

Τεχνολογία πολυμέσων

Τελική Εργασία: Battleship

Ονοματεπώνυμο: Ελένη Παπαδοπούλου

Αριθμός μητρώου: 03117705

Σχεδιασμός υλοποίησης

Η εργασία υλοποιήθηκε στο το περιβάλλον του Netbeans IDE 12.0, με JDK 8 και με χρήση της JavaFX και του JavaFX Scene Builder για το σχεδιασμό της γραφικής διεπαφής.

Παρακάτω αναφέρονται και εξηγούνται οι κλάσεις του προγράμματος με τα πεδία και τις μεθόδους τους. Παραλείπεται η αναφορά για τις μεθόδους των setters και getters (ίδια μορφή για κάθε κλάση) καθώς και για τους constructor των κλάσεων (δίνουν τις κατάλληλες αρχικές τιμές στα πεδία των κλάσεων τους).

Κλάση Cell

<i>Cell</i>
boolean isshot
boolean isblank
Ship ship
//no methods

Τα αντικείμενα της κλάσης αυτής είναι τα **κελιά** που απαρτίζουν το **ταμπλό κάθε παίκτη**. Για κάθε κελί αποθηκεύεται η πληροφορία για το εάν έχει ξανά κτυπηθεί (issht), η πληροφορία για το εάν είναι κενό, δηλαδή χωρίς πλοίο (isblank) και η πληροφορία για το πλοίο που μπορεί να έχει (ship).

Κλάση Player

<i>Player</i>
cell [][] Board int Points int unsank_ships float hit_perc int times_played int times_hit boolean played_first int shoot_row, shoot_column int last_hit_row, last_hit_column
hitcell(int row, int column) enemyPlays() int hitHistory(int row, int column) int firstX(int left,int right, int up, int down, x)

Τα αντικείμενα της κλάσης αυτής είναι οι **παίκτες του παιχνιδιού**.

Για κάθε παίκτη αποθηκεύονται το ταμπλό του αντιπάλου του (Board)*, οι πόντοι του (Points), τα αβύθιστα πλοία (unsank_ships), το ποσοστό επιτυχών βολών (hit_perc), οι φορές που έπαιξε (times_played), οι φορές που ήταν εύστοχη η βολή του (times_hit), η πληροφορία για το αν παίζει πρώτος (played_first) .

* Επιλεκτικέ ο παίκτης να κρατά πληροφορία για το ταμπλό του αντιπάλου του αφού με βάση αυτό ανανεώνονται κάποια πεδία του (Points, unsank_ships κτλ).

Σε αυτή τη κλάση γίνεται η εκτέλεση βολής από κάποιο παίκτη με χρήση της μεθόδου **hitcell**.

Η μέθοδος αυτή ανανεώνει τη ταμπλό κάθε παίκτη με βάση το αποτέλεσμα της βολής.

Η μέθοδος αυτή ελέγχει και την περίπτωση ο παίκτης να επιλέξει κάποια βολή η οποία έχει ξαναγίνει, με τη χρήση του alreadyshotException.

Τέλος σε αυτή τη κλάση γίνεται και η υλοποίηση του αλγόριθμου με τον οποίο ο υπολογιστής (enemy) επιλέγει τις συντεταγμένες της βολής που θα εκτελέσει. Για το σκοπό αυτό χρησιμοποιούνται τα πεδία shoot_row, shoot_column, (επιλεγόμενες συντεταγμένες του enemy) και last_hit_row, last_hit_column (συντεταγμένες προηγούμενης εύστοχης βολής) καθώς και οι μέθοδοι hitHistory και firstX.

Περιγραφή αλγορίθμου για επιλογή συντεταγμένων βολής υπολογιστή:

Αρχικά επιλέγονται τυχαία νέες συντεταγμένες στο ταμπλό για κάθε βολή του υπολογιστή.

Όταν καταφέρει (τυχαία) να κτυπήσει ένα κομμάτι κάποιου πλοίου, ενημερώνονται οι συντεταγμένες προηγούμενης εύστοχης βολής και ξεκινά η διαδικασία εύρεσης του υπόλοιπου πλοίου μέχρι αυτό να βυθιστεί.

Στην αρχή σημειώνονται τα γειτονικά κελία της επιτυχημένης βολής όπως φαίνεται παρακάτω

	up(2)	
left(1)	last hit (2)	right(3)
	down(4)	

Έπειτα εφόσον κανένα από αυτά δεν έχει ξανακτυπηθεί, επιλέγονται οι συντεταγμένες του αριστερού κελιού (left) ως οι επιλεγόμενες συντεταγμένες βολής του enemy.

Αν οι επιλεγόμενες συντεταγμένες δεν ήταν εύστοχες επιλέγονται με τη σειρά το κελί right, μετά το up και τέλος το down, μέχρι να βρεθεί εύστοχη βολή.

Όταν βρεθεί η εύστοχη βολή ενημερώνονται οι συντεταγμένες προηγούμενης εύστοχης βολής και τρέχει ο ίδιος αλγόριθμος για τις νέες συντεταγμένες.

Στην περίπτωση που δύο συνεχόμενα κελιά έχουν κτυπηθεί, έχει φανερωθεί ο προσανατολισμός του πλοίου και ο enemy εκτελεί βολές προς αυτή τη κατεύθυνση μέχρι να ενημερωθεί ότι το πλοίο βυθίστηκε.

unknown(0)	last hit (2)	last hit (2)	unknown(0)

Εάν φτάσει κάποιο όριο του ταμπλό ή φτάσει σε κάποιο κελί το οποίο έχει ξανά επισκεφτεί και το πλοίο δεν έχει βυθιστεί ψάχνει το υπόλοιπο κομμάτι του πλοίου προς την αντίθετη φορά.

unknown(0)	last hit (2)	last hit (2)	no hit(1)

Αφού ενημερωθεί ο αντίπαλος ότι το πλοίο έχει βυθιστεί, ξεκινά τις τυχαίες βολές και επαναλαμβάνεται η διαδικασία.

Στα σχεδιαγράμματα φαίνονται οι περιπτώσεις που περιγράφονται παραπάνω. Οι αριθμοί στις παρενθέσεις είναι οι κωδικοί με τους οποίους αναπαρίσταται στον κώδικα η κάθε περίπτωση.

Κλάση Ship

<i>Ship</i>
int id String name int length int shoot_points int sink_points int count_hits boolean issank int Hit()

Για τη κλάση αυτή κάθε public μέθοδος είναι τεκμηριωμένη σύμφωνα με τις προδιαγραφές του εργαλείου **Javadoc**.

Τα αντικείμενα της κλάσης αυτής είναι τα **πλοία** του παιχνιδιού.

Για κάθε πλοίο αποθηκεύεται το αναγνωριστικό του (id), το όνομα του (name), το μήκος του (length), οι πόντοι που δίνει όταν κτυπηθεί (shoot_points), οι πόντοι που δίνει όταν βυθιστεί (sink_points), οι φορές που έχει κτυπηθεί (count_hits) και η πληροφορία για το αν είναι βυθισμένο ή όχι (issank).

Σε αυτή τη κλάση γίνεται η υλοποίηση μίας μεθόδου (Hit()) η οποία επιστρέφει τους συνολικούς πόντους που δίνει η εύστοχη βολή ενός πλοίου (με βάση τη κατάσταση του) και ενημερώνει εάν το πλοίο έχει βυθιστεί. Τη μέθοδο αυτή καλεί η μέθοδος hitcell της κλάσης Player για να ενημερώσει σωστά τους πόντους του κάθε παίκτη και την κατάσταση των πλοίων στο ταμπλό (issank).

Κλάση Initialize Board

<i>Inititalize_Boad</i>
//no variables
Cell[][] initBoard() Cell[][] readFile(String file_name) Cell[][] addShiptoBoard(Cell[][] Board, int ship_id, int row, int column, int orientation) Ship createShip(int id)

Η κλάση αυτή χρησιμοποιείται για την **αρχικοποίηση των ταμπλό** των δύο παικτών με τα πλοία από τα αρχεία.

Για αυτή τη διαδικασία γίνεται χρήση τεσσάρων μεθόδων

Η `initBoard` κάνει την αρχικοποίηση κάθε ταμπλό με ένα διδιάστατο πίνακα 10x10 με στοιχεία τύπου `Cell`.

Η `readFile` διαβάζει από το επιθυμητό αρχείο και καλεί την `addShiptoBoard` με τα σωστά ορίσματα.

Η `addShiptoBoard` προσθέτει τα πλοία στα σωστά κελία κάθε ταμπλό, με τη βοήθεια της `createShip`. Επίσης κάνει τους σωστούς ελέγχους και ρίχνει τα σωστά exceptions στις περιπτώσεις που τα αρχεία εισόδου περιγράφουν λανθασμένα ταμπλό (`OversizeException`, `OverlapTilesException`, `AdjacentTilesException`, `InvalidCountException`).

Η `createShip` επιστρέφει ένα `Ship` αντικείμενο, το οποίο περιέχει τις σωστές πληροφορίες του πλοίου με βάση το αναγνωριστικό του.

Κλάσεις για εξαιρέσεις

`OversizeException`

`OverlapTilesException`

`AdjacentTilesException`

`InvalidCountException`

`alreadyshotException`

Οι κλάσεις για τις εξαιρέσεις, τα πιθανά σφάλματα στο πρόγραμμα.

Κλάση BattleshipUI

Σε αυτή τη κλάση υπάρχει η **κύρια μέθοδος** του προγράμματος. Η κλάση αυτή ενεργοποιεί την γραφική διεπαφή και φορτώνει το FXML αρχείο που δημιουργήθηκε από το Scene Builder.

Κλάση FXML LayoutController

<i>FXML_LayoutController</i>
<code>Cell[] boardplayer</code> <code>Cell[] boardenemy</code> <code>Player player</code> <code>Player enemy</code> <code>Int round</code> <code>Int maxrounds</code>
<code>StackPane[] stackPanes_enemy</code> <code>StackPane[] stackPanes_player</code> ... labels, TextFields, Buttons, GridPanes ...
<code>initPlayer()</code> <code>initEnemy()</code> <code>RefreshPlayer()</code> <code>RefreshEnemy()</code> <code>start(event)</code> <code>hitcell(event)</code> <code>initialize()</code>

Η κλάση αυτή είναι υπεύθυνη για την **ροή του παιχνιδιού** και για τον παράλληλο **έλεγχο** και **ενημέρωση** της γραφικής διεπαφής.

Μέθοδος start

Το παιχνίδι ξεκινά με το πάτημα του κουμπιού start_button, το οποίο έχει μέθοδο ενεργοποίησης την start. Σε αυτή τη μέθοδο δημιουργούνται τα ταμπλό τόσο του παίκτη όσο και του αντιπάλου, τα αντικείμενα για τον παίκτη και τον αντίπαλο και ενημερώνεται η γραφική διεπαφή κατάλληλα (κλήση των μεθόδων initPlayer, initEnemy). Εδώ γίνεται και η τυχαία επιλογή για το ποιος ξεκινά και ενημερώνεται με μήνυμα η γραφική διεπαφή.

Μέθοδος hitcell

Η μέθοδος αυτή καλείται με κάθε πάτημα του κουμπιού hit_button. Σε αυτή τη μέθοδο εκτελούν τις βολές τους παίκτης και αντίπαλος, ενημερώνονται οι γύροι και γίνεται έλεγχος για το τέλος του παιχνιδιού. Αφού παίξει κάποιος παίκτης ενημερώνεται κατάλληλα η γραφική διεπαφή με κλήση των μεθόδων RefreshPlayer, RefreshEnemy.

Στο τέλος του παιχνιδιού ανακοινώνεται ο νικητής με μήνυμα στην οθόνη.

Εξήγηση - Παραδοχές γραφικής διεπαφής

Η γραφική διεπαφή έχει την παρακάτω μορφή

Medialab Battleship

Player		Enemy	
Points:		Points:	
Unsunk ships:		Unsunk ships:	
Hit perc(%):		Hit perc(%):	

0

1

2

3

4

5

6

7

8

9

0

1

2

3

4

5

6

7

8

9

NEW GAME

hit cell coordinates

HIT

player ID

enemy ID

Enemy Shoots:

Round:

Χωρίζεται σε **3** κομμάτια από πάνω προς τα κάτω.

Στο **πρώτο κομμάτι** φαίνονται και για τους δύο παίκτες οι πόντοι τους, τα αβύθιστα τους πλοία και το ποσοστό επιτυχίας. Επίσης με την εκκίνηση του παιχνιδιού φαίνεται κάτω από το όνομα του παίκτη που παίζει πρώτος σχετικό μήνυμα ενημέρωσης.

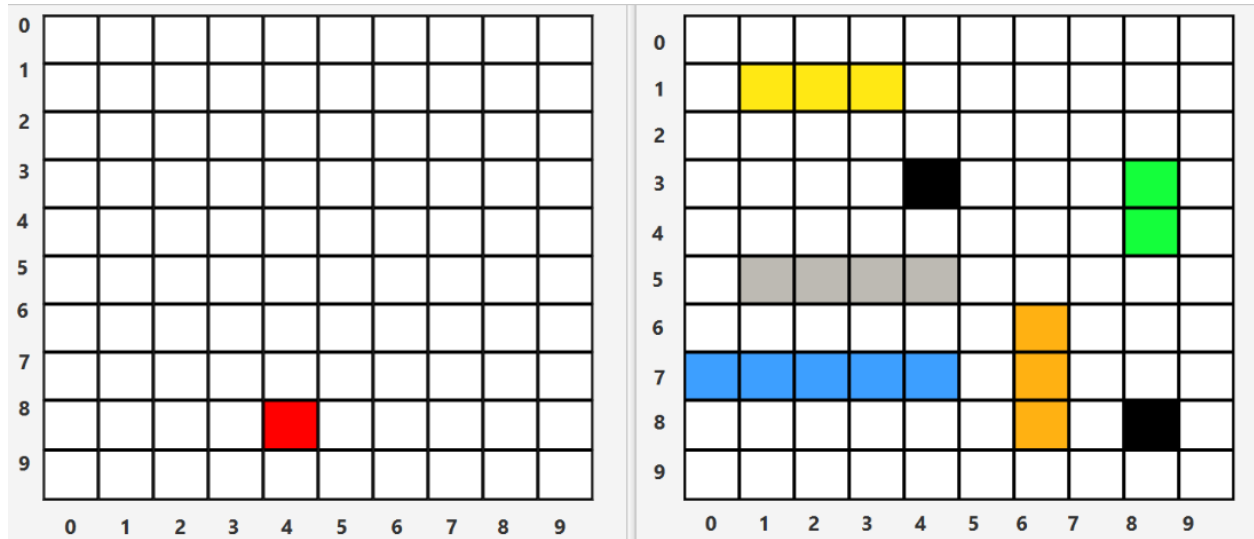
Player	Points:	0	Enemy	Points:	0
YOU PLAY FIRST	Unsunk ships:	5		Unsunk ships:	5
	Hit perc(%):	0.0		Hit perc(%):	0.0

Player	Points:	0	Enemy	Points:	0
	Unsunk ships:	5	ENEMY PLAYS FIRST	Unsunk ships:	5
	Hit perc(%):	0.0		Hit perc(%):	0.0

Το **δεύτερο κομμάτι** αποτελείται από τα ταμπλό των δύο παικτών.

Το αριστερά ταμπλό (κάτω από τον Player) είναι το ταμπλό του Enemy, τα πλοία του οποίου είναι κρυφά στον παίκτη. Το δεξιά ταμπλό (κάτω από τον Enemy) είναι το ταμπλό του Player, τα πλοία του οποίου είναι φανερά στον παίκτη και παρουσιάζονται με τα αντίστοιχα χρώματα (μπλε → 1, γκρι → 2, πορτοκαλί → 3, κίτρινο → 4, πράσινο → 5).

Τα δύο ταμπλό ενημερώνονται σε κάθε γύρο με τις βολές των παικτών (επιτυχημένη βολή → κόκκινο χρώμα, αποτυχημένη βολή → μαύρο χρώμα).



Στο **τρίτο κομμάτι** εισάγονται οι επιλογές του χρήστη.

Αρχικά υπάρχει η επιλογή για σενάριο παιχνιδιού τόσο για τον enemy όσο και για τον player. Εδώ εισάγεται ένας αριθμός ο οποίος αντιστοιχεί στο ID του αρχείου που περιγράφει την αρχική κατάσταση των πλοίων και για τους δύο παίκτες.

Εάν εισαχθεί αρχείο με λάθος τοποθέτηση πλοίων εντοπίζεται η αντίστοιχη εξαίρεση και ενημερώνεται η γραφική διεπαφή με μήνυμα όπως φαίνεται παρακάτω.

Στο φάκελο με τα αρχεία για τα αναγνωριστικά 4, 5 και για τους δύο παίκτες δημιουργήθηκαν αρχεία που προκαλούν εξαιρέσεις.

NEW GAME	hit cell coordinates			HIT
player ID	1	Enemy File: INVALID COUNT!		
enemy ID	4			

NEW GAME	hit cell coordinates			HIT
player ID	1	Enemy File: OVERLAP TILES!		
enemy ID	5			

NEW GAME	hit cell coordinates			HIT
player ID	4	Player File: OVERSIZE!		
enemy ID	1			

NEW GAME	hit cell coordinates			HIT
player ID	5	Player File: ADJACENT TILES!		
enemy ID	1			

Επίσης υπάρχει η φόρμα συμπλήρωσης συντεταγμένων για κάθε βολή του παίκτη. Στη περίπτωση εισαγωγής λανθασμένων συντεταγμένων ή συντεταγμένων που έχουν ήδη κτυπηθεί ενημερώνεται με κατάλληλο μήνυμα όπως φαίνεται παρακάτω

hit cell coordinates
player ID Wrong value, try again
enemy ID

hit cell coordinates
player ID Already shot, try again
enemy ID

Τέλος παρουσιάζονται οι συντεταγμένες βολής που επέλεξε ο αντίπαλος και ο αριθμός γύρου. Όταν τελειώσει το παιχνίδι εμφανίζεται μήνυμα ενημέρωσης για τον νικητή.

hit cell coordinates
player ID
enemy ID

Enemy Shoots: 8 4
Round: 40
ENEMY WON!

Στιγμιότυπο παιχνιδιού:

Player	Points:	5200	Enemy	Points:	2750
YOU PLAY FIRST	Unsunk ships:	0		Unsunk ships:	4
	Hit perc(%):	100.0		Hit perc(%):	29.41

hit cell coordinates
player ID
enemy ID

Enemy Shoots: 5 7
Round: 17
YOU WON!