

ΟΜΑΔΙΚΗ ΕΡΓΑΣΙΑ(PROJECT)ΣΤΟΝ

ΟΝΤΟΚΕΝΤΡΙΚΟ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ 2017-2018

1. Αδριανός Σύρρος Α.Μ.: 1051974 email: asyrros@ceid.upatras.gr
2. Γιώργος Σφατίδης Α.Μ.:1047175 email: sfatidis@ceid.upatras.gr
3. Ιωάννης Χριστοδούλου Α.Μ.:1047104 email : christodoulou@ceid.upatras.gr



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΑΤΡΩΝ

UNIVERSITY OF PATRAS

ΕΡΓΑΣΙΑ 1 (JAVA):

Μας ζητείται να υλοποιήσουμε το κλασσικό παιχνίδι <<Ναυμαχία>> όπου ο χρήστης θα παίζει με αντίπαλο τον υπολογιστή σε γλωσσά προγραμματισμού Java.

Αναλυτικότερα ο κώδικας αποτελείται από τις εξής κλάσεις:

Κλάση Tiles: Σε αυτή γίνεται η δήλωση των enums που ζητούνται(SEA,SHIP,MISS,HIT) και αναφέρονται στις 4 διαφορετικές περιπτώσεις που μπορούν να πάρουν τα κελιά του πίνακα.

Κλάση Tiles2: Περιλαμβάνει τις μεθόδους fill() η όποια γεμίζει τον πίνακα με το αρχικό σύμβολο SEA ~ και draw() και PCdraw()(για τον υπολογιστή αντίστοιχα)που τυπώνει το σύμβολο του κελιού και γενικά τον πίνακα.

Κλάση Ship:Είναι μια αφηρημένη κλάση η οποία αναλαμβάνει την δήλωση των πλοίων και την τοποθέτηση τους στον πίνακα του παίκτη σύμφωνα με τις πληροφορίες που δίνει ο ίδιος δηλαδή το μέγεθος(size),τη γραμμή(row) και τη στήλη(col) απ' όπου θέλει ο παίκτης να ξεκινά, καθώς και την κατεύθυνση του(0(κάθετη)ή 1(οριζόντια)).Τα παραπάνω γίνονται με την χρήση της μεθόδου PlaceShip() η οποία περιλαμβάνει και τις απαραίτητες εξαιρέσεις που θα αναφερθούν και παρακάτω. Μετά από την τοποθέτηση κάθε πλοίου εμφανίζεται ο πίνακας με τις νέες αλλαγές του. Παράδειγμα κώδικα:

```
Do you want your ships placed random? (1/0)
1
Place your Ship!
Give a row:
0
Give a column:
0
Horizontal or Vertical(1/0)? :
0
S ~ ~ ~ ~ ~ ~ ~ ~
S ~ ~ ~ ~ ~ ~ ~ ~
S ~ ~ ~ ~ ~ ~ ~ ~
S ~ ~ ~ ~ ~ ~ ~ ~
S ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~
Place your Ship!
Give a row:
```

Κλάση Board:Η οποία χρησιμοποιεί τις μεθόδους της Tiles2 που αφορούν τον πίνακα και δηλώνει δύο λογικές μεταβλητές την ShipsSunk και την PCShipsSunk.

Κλάση Player:Με τα στοιχεία και τις διαθέσιμες ενέργειες των παικτών κατά την διάρκεια του παιχνιδιού, δηλαδή την ρίψη «ορπιλών» στον αντίπαλο παίκτη(Player,Fire,PCFire) καθώς και την μέθοδο getStats() η οποία υπολογίζει τα στατιστικά του παίκτη. Επίσης περιέχει την

μέθοδο PlaceAllShips() για την τυχαία τοποθέτηση των πλοίων στον πίνακα.
Παράδειγμα κώδικα:

```
Choose row for the shot
0
Choose column for the shot
0
|-----HIT-----
-----MISS-----
-----PLAYER-----
S ~ ~ S ~ ~ ~ ~ S ~
S ~ ~ S ~ ~ ~ ~ S ~
S ~ ~ S ~ ~ ~ ~ S ~
S ~ ~ ~ ~ ~ ~ ~ ~
S ~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~ 0
S S ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~
S S S S ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~

-----PC-----
X S S ~ ~ ~ S ~ S ~
~ ~ ~ ~ ~ ~ S ~ S ~
~ ~ ~ ~ ~ ~ ~ S ~
~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~
S S S S ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~
S S S S S ~ ~ ~ ~
~ ~ ~ ~ ~ ~ ~ ~
1
```

Κλάση Game: Αποτελεί την κεντρική κλάση του παιχνιδιού. Περιέχει την main στην οποία αρχικοποιούνται οι πίνακες των δυο παικτών. Εμφανίζει μήνυμα στον παίκτη για την προτίμηση του σχετικά με την τοποθέτηση των πλοίων του στον πίνακα (δηλαδή αν θέλει να τα τοποθετήσει ο ίδιος ή αν θέλει να τοποθετηθούν τυχαία από τον υπολογιστή) και ανάλογα με την επιλογή του καλεί τις κατάλληλες μεθόδους (player.placeShip, Board.drawboards) με τα κατάλληλα μηνύματα σε περίπτωση λάθους ("Wrong input! Please Insert valid input (Y/N or y/n):"). Ύστερα καλεί τις κατάλληλες μεθόδους για την έναρξη του παιχνιδιού (player.Fire, player.PCFire). Επίσης περιέχει τις παραμέτρους για την λήξη ενός παιχνιδιού (το παιχνίδι τελειώνει όταν ένας από τους δύο παίκτες βυθίσει όλα τα πλοία του αντίπαλού του ή όταν οι δύο παίκτες εξαντλήσουν τις τορπίλες τους, έχουν 25 τορπίλες διαθέσιμες).

Κλάσεις Carrier, Cruiser, Destroyer, Submarine, Battleship: Αποτελούν κλάσεις που αναφέρονται στην Ship. Επειδή η Ship είναι μια αφηρημένη κλάση η παραγωγή αντικειμένων από αυτή δεν είναι δυνατή, συνεπώς η δημιουργία κλάσεων που κληρονομούν την Ship ήταν απαραίτητη για να είναι δυνατή η χρησιμοποίηση των μεθόδων της με τη χρήση της super.

Κλάση `AdjacentTilesException`: Είναι η εξαίρεση που αφορά την τοποθέτηση πλοίων σε γειτονικά κελιά. Ο έλεγχος της γίνεται μέσα στην κλάση `Ship` κατά την τοποθέτηση των πλοίων. Ελέγχει τα γειτονικά κελιά του πλοίου που τοποθετούμε. Παράδειγμα κώδικα:

[illegible]

Κλάση NewException: Αποτρέπει την τοποθέτηση πλοίων στα ίδια κελιά. Ο έλεγχος της γίνεται μέσα στην κλάση Ship κατά την τοποθέτηση των πλοίων. Ελέγχει τα κελιά τα οποία καταλαμβάνει το καινούριο πλοίο έτσι ώστε να μην συμπίπτουν με αυτά ενός άλλου πλοίου. Παράδειγμα κώδικα:

Do you want your ships placed random? (1/0)

1

Place your Ship!

Give a row:

0

Give a column:

0

Horizontal or Vertical(1/0)? :

0

S ~ ~ ~ ~ ~ ~ ~ ~ ~

S ~ ~ ~ ~ ~ ~ ~ ~ ~

S ~ ~ ~ ~ ~ ~ ~ ~ ~

S ~ ~ ~ ~ ~ ~ ~ ~ ~

S ~ ~ ~ ~ ~ ~ ~ ~ ~

~ ~ ~ ~ ~ ~ ~ ~ ~

~ ~ ~ ~ ~ ~ ~ ~ ~

~ ~ ~ ~ ~ ~ ~ ~ ~

~ ~ ~ ~ ~ ~ ~ ~ ~

~ ~ ~ ~ ~ ~ ~ ~ ~

Place your Ship!

Give a row:

1

Give a column:

0

Horizontal or Vertical(1/0)? :

1

Warning: Ships start fromt the same point!

Place your Ship!

Give a row:

Κλάση `OverlapTilesException`: Αποτρέπει την τοποθέτηση πλοίων με ίδιο σημείο έναρξης. Ο έλεγχος της γίνεται μέσα στην κλάση `Ship` κατά την τοποθέτηση των πλοίων. Ελέγχει αν το κελί από το οποίο αρχίζει το καινούργιο πλοίο είναι κενό άρα και διαθέσιμο. Παράδειγμα κώδικα:

Do you want your ships placed random? (1/0)

1

Place your Ship!

Give a row:

0

Give a column:

0

Horizontal or Vertical(1/0)? :

0

S ~ ~ ~ ~ ~ ~ ~ ~

S ~ ~ ~ ~ ~ ~ ~ ~

S ~ ~ ~ ~ ~ ~ ~ ~

S ~ ~ ~ ~ ~ ~ ~ ~

S ~ ~ ~ ~ ~ ~ ~ ~

~ ~ ~ ~ ~ ~ ~ ~ ~ ~

~ ~ ~ ~ ~ ~ ~ ~ ~ ~

~ ~ ~ ~ ~ ~ ~ ~ ~ ~

~ ~ ~ ~ ~ ~ ~ ~ ~ ~

~ ~ ~ ~ ~ ~ ~ ~ ~ ~

Place your Ship!

Give a row:

0

Give a column:

0

Horizontal or Vertical(1/0)? :

1

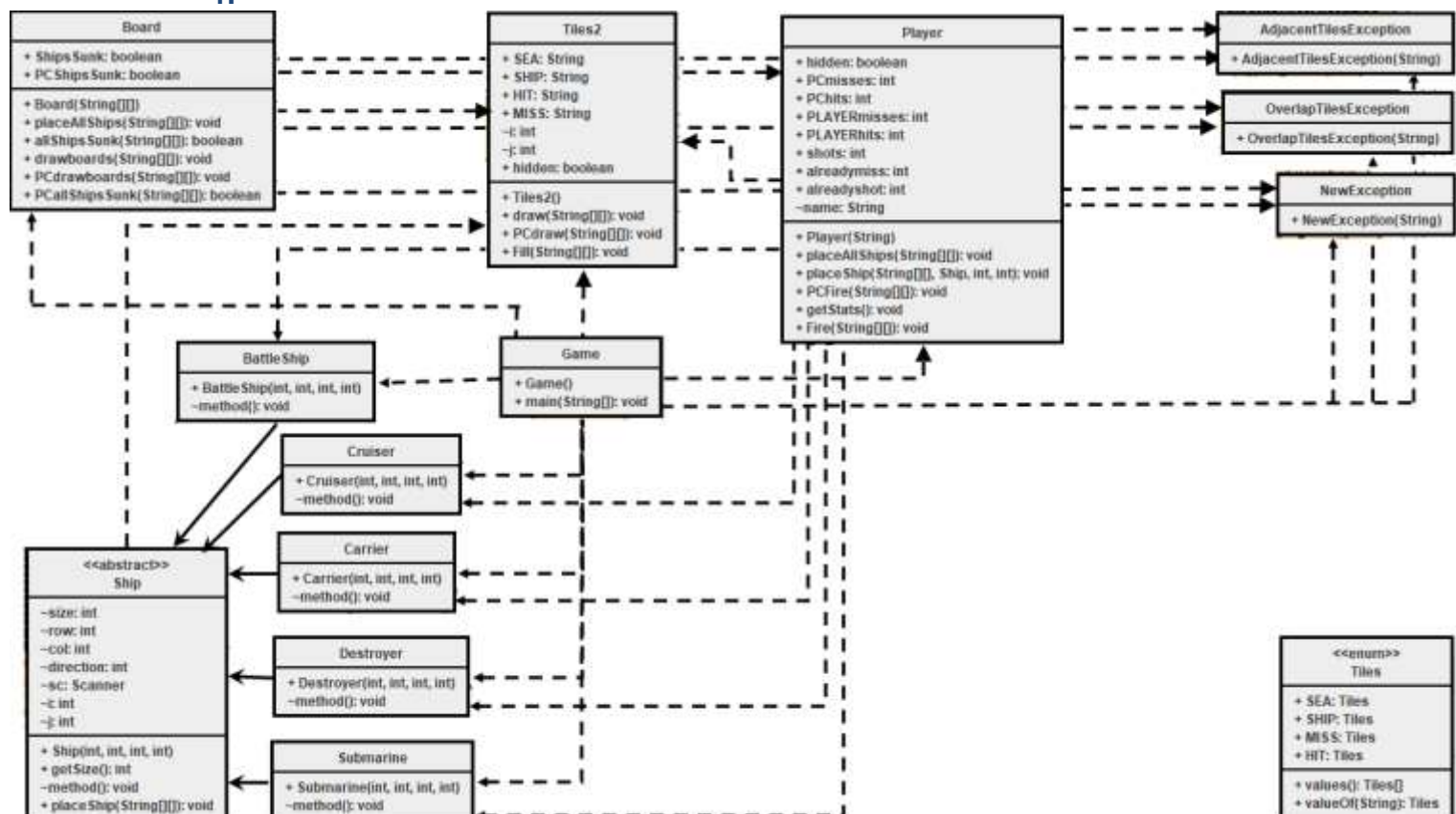
Warning: Ships start fromt the same point!

Place your Ship!

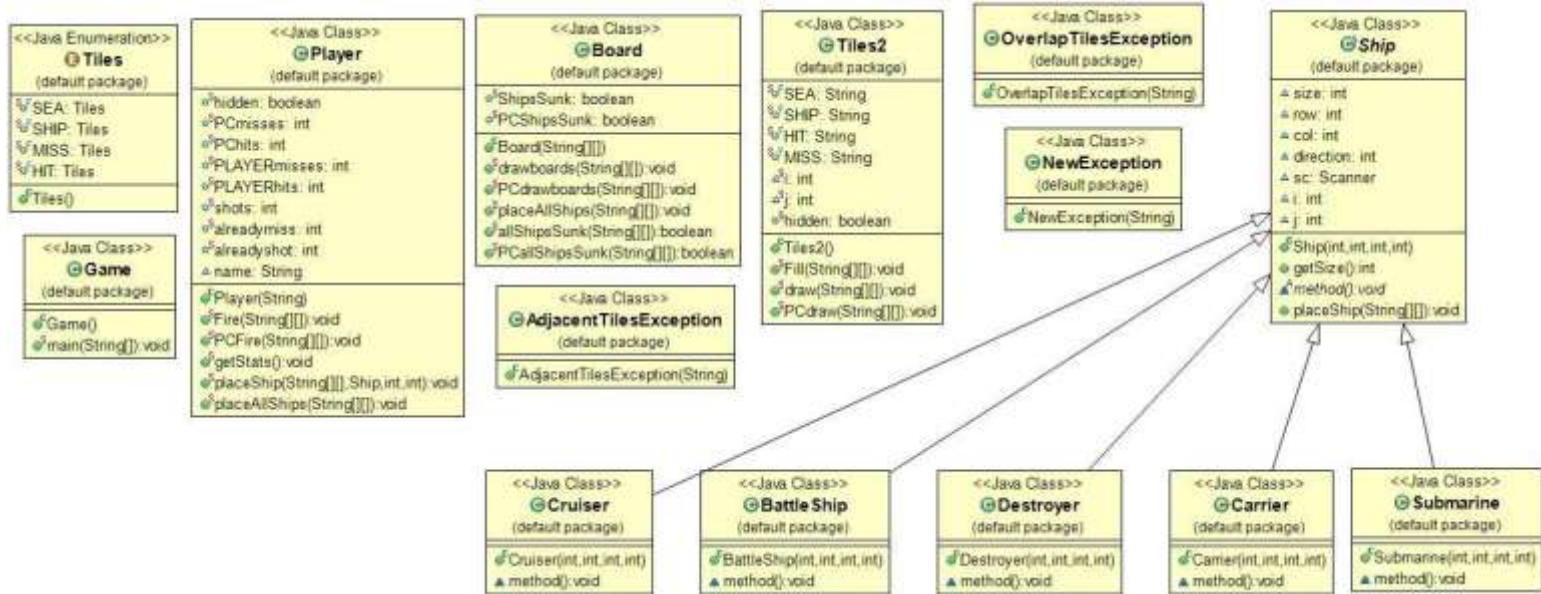
Give a row:

Διαγράμματα Κλάσεων(UML):

1.



2.



3.

