος(9) κερδίζει τον Ιππότη(8) και όλα τα χαμηλότερα σε βαθμό πiónια. Αυτός ο κανόνας συνεχίζεται μέχρι το τελευταίο πiónι.
- Οποιοδήποτε πiónι επιτεθεί στην παγίδα χάνει αυτόματα, εκτός από τον νάνο (3), που εξουδετερώνει την παγίδα και την βγάζει εκτός παιχνιδιού, ενώ παίρνει και τη θέση που είχε η παγίδα αυτή.
- Η μικρότερη αξία ανήκει στο πiónι εξολοθρευτή (1 ή 5). Οποιοδήποτε πiónι επιτεθεί σε αυτό κερδίζει. Όμως αυτό το πiónι έχει μια εξαιρετική δυνατότητα. Αν και μόνο αν επιτεθεί πρώτο στο Δράκο (10), τον βγάζει εκτός παιχνιδιού και παίρνει τη θέση του στο ταμπλό. (ΠΡΟΣΟΧΗ: Αν επιτεθεί πρώτος ο Δράκος στον εξολοθρευτή τότε υπερσχύει ο Δράκος)
- Οποιοδήποτε πiónι μπορεί να επιτεθεί και να κατακτήσει (κερδίσει) τη Σημαία.

Διάσωση πιονιού που έχει βγει εκτός παιχνιδιού

Αν ένα πiónι φτάσει στην πρώτη σειρά του αντιπάλου, τότε μπορεί να επαναφέρει στο παιχνίδι ένα πiónι από τα δικά του που έχουν βγει εκτός παιχνιδιού και να το τοποθετήσει σε όποια θέση θέλει από τις 30 που ανήκουν στο στρατό του (3 πρώτες σειρές του παίκτη).

Περιορισμοί

- Οι ανιχνευτές δε μπορούν να κάνουν διάσωση
- Δε μπορεί να επαναφερθεί στο παιχνίδι μία παγίδα
- Κάθε παίκτης μπορεί να κάνει μέχρι δύο διασώσεις.
- Δε γίνεται το ίδιο πiónι να κάνει δύο διασώσεις.

Τέλος του παιχνιδιού

Νικητής του παιχνιδιού είναι ο παίκτης που θα καταφέρει να κατακτήσει τη Σημαία του αντιπάλου του. Νίκη επιτυγχάνεται επίσης και όταν ο αντίπαλος παίκτης είναι ανίκανος να κινηθεί επειδή είτε του έχουν απομείνει μόνο η Σημαία και οι Παγίδες ή επειδή οι Παγίδες εμποδίζουν την κίνηση των τελευταίων πιονιών που μπορούν να μετακινηθούν.

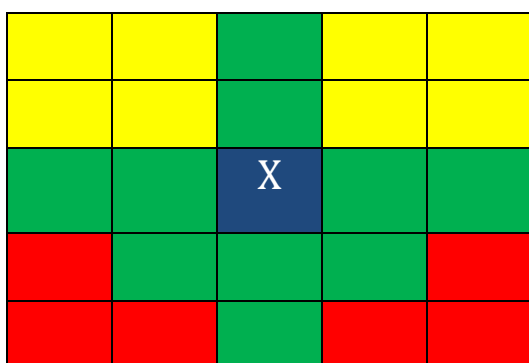
Ειδικοί κανόνες (15%)

Χωρίς τους ειδικούς κανόνες, μόνο τα πiónια με βαθμό 1-3 έχουν κάποιες ειδικές δυνατότητες. Όμως με τους παρακάτω κανόνες σας ζητείται για κάποια πiónια με βαθμό κατάταξης 4-10 να έχουν και αυτά κάποια ειδική δύναμη. **Τα πiónια με αυτές τις ειδικές δυνάμεις, μπορούν να κινούνται είτε όπως πριν, είτε να χρησιμοποιήσουν την ειδική δύναμη τους.**

Ειδικές δυνάμεις

Πέταγμα: Ο δράκος μπορεί να πετάξει πάνω από τα υπόλοιπα πiónια είτε οριζόντια είτε κάθετα και να προσγειωθεί στο πρώτο κενό τετράγωνο. Μπορεί να πετάξει και πάνω από τις απαγορευμένες ζώνες. Μόλις προσγειωθεί σε ένα κενό τετράγωνο, μπορεί να **κάνει αμέσως επίθεση** σε ένα γειτονικό αντίπαλο πiónι.

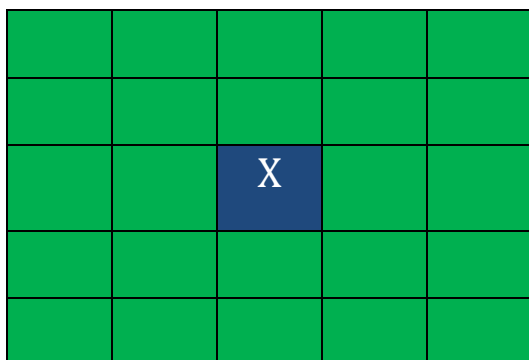
Ταχύτητα: Οι ιππότες και οι αναβάτες μπορούν να μετακινηθούν 2 ελεύθερα τετράγωνα, κάθετα ή οριζόντια (ή συνδυασμός αυτών των δύο) , ή μπορούν να επιτεθούν αμέσως σε πiónι που βρίσκεται 2 τετράγωνα μακριά. **Δεν γίνεται όμως να περάσουν από τις απαγορευμένες ζώνες.**



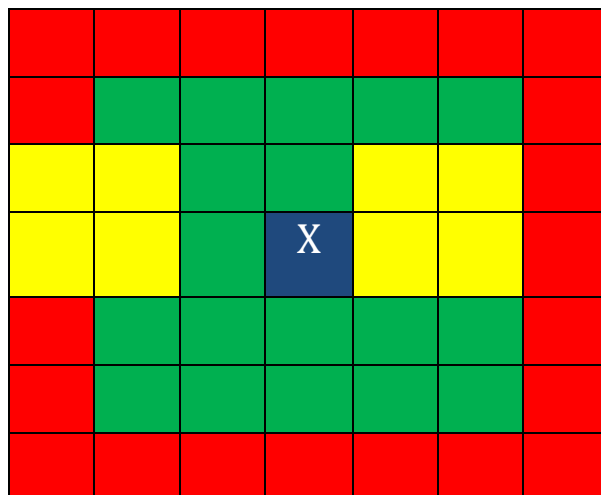
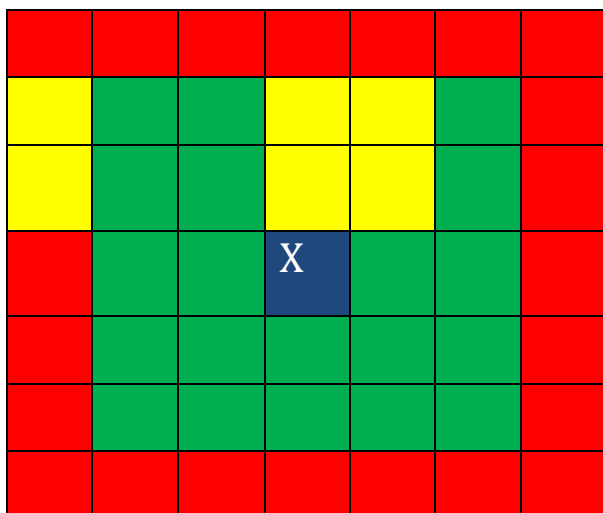
Στον παραπάνω πίνακα ο ιππότης ή ο αναβάτης που είναι στη θέση X μπορεί να μετακινηθεί στις πράσινες θέσεις. **Αν μετά από το πρώτο ελεύθερο τετράγωνο που θα μετακινηθεί ο ιππότης ή ο αναβάτης υπάρχει αντίπαλο γειτονικό πιόνι τότε μπορεί να του επιτεθεί αμέσως.** Οι κίτρινες γραμμές δείχνουν τις απαγορευμένες ζώνες

Επίθεση με Βέλη : Τα ξωτικά μπορούν να επιτεθούν με βέλη σε οποιοδήποτε πιόνι του αντιπάλου που βρίσκεται σε απόσταση 2 τετραγώνων μακριά, κάθετα, οριζόντια ή διαγώνια ή οποιοσδήποτε συνδυασμός αυτών ακόμα και αν παρεμβάλλεται η απαγορευμένη ζώνη . Αν το πιόνι είναι μικρότερο σε κατάταξη (1-3) τότε βγαίνει από το παιχνίδι. Αν είναι ίσο ή μεγαλύτερο σε κατάταξη (4-10) , παγίδα ή σημαία, τότε τίποτα δεν αλλάζει. Σε κάθε περίπτωση, το ξωτικό μένει στη θέση του.

Οι θέσεις που είναι 2 τετράγωνα μακριά είναι οι 24 γύρω από το πιόνι που φαίνονται στον παρακάτω πίνακα.



Αν παρεμβάλλεται η απαγορευμένη ζώνη (κίτρινα τετράγωνα) οι θέσεις που μπορεί να επιτεθεί το πιόνι στη θέση X φαίνονται με πράσινο χρώμα στους παρακάτω πίνακες. Το κόκκινο χρώμα δείχνει τις θέσεις που δε μπορεί να κάνει επίθεση.



Μαγεία (Bonus 5%) : Η μάγισσα μπορεί να μαγέψει τα αντίπαλα πιόνια έτσι ώστε να αλλάξουν στρατόπεδο. Οποιοδήποτε πιόνι που βρίσκεται σε απόσταση 2 τετραγώνων μπροστά, πίσω δεξιά, αριστερά ή διαγώνια (η οποιοδήποτε συνδυασμό και αν παρεμβάλλεται η απαγορευμένη ζώνη) μπορεί να επιλεγεί από τη μάγισσα γι αυτό το σκοπό. Αν το πιόνι που επιλέχθηκε είναι μικρότερο σε κατάταξη (δηλαδή 1-5) , τότε το πιόνι αλλάζει στρατόπεδο και παραμένει στο νέο στρατόπεδο μέχρι το τέλος του παιχνιδιού. Μπορεί να επιτεθεί σαν μέλος του νέου στρατοπέδου ενάντια στο παλιό. Αν το πιόνι είναι παγίδα, σημαία, ή έχει κατάταξη (6-10), τίποτα δεν αλλάζει στο παιχνίδι. **Οι θέσεις που μπορεί να επιτεθεί η μάγισσα (με τη δύναμη μαγείας) είναι ακριβώς οι ίδιες με τις θέσεις που περιγράφηκαν παραπάνω για την επίθεση με βέλη των ξωτικών.**

Πίνακας με τις ειδικές δυνάμεις κάθε πιονιού

Ονομασία	Δυνάμεις που υποστηρίζει	Πιόνια που κερδίζει	Πιόνια που χάνει
10: Δράκος (Dragon)	Πέταγμα	1(αν επιτεθεί πρώτος) 2,3,4,5,6,7,8,9, Σημαία	Παγίδα, 1 (αν είναι σε άμυνα)
9: Μάγος (Mage)		1,2,3,4,5,6,7,8, Σημαία	Παγίδα, 10
8: Ιππότης (Knight)	Ταχύτητα	1,2,3,4,5,6,7, Σημαία	Παγίδα, 10,9
7: Αναβάτης (Beast Rider)	Ταχύτητα	1,2,3,4,5,6, Σημαία	Παγίδα, 10,9,8
6: Μάγισσα (Sorceress)	Μαγεία (Bonus)	1,2,3,4,5, Σημαία	Παγίδα, 10,9,8,7
5: Lava Beast (κόκκινο) / Yeti (μπλε)		1,2,3,4, Σημαία	Παγίδα, 10,9,8,7,6
4: Ξωτικό (Elf)	Επίθεση με Βέλη	1,2,3, Σημαία	Παγίδα, 10,9,8,7,6,5
3: Νάνος (Dwarf)	Εξουδετέρωση Παγίδας	1,2, Παγίδα, Σημαία	10,9,8,7,6,5,4
2: Ανιχνευτής (Scout)	Μπορεί να μετακινηθεί παραπάνω από 1 ελεύθερα κελιά	1, Σημαία	Παγίδα, 10,9,8,7,6,5,4,3
1: Εξολοθρευτής (Slayer)	Μπορεί να εξουδετερώσει το Δράκο	10 (αν επιτεθεί πρώτος), Σημαία	10(αν είναι σε άμυνα) 9,8,7,6,5,4,3,2, Παγίδα
Παγίδα	-	1-2, 4-10	3
Σημαία	-	-	1-10

Υποδείξεις για Σχεδίαση/Υλοποίηση

Κατά τη διάρκεια της σχεδίασης του παιχνιδιού (και εν γένει οποιουδήποτε λογισμικού ή τεχνικού έργου), επιδιώκεται η αποσύνθεση του συστήματος σε μικρότερα τμήματα, με στόχο την ανάθεση σαφώς ορισμένων και καθορισμένων αρμοδιοτήτων σε κάθε τμήμα και την επικύρωση ότι όλα τα τμήματα μαζί επιτυγχάνουν τους σκοπούς του συστήματος. Επομένως, η σχεδίαση είναι μια διαδικασία επίλυσης και κατακερματισμού του αρχικού προβλήματος σε επιμέρους μικρότερα και ευκολότερα επιλύσιμα υπο-προβλήματα που θα ικανοποιούν τις λειτουργικές απαιτήσεις και θα υπόκεινται σε συγκεκριμένες αρχές καλής σχεδίασης.

Μια τέτοια αρχή αποτελεί η αποσύνδεση του μοντέλου (model), που περιγράφει τα δεδομένα, τη συμπεριφορά τους και το σύνολο των κανόνων που τα διέπει, από την απεικόνιση τους (view). Βασικός στόχος μιας τέτοιας αποσυνδεδεμένης προσέγγισης είναι η ελαχιστοποίηση των απαιτούμενων επεμβάσεων σε κώδικα που μπορούν να έχουν μελλοντικές αλλαγές είτε στο μοντέλο είτε στο τρόπο/μέσο απεικόνισης. Αυτό οδηγεί σε καλύτερη ποιότητα κώδικα, με μικρότερο κόστος συντήρησης, επέκτασης και επαναχρησιμοποίησης. Για να γίνει πιο κατανοητό αυτό θεωρήστε το ακόλουθο σενάριο. Υποθέστε ότι αρχικά έχετε σχεδιάσει μια εφαρμογή που για διάφορους λόγους εκτυπώνει τα αποτελέσματα της στην κονσόλα. Έχοντας ακολουθήσει μια τέτοια αρχιτεκτονική 'χαλαρής' σύνδεσης η μετάβαση σε ένα γραφικό παραθυρικό περιβάλλον γίνεται ομαλά, απλώς τροποποιώντας κατάλληλα (επεκτείνοντας) τις διαδικασίες εκείνες που ήταν υπεύθυνες για την αποτύπωση του μοντέλου στην κονσόλα.

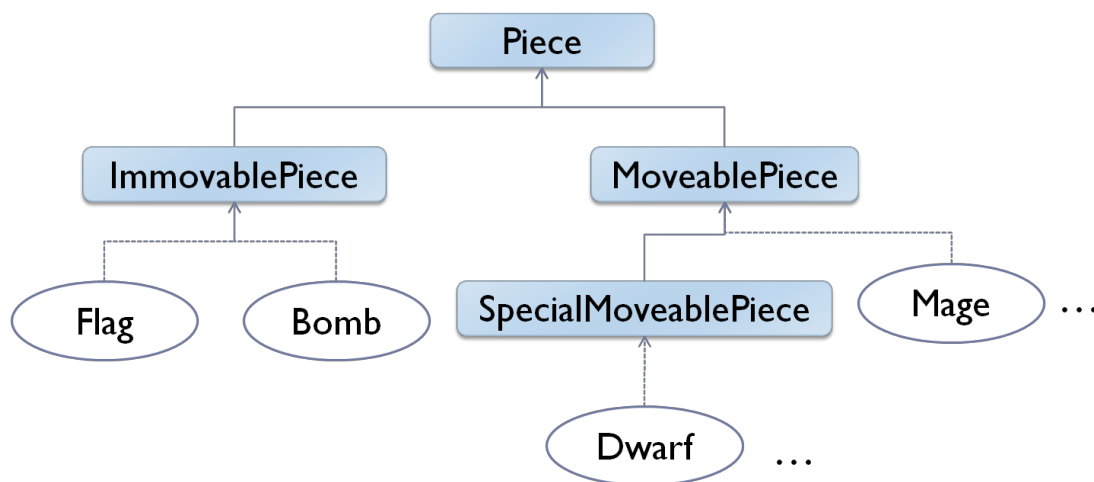
Στη συνέχεια ακολουθεί μια προτεινόμενη σχεδίαση της εφαρμογής, την οποία μπορείτε να ακολουθήσετε και να επεκτείνετε κατάλληλα. Η συγκεκριμένη στρατηγική σχεδίασης δεν είναι μοναδική, καθώς ένα αντικειμενοστραφές σύστημα λογισμικού μπορεί προφανώς να δομηθεί με πάρα πολλούς τρόπους. Αρκεί να αναλογιστεί κανείς τον αριθμό των δυνατών κλάσεων, των λειτουργιών που μπορούν να περιλαμβάνουν και των δυνατών συσχετίσεων μεταξύ τους. Συνεπώς, είστε ελεύθεροι να προτείνετε τη δική σας σχεδίαση εφόσον ακολουθεί τις αρχές που περιγράψαμε και είναι κατάλληλα τεκμηριωμένη.

MVC (Model-View-Controller) pattern

Βασική αρχή στην ανάπτυξη της παρούσας εργασίας θα πρέπει να είναι το MVC (Model-View-Controller) pattern. Σύμφωνα με αυτό το πρότυπο θα πρέπει να διαχωρίζεται η ανάπτυξη της γραφικής διεπαφής του παιχνιδιού (View) από τον πυρήνα του παιχνιδιού που περιέχει όλη την πληροφορία κατάστασης (Model) και από τον μηχανισμό διαχείρισης και ενημέρωσης των ενεργειών του παιχνιδιού με τη γραφική του απεικόνιση (Controller). Πιο συγκεκριμένα το Model μπορεί να θεωρηθεί ότι αποτελείται από οτιδήποτε σχετίζεται με τα δεδομένα του παιχνιδιού. Υπό αυτήν την έννοια τα «Πιόνια», το «ταμπλό», οι «Παίχτες» αποτελούν μέρος του μοντέλου του παιχνιδιού καθώς περιγράφουν τα εκάστοτε δεδομένα που καλείται να διαχειριστεί ο Controller του παιχνιδιού. Ο Controller επιφορτίζεται με τη διαχείριση της αλληλεπίδρασης της γραφικής διεπαφής με το μοντέλο. Το MVC pattern σας δίνει την δυνατότητα να διαχωρίσετε την υλοποίηση των παραπάνω και κατά συνέπεια καταστούν το πρόγραμμα σας ευκολότερο και στην ανάπτυξη και στην αποσφαλμάτωση (το κάθε συστατικό μπορεί να αναπτυχθεί και να δοκιμαστεί ξεχωριστά).

Προτεινόμενες Κλάσεις – Model

Μια βασική οντότητα/κλάση του παιχνιδιού είναι η abstract κλάση **Piece**. Η κλάση αυτή μοντελοποιεί ένα πιόνι του παιχνιδιού και ορίζει τα γνωρίσματα της (μεταβλητές στιγμιότυπων, π.χ. θέση πιονιού στο ταμπλό, βαθμός κατάταξης) και τις χρήσιμες μεθόδους της (πχ. μέθοδος attack). Μία πιθανή ιεραρχία είναι η ακόλουθη:



Για παράδειγμα η σημαία και η βόμβα δε μπορούν να μετακινηθούν και να επιτεθούν οπότε θα μπορούσαν να **είναι στιγμιότυπα** της κλάσης Immoveable Piece. Όλα τα υπόλοιπα που μπορούν να μετακινηθούν ανήκουν στην κατηγορία Moveable Piece που θα έχει μία μέθοδο που θα μπορούσε να επιστρέφει τις πιθανές θέσεις που μπορεί να πάει ένα πιόνι. Επίσης, από τα πιόνια που κινούνται για αυτά που έχουν ειδικές δυνάμεις μπορεί να υπάρξει εξειδικευμένη κλάση η SpecialMoveablePiece.

Θα μπορούσατε επίσης να σχεδιάσετε την λειτουργικότητα μιας κάρτας σαν διεπαφή (interface) και όχι απλώς σαν κλάση. Σε αυτή την περίπτωση, όλα τα διαφορετικά είδη καρτών θα πρέπει να υπακούν και να υλοποιούν το συμβόλαιο που έχει ορίσει η διεπαφή. Σε κάθε περίπτωση όμως θα πρέπει να συμπεριλάβετε στη σχεδίαση σας την τεχνική του πολυμορφισμού (π.χ. τη μέθοδο match). Η πολυμορφική αυτή μέθοδος είναι κοινή για όλες τις κλάσεις της ιεραρχίας, αλλά έχει πολλές διαφορετικές μορφές και υλοποιήσεις. Αρκεί να σκεφτούμε ότι η υλοποίηση της μεθόδου match διαφέρει από κάρτα σε κάρτα.

Μια σημαντική οντότητα του παιχνιδιού είναι η **Board**, η οποία προσομοιώνει το ταμπλό του παιχνιδιού. Το ταμπλό του παιχνιδιού αποτελείται από 80 τετράγωνα, όπου 8 από αυτά είναι απαγορευμένες ζώνες.

Μια από τις πιο στοιχειώδεις ίσως οντότητες του παιχνιδιού είναι αυτή του **Player**. Όπως υποδηλώνει και το όνομα της, η κλάση αυτή προσομοιώνει τον παίκτη του παιχνιδιού. Κάθε παίκτης διατηρεί ένα σύνολο από ενεργά πιόνια που βρίσκονται στο ταμπλό.

Controller

Ο Controller έχει πρόσβαση στα δεδομένα του παιχνιδιού και στην διεπαφή. Από την διεπαφή λαμβάνει πληροφορία για τις κινήσεις των παιχτών και με αυτήν θα πρέπει να ανανεώσει το μοντέλο του παιχνιδιού. Αντίστοιχα η ανανέωση του μοντέλου του παιχνιδιού θα πρέπει να χρησιμοποιείται από τον Controller για την ανανέωση της διεπαφής (ως αποτέλεσμα της κίνησης του παίχτη). Η κλάση αυτή αποτελεί τον «εγκέφαλο» του παιχνιδιού, καθώς ενορχηστρώνει όλες τις δυνατές λειτουργίες του και παρέχει προς τα έξω τις απαραίτητες μεθόδους ώστε να επικοινωνεί ο πυρήνας του παιχνιδιού (το σύνολο των κανόνων) με τη γραφική του αναπαράσταση. Συγκεκριμένα, ελέγχει για την αρχικοποίηση του, για την ομαλή διεξαγωγή του, την σειρά που θα παίξουν οι παίκτες, την ορθότητα των κινήσεων, τον τερματισμό του παιχνιδιού και την ανάδειξη του νικητή. Σκεφτείτε αναλυτικά τη ροή του παιχνιδιού και συμπεριλάβετε στη σχεδίαση σας όλες τις απαραίτητες λειτουργίες της κλάσης.

Γραφική Απεικόνιση

Η γραφική διεπαφή του παιχνιδιού πρέπει να παρουσιάζει το ταμπλό και να δίνει την δυνατότητα στους παίκτες να παίξουν ταυτόχρονα το παιχνίδι σε έναν υπολογιστή. Ένα σχεδιαστικό παράδειγμα υλοποίησης της εν λόγω διεπαφής παρουσιάζεται στην Εικόνα 2. Σε αυτήν την εικόνα κάθε πλευρά της διεπαφής αντιστοιχεί στα πιόνια ενός παίκτη ενώ όταν πατάει κάποιος ένα πιόνι, φαίνονται οι πιθανές θέσεις που μπορεί είτε να πάει είτε να επιτεθεί. Η γραφική απεικόνιση αυτή είναι ενδεικτική. Οποιαδήποτε διαφοροποίηση που δεν μειώνει την λειτουργικότητα του παιχνιδιού είναι αποδεκτή. Για παράδειγμα κάποιος θα μπορούσε να προσθέσει οτιδήποτε του αρέσει που να κάνει το παιχνίδι πιο ευχάριστο όπως πληροφορία για το ποια πιόνια έχουν φύγει για κάθε παίκτη, ονόματα παιχτών, ήχους κλπ.



Εικόνα 2

Προτεινόμενες Κλάσεις – View

Τέλος το View είναι η γραφική διεπαφή με την οποία αλληλεπιδρούν οι τελικοί χρήστες του παιχνιδιού. Όσον αφορά την διεπαφή η χρήση κληρονομικότητας μπορεί να διευκολύνει την ανάπτυξη μέσω της δημιουργίας νέων τύπων UI συστατικών τα οποία επεκτείνουν τη λειτουργικότητα των υπαρχόντων (για παράδειγμα η δημιουργία ενός τύπου PieceButton μπορεί να σας βοηθήσει στον χειρισμό των πιονιών ως Buttons αλλά με τα χαρακτηριστικά που εσείς επιθυμείτε να έχει ένα «Πιόνι» του παιχνιδιού). Μπορείτε να χρησιμοποιήσετε Grid Layout το μιας και το ταμπλό του παιχνιδιού ουσιαστικά είναι ένα grid διαστάσεων 8x10.

Βοηθητικά

Παρακάτω σας δίνονται κάποια ενδεικτικά links τα οποία μπορούν να σας βοηθήσουν στην υλοποίηση:

Επίσημες Οδηγίες στα Αγγλικά:

<http://www.hasbro.com/common/instruct/45068.pdf>

Σύντομο επεξηγηματικό υλικό για το παιχνίδι μπορεί να βρεθεί εδώ:

https://www.youtube.com/watch?v=o_1-w2Tlqac

Ενδεικτικό βοήθημα για το MVC (Model View Controller) μοντέλο σε Java

<http://www.newthinktank.com/2013/02/mvc-java-tutorial/>

Οι φωτογραφίες των καρτών έχουν ανέβει στο Elearn!

Καλή Εργασία