

HUNGER GAMES

MAY THE ODDS BE EVER IN YOUR FAVOR

ΜΕΡΟΣ Α

Δομές Δεδομένων

Ακαδημαϊκό έτος
2019 - 2020

Φωτεινή Σαββίδου
(9657, sfoteini@ece.auth.gr)

Πίνακας Περιεχομένων

Περιγραφή του παιχνιδιού Hunger Games	3
Υλοποίηση του παιχνιδιού σε Java	4
Η κλάση Weapon	4
Η κλάση Food	4
Η κλάση Trap	5
Η κλάση Board	5
Η κλάση Player	8
Η κλάση Game	10

Περιγραφή του παιχνιδιού Hunger Games

Το Hunger Games είναι ένα τηλεοπτικό παιχνίδι επιβίωσης. Οι παίκτες διαγωνίζονται σε ένα νησί του οποίου η διάμετρος μειώνεται με την πάροδο του χρόνου. Νικητής του παιχνιδιού αναδεικνύεται ο παίκτης που θα καταφέρει να παραμείνει ζωντανός και να επιβιώσει από τους κινδύνους που εγκυμονεί το νησί. Προκειμένου να το πετύχει αυτό πρέπει να αναζητήσει τροφή και όπλα, τα οποία βρίσκονται στο κέντρο του νησιού. Τα όπλα είναι απαραίτητα για την αντιμετώπιση των παγίδων που είναι κρυμμένες στο νησί, αλλά και για την εξόντωση των αντιπάλων.

Στην παρούσα εργασία υλοποιούμε μια απλουστευμένη προσομοίωση του παιχνιδιού Hunger Games. Συγκεκριμένα, δύο παίκτες διαγωνίζονται σε ένα ταμπλό αρχικής διάστασης 20×20. Σε κάθε γύρο του παιχνιδιού οι παίκτες μετακινούνται τυχαία κατά μία θέση (δεξιά, αριστερά, πάνω, κάτω, διαγώνια). Σκοπός τους είναι να φτάσουν στο κέντρο του ταμπλό, όπου είναι τοποθετημένα τα τρόφιμα και τα όπλα. Για να το πετύχουν, όμως, αυτό θα πρέπει να αποφύγουν τις διάφορες παγίδες που είναι τοποθετημένες περιμετρικά των εφοδίων. Παράλληλα, η διάσταση του ταμπλό μικραίνει και το παιχνίδι λήγει όταν το ταμπλό αποκτήσει διάσταση 4×4. Νικητής του παιχνιδιού αναδεικνύεται ο παίκτης που θα παραμείνει ζωντανός ή αυτός που θα συγκεντρώσει τους περισσότερους πόντους (σε περίπτωση που και οι δύο παίκτες επιβιώσουν).

Ένας παίκτης συγκεντρώνει πόντους όταν συλλέγει ένα εφόδιο, όταν δηλαδή μετακινηθεί σε ένα πλακίδιο του ταμπλό που περιλαμβάνει ένα εφόδιο. Αντίθετα, οι παίκτες χάνουν πόντους όταν συναντούν μια παγίδα και δεν διαθέτουν το κατάλληλο όπλο για να την αντιμετωπίσουν. Υπάρχουν τρία είδη όπλων: τόξο, σπαθί και πιστόλι. Το τόξο χρησιμοποιείται για την αντιμετώπιση των ζώων, το σπαθί για την αντιμετώπιση των παγίδων που περιλαμβάνουν σχοινιά, ενώ το πιστόλι για την εξόντωση του αντιπάλου (στη συγκεκριμένη εργασία δεν υλοποιούμε το δικαίωμα των παικτών να χρησιμοποιήσουν το πιστόλι προκειμένου να σκοτώσουν τον αντίπαλό τους).

Υλοποίηση του παιχνιδιού σε Java

Η προσομοίωση του παιχνιδιού Hunger Games βασίζεται στη δημιουργία ενός ταμπλό μεγέθους 20×20, που θα περιλαμβάνει συγκεκριμένο αριθμό όπλων, εφοδίων και παγίδων και δύο παικτών, οι οποίοι θα έχουν τη δυνατότητα να μετακινούνται τυχαία, να συλλέγουν όπλα και τρόφιμα και να αντιμετωπίζουν παγίδες. Για την υλοποίηση του παιχνιδιού σε γλώσσα Java χρησιμοποιούμε έξι κλάσεις, τις εξής: Weapon, Food, Trap, Board, Player, Game.

Η κλάση Weapon

Η κλάση Weapon αντιπροσωπεύει τα όπλα του παιχνιδιού. Κάθε όπλο χαρακτηρίζεται από έναν μοναδικό αριθμό (id), τις συντεταγμένες x, y του πλακιδίου στο οποίο βρίσκεται, τον αριθμό του παίκτη ο οποίος μπορεί να πάρει το όπλο (playerId) και από το είδος του (type). Η μεταβλητή type μπορεί να πάρει τρεις τιμές: bow, sword, pistol.

Υλοποιούμε τρεις συναρτήσεις αρχικοποίησης (constructors). Η πρώτη συνάρτηση δεν έχει ορίσματα και αρχικοποιεί τις μεταβλητές του αντικειμένου σε μηδέν, η δεύτερη δέχεται πέντε ορίσματα (id, x, y, playerId, type) και αρχικοποιεί αντίστοιχα τις μεταβλητές και η τρίτη αρχικοποιεί το αντικείμενο με βάση ένα άλλο αντικείμενο του ίδιου τύπου.

Στις μεθόδους της κλάσης περιλαμβάνονται ακόμα οι συναρτήσεις εκχώρησης τιμής και επιστροφής της τιμής όλων των μεταβλητών της κλάσης (συναρτήσεις get και set).

Η κλάση Food

Η κλάση Food αναπαριστά τα εφόδια (τρόφιμα) του παιχνιδιού. Κάθε εφόδιο χαρακτηρίζεται από έναν μοναδικό αριθμό (id), τις συντεταγμένες x, y του πλακιδίου στο οποίο βρίσκεται και τους πόντους τους οποίους κερδίζει ο παίκτης όταν το τρώει (points).

Υλοποιούμε τρεις συναρτήσεις αρχικοποίησης (constructors). Η πρώτη συνάρτηση δεν έχει ορίσματα και αρχικοποιεί τις μεταβλητές του αντικειμένου σε μηδέν, η δεύτερη δέχεται τέσσερα ορίσματα (id, x, y, points) και αρχικοποιεί αντίστοιχα τις μεταβλητές και η τρίτη αρχικοποιεί το αντικείμενο με βάση ένα άλλο αντικείμενο του ίδιου τύπου.

Στις μεθόδους της κλάσης περιλαμβάνονται ακόμα οι συναρτήσεις εκχώρησης τιμής και επιστροφής της τιμής όλων των μεταβλητών της κλάσης (συναρτήσεις get και set).

Η κλάση Trap

Η κλάση Trap αναπαριστά τις παγίδες του παιχνιδιού. Κάθε παγίδα χαρακτηρίζεται από έναν μοναδικό αριθμό (id), τις συντεταγμένες x, y του πλακιδίου στο οποίο βρίσκεται, το είδος της παγίδας (type) και τους πόντους τους οποίους χάνει ο παίκτης όταν δεν έχει το κατάλληλο όπλο για να την αντιμετωπίσει (points). Η μεταβλητή type μπορεί να πάρει δύο τιμές: animal, ropes.

Υλοποιούμε τρεις συναρτήσεις αρχικοποίησης (constructors). Η πρώτη συνάρτηση δεν έχει ορίσματα και αρχικοποιεί τις μεταβλητές του αντικειμένου σε μηδέν, η δεύτερη δέχεται πέντε ορίσματα (id, x, y, type, points) και αρχικοποιεί αντίστοιχα τις μεταβλητές και η τρίτη αρχικοποιεί το αντικείμενο με βάση ένα άλλο αντικείμενο του ίδιου τύπου.

Στις μεθόδους της κλάσης περιλαμβάνονται ακόμα οι συναρτήσεις εκχώρησης τιμής και επιστροφής της τιμής όλων των μεταβλητών της κλάσης (συναρτήσεις get και set).

Η κλάση Board

Η κλάση Board αναπαριστά το ταμπλό του παιχνιδιού, το οποίο χαρακτηρίζεται από τις διαστάσεις του (N, M), τον αριθμό των όπλων, των εφοδίων και των παγίδων που υπάρχουν (W, F, T αντίστοιχα). Στις μεταβλητές της κλάσης περιλαμβάνονται ακόμα τρεις δισδιάστατοι πίνακες μεγέθους 4x2, στους οποίους αποθηκεύονται οι συντεταγμένες των κορυφών των τετραγώνων που περιέχουν όπλα, τρόφιμα και παγίδες (πίνακες weaponAreaLimits, foodAreaLimits, trapAreaLimits αντίστοιχα), όπως φαίνονται στην *Εικόνα 1* και τρεις μονοδιάστατοι πίνακες, στους οποίους αποθηκεύονται όλα τα όπλα, τα τρόφιμα και οι παγίδες που υπάρχουν στο ταμπλό (πίνακες weapons, food, traps αντίστοιχα).

Υλοποιούμε τρεις συναρτήσεις αρχικοποίησης (constructors). Η πρώτη συνάρτηση δεν έχει ορίσματα και αρχικοποιεί τις μεταβλητές του αντικειμένου σε μηδέν και τους πίνακες σε null, η δεύτερη δέχεται πέντε ορίσματα (N, M, W, F, T), αρχικοποιεί τις αντίστοιχες μεταβλητές, αποθηκεύει στους πίνακες ορίων τις κατάλληλες συντεταγμένες (οι συντεταγμένες αυτές φαίνονται στην *Εικόνα 1*) και δημιουργεί τους πίνακες weapons, food, traps διάστασης W, F, T αντίστοιχα. Η τρίτη συνάρτηση αρχικοποιεί το αντικείμενο με βάση ένα άλλο αντικείμενο του ίδιου τύπου.

Στις μεθόδους της κλάσης περιλαμβάνονται ακόμα οι συναρτήσεις εκχώρησης τιμής και επιστροφής της τιμής όλων των μεταβλητών της κλάσης (συναρτήσεις get και set).

Η συνάρτηση *createRandomWeapon()* αρχικοποιεί τον πίνακα των όπλων με τυχαίο τρόπο. Δηλαδή, ορίζει τυχαία τις συντεταγμένες κάθε αντικειμένου (οι συντεταγμένες θα πρέπει να είναι μοναδικές ως προς κάθε αντικείμενο), τον μοναδικό αριθμό του, τον αριθμό του παίκτη στον οποίον ανήκει και το είδος του. Πρέπει τα μισά όπλα να ανήκουν στον παίκτη 1 και τα άλλα μισά τον παίκτη 2. Ακόμα, το 1/3 των όπλων είναι του τύπου bow, το 1/3 του τύπου pistol και το υπόλοιπο 1/3 του τύπου sword. Η κατανομή των όπλων φαίνεται στην *Εικόνα 2*. Με βάση τα παραπάνω θεωρούμε ότι ο αριθμός των όπλων είναι πολλαπλάσιος του 6. Για την υλοποίηση της συνάρτησης, δημιουργούμε έναν δισδιάστατο

πίνακα που περιέχει τις πιθανές συντεταγμένες των όπλων (άσπρη περιοχή στην Εικόνα 1) και μία λίστα (του τύπου ArrayList). Η λίστα χρησιμεύει για την αποθήκευση των συντεταγμένων κάθε νέου αντικειμένου και για τον έλεγχο της εκχώρησης μοναδικών συντεταγμένων σε κάθε νέο αντικείμενο. Συγκεκριμένα, καλούμε τη συνάρτηση *Math.random()* για την παραγωγή ενός τυχαίου αριθμού στο διάστημα [0, 15]. Στη συνέχεια ελέγχουμε αν ο αριθμός αυτός περιλαμβάνεται στη λίστα. Αν δεν περιλαμβάνεται, τότε εκχωρούμε στις συντεταγμένες x, y του αντικειμένου το ζεύγος αριθμών που αντιστοιχεί στη σειρά του πίνακα που δείχνει ο τυχαίος αριθμός, δηλαδή

```
x = coordinates[r][0];
```

```
y = coordinates[r][1];
```

(όπου *coordinates* ο πίνακας συντεταγμένων και *r* ο τυχαίος αριθμός που παράγουμε) και αποθηκεύουμε τον αριθμό στη λίστα. Από την άλλη πλευρά, αν ο τυχαίος αριθμός περιλαμβάνεται στη λίστα, τότε παράγουμε εκ νέου έναν τυχαίο αριθμό και επαναλαμβάνουμε την παραπάνω διαδικασία.

Οι συναρτήσεις *createRandomFood()* και *createRandomTrap()* υλοποιούνται με παρόμοιο τρόπο. Για τον υπολογισμό των πόντων των εφοδίων και των παγίδων χρησιμοποιούμε τη συνάρτηση *Math.random()* για την παραγωγή τυχαίων αριθμών στο διάστημα [1, 10] και [-10, -1], αντίστοιχα.

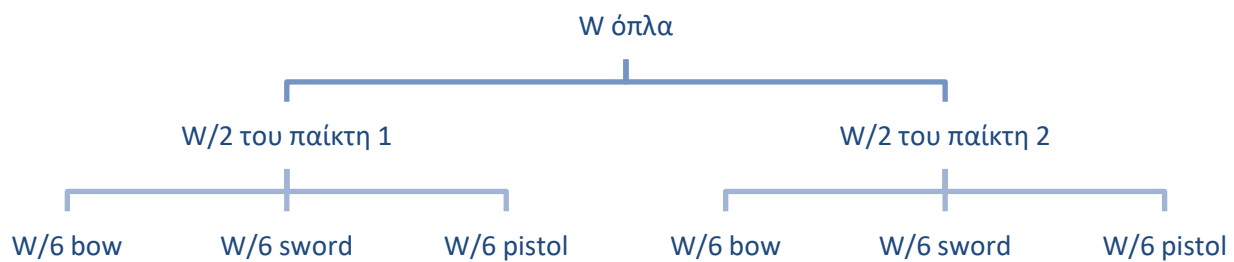
Η συνάρτηση *createRandomBoard()* καλεί τις συναρτήσεις *createRandomWeapon()*, *createRandomFood()*, *createRandomTrap()* και δημιουργεί με τυχαίο τρόπο το ταμπλό του παιχνιδιού.

Η συνάρτηση *resizeBoard()* δέχεται ως όρισμα τους δύο παίκτες και ελέγχει αν ένας τουλάχιστον από αυτούς βρίσκεται στο εξωτερικό περίβλημα του ταμπλό. Αν κανένας παίκτης δεν βρίσκεται στα εξωτερικά πλακίδια, τότε αυτά αφαιρούνται, δηλαδή μειώνονται οι διαστάσεις του ταμπλό κατά δύο μονάδες.

Η συνάρτηση *getStringRepresentation()* δημιουργεί και επιστρέφει έναν πίνακα διάστασης N×M, ο οποίος αναπαριστά κάθε πλακίδιο του ταμπλό με ένα αλφαριθμητικό. Συγκεκριμένα, αν ένα πλακίδιο του ταμπλό δεν περιέχει κάποιο αντικείμενο, τότε το συμβολίζουμε με το αλφαριθμητικό “___”. Αν περιέχει ένα όπλο, τότε το συμβολίζουμε με “W + playerId + id”, όπου *playerId* το αναγνωριστικό του παίκτη και *id* ο μοναδικός αριθμός του όπλου. Αν το πλακίδιο περιέχει εφόδιο ή παγίδα γράφουμε αντίστοιχα “F + id”, “T + id”, όπου *id* ο μοναδικός αριθμός του εφοδίου ή του όπλου. Για την υλοποίηση της συνάρτησης δημιουργούμε αρχικά έναν δισδιάστατο πίνακα μεγέθους N×M και εκχωρούμε σε κάθε κελί του το αλφαριθμητικό “___”. Στη συνέχεια ελέγχουμε κάθε αντικείμενο των πινάκων *weapons*, *food* και *traps* και ανάλογα με τις συντεταγμένες του κάθε αντικειμένου εκχωρούμε το κατάλληλο αλφαριθμητικό (όπως αναφέρεται παραπάνω) στο αντίστοιχο κελί του πίνακα. Ο έλεγχος των εφοδίων πραγματοποιείται μόνο σε περίπτωση που το ταμπλό του παιχνιδιού έχει διάσταση μεγαλύτερη ή ίση του 6 ($N = M \geq 6$). Ομοίως, ο έλεγχος των παγίδων πραγματοποιείται μόνο όταν το ταμπλό έχει διάσταση μεγαλύτερη ή ίση του 8 ($N = M \geq 8$).

		...	$\frac{M}{2} - 4$	$\frac{M}{2} - 3$	$\frac{M}{2} - 2$	$\frac{M}{2} - 1$	$\frac{M}{2}$	$\frac{M}{2} + 1$	$\frac{M}{2} + 2$	$\frac{M}{2} + 3$...	
...												
$\frac{N}{2} - 4$			(-4, -4)	(-3, -4)	(-2, -4)	(-1, -4)	(1, -4)	(2, -4)	(3, -4)	(4, -4)		
$\frac{N}{2} - 3$			(-4, -3)	(-3, -3)	(-2, -3)	(-1, -3)	(1, -3)	(2, -3)	(3, -3)	(4, -3)		
$\frac{N}{2} - 2$			(-4, -2)	(-3, -2)	(-2, -2)	(-1, -2)	(1, -2)	(2, -2)	(3, -2)	(4, -2)		
$\frac{N}{2} - 1$			(-4, -1)	(-3, -1)	(-2, -1)	(-1, -1)	(1, -1)	(2, -1)	(3, -1)	(4, -1)		
$\frac{N}{2}$			(-4, 1)	(-3, 1)	(-2, 1)	(-1, 1)	(1, 1)	(2, 1)	(3, 1)	(4, 1)		
$\frac{N}{2} + 1$			(-4, 2)	(-3, 2)	(-2, 2)	(-1, 2)	(1, 2)	(2, 2)	(3, 2)	(4, 2)		
$\frac{N}{2} + 2$			(-4, 3)	(-3, 3)	(-2, 3)	(-1, 3)	(1, 3)	(2, 3)	(3, 3)	(4, 3)		
$\frac{N}{2} + 3$			(-4, 4)	(-3, 4)	(-2, 4)	(-1, 4)	(1, 4)	(2, 4)	(3, 4)	(4, 4)		
...												

Εικόνα 1: Αναπαράσταση ενός ταμπλό μεγέθους 12×12. Η μπλε περιοχή αντιστοιχεί στις παγίδες, η γκρι στα εφόδια και η άσπρη στα όπλα. Φαίνονται ακόμα οι αριθμοδείκτες των πλακιδίων για αναπαράσταση του ταμπλό σε δισδιάστατο πίνακα μεγέθους N×M, όπου N = 12, M = 12.



Εικόνα 2: Αναπαράσταση της κατανομής των όπλων για δεδομένο αριθμό όπλων W .

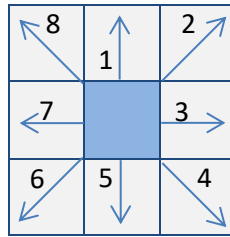
Η κλάση Player

Η κλάση Player αντιπροσωπεύει έναν παίκτη του παιχνιδιού. Κάθε παίκτης χαρακτηρίζεται από τον μοναδικό κωδικό του (id), το όνομά του (name), τους πόντους του (score), τις συντεταγμένες x , y του πλακιδίου στο οποίο βρίσκεται, το ταμπλό του παιχνιδιού στο οποίο συμμετέχει (board) και από τρία αντικείμενα του τύπου Weapon, τα οποία αναπαριστούν τα τρία όπλα που μπορεί να πάρει ο παίκτης (σπαθί, τόξο, πιστόλι). Σε περίπτωση που ο παίκτης δεν έχει κάποιο όπλο, τότε η τιμή του είναι null.

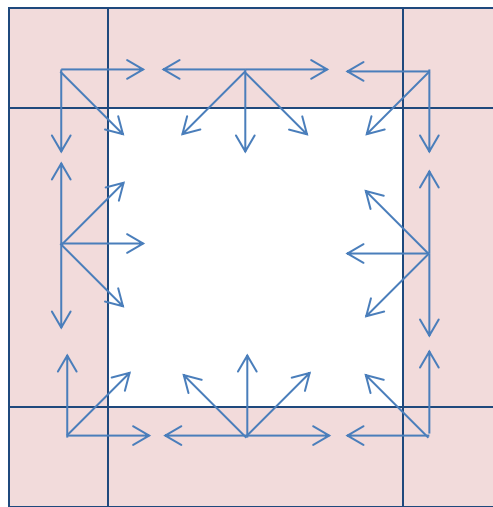
Υλοποιούμε δύο συναρτήσεις αρχικοποίησης (constructors). Η πρώτη συνάρτηση δεν έχει ορίσματα και αρχικοποιεί τις μεταβλητές του αντικειμένου σε μηδέν και τα αντικείμενα των τύπων Board και Weapon σε null και η δεύτερη δέχεται έξι ορίσματα (id, score, name, board, x , y), αρχικοποιεί τις αντίστοιχες μεταβλητές και θέτει τα αντικείμενα του τύπου Weapon σε null (στην αρχή του παιχνιδιού οι παίκτες δεν διαθέτουν όπλα).

Στις μεθόδους της κλάσης περιλαμβάνονται ακόμα οι συναρτήσεις εκχώρησης τιμής και επιστροφής της τιμής όλων των μεταβλητών της κλάσης (συναρτήσεις get και set).

Η συνάρτηση *getRandomMove()* παράγει έναν τυχαίο αριθμό στο διάστημα $[1, 8]$, ο οποίος αντιπροσωπεύει την κίνηση του παίκτη, όπως φαίνεται στην Εικόνα 3 και επιστρέφει έναν μονοδιάστατο πίνακα με τις νέες συντεταγμένες. Αν ο παίκτης βρίσκεται στα εξωτερικά πλακίδια οι διαθέσιμες κινήσεις είναι λιγότερες. Έτσι, αρχικά ελέγχουμε αν ο παίκτης βρίσκεται στο εσωτερικό του ταμπλό, ώστε να καλέσουμε τη συνάρτηση *Math.random()* για την παραγωγή ενός τυχαίου αριθμού στο διάστημα $[1, 8]$. Αν ο παίκτης βρίσκεται στα εξωτερικά πλακίδια, τότε ελέγχουμε ξεχωριστά τις εξής οκτώ περιπτώσεις: πάνω δεξιά γωνία, πάνω αριστερή γωνία, κάτω δεξιά γωνία, κάτω αριστερή γωνία, πρώτη γραμμή, τελευταία γραμμή, πρώτη στήλη και τελευταία στήλη και παράγουμε έναν τυχαίο αριθμό στα αντίστοιχα διαστήματα, όπως φαίνεται στην Εικόνα 4. Τέλος, ανάλογα με τον αριθμό της κίνησης υπολογίζουμε τις νέες συντεταγμένες.



Εικόνα 3: Αναπαράσταση των διαθέσιμων κινήσεων του κάθε παίκτη.



Εικόνα 4: Αναπαράσταση των διαθέσιμων κινήσεων του κάθε παίκτη στα εξωτερικά πλακίδια του ταμπλό.

Η συνάρτηση *move()* καλεί τη συνάρτηση *getRandomMove()* για την εύρεση της νέας θέσης του παίκτη και μετακινεί τον παίκτη στο νέο πλακίδιο. Επιστρέφει έναν μονοδιάστατο πίνακα που περιλαμβάνει τις νέες συντεταγμένες *x*, *y*, τον αριθμό των όπλων και των εφοδίων που μάζεψε ο παίκτης στην κίνηση αυτή και τον αριθμό των παγίδων που συνάντησε. Κάθε παίκτης μπορεί να μαζέψει μόνο τα όπλα, των οποίων η μεταβλητή *playerId* ταυτίζεται με τον κωδικό του παίκτη (*id*). Όταν ο παίκτης παίρνει ένα όπλο μηδενίζονται οι συντεταγμένες του (εξαφανίζεται δηλαδή από το ταμπλό) και εμφανίζεται σχετικό μήνυμα. Ανάλογα με το είδος του όπλου, ενημερώνεται και το αντίστοιχο αντικείμενο του τύπου *Weapon* του παίκτη. Όταν ένας παίκτης συλλέγει ένα εφόδιο, τότε μηδενίζονται οι συντεταγμένες του εφοδίου και προστίθενται οι πόντοι του εφοδίου στους βαθμούς του παίκτη. Όταν ο παίκτης συναντάει μία παγίδα, τότε ελέγχεται αν διαθέτει το κατάλληλο όπλο για να την αντιμετωπίσει. Αν δεν το διαθέτει, τότε αφαιρούνται από τους βαθμούς του παίκτη οι πόντοι της παγίδας και εμφανίζεται αντίστοιχο μήνυμα. Σε κάθε κίνηση ο παίκτης μπορεί να μαζέψει μόνο ένα αντικείμενο, αφού σε κάθε πλακίδιο υπάρχει μοναδικό αντικείμενο.

Η κλάση Game

Η κλάση Game αντιπροσωπεύει το παιχνίδι, το οποίο χαρακτηρίζεται από μία μόνο μεταβλητή: τον τρέχον γύρο (round).

Υλοποιούμε δύο συναρτήσεις αρχικοποίησης (constructors). Η πρώτη συνάρτηση δεν έχει ορίσματα και αρχικοποιεί τη μεταβλητή του αντικειμένου σε μηδέν και η δεύτερη δέχεται ως όρισμα έναν ακέραιο αριθμό, οποίος εκχωρείται στη μεταβλητή round.

Στις μεθόδους της κλάσης περιλαμβάνονται ακόμα οι συναρτήσεις εκχώρησης τιμής και επιστροφής της τιμής της μεταβλητής της κλάσης (συναρτήσεις get και set).

Η συνάρτηση *main()* είναι η κύρια συνάρτηση του προγράμματος. Δημιουργούμε ένα ταμπλό διάστασης 20×20 με 6 όπλα, 10 εφόδια και 8 παγίδες, τυχαία κατανεμημένα στο ταμπλό. Ορίζουμε δύο παίκτες· ο πρώτος βρίσκεται στην πάνω αριστερή γωνία του ταμπλό και ο δεύτερος στην κάτω δεξιά. Σε κάθε γύρο του παιχνιδιού οι παίκτες μετακινούνται τυχαία κατά μία θέση, εκτυπώνεται το ταμπλό και η κατάσταση του κάθε παίκτη (νέα θέση, αριθμός όπλων, βαθμοί). Ορίζουμε μία μεταβλητή *resizeTime*, η οποία αυξάνεται σε κάθε γύρο του παιχνιδιού και χρησιμοποιείται για να ελέγξουμε πότε πρέπει να μικρύνει το ταμπλό. Συγκεκριμένα, όταν η μεταβλητή *resizeTime* είναι μεγαλύτερη ή ίση του τρία, καλούμε τη συνάρτηση *resizeBoard()*. Αν οι διαστάσεις του ταμπλό μικρύνουν (αυτό συμβαίνει όταν κανένας παίκτης δε βρίσκεται στα εξωτερικά πλακίδια), τότε η τιμή της μεταβλητής *resizeTime* γίνεται μηδέν. Αν το ταμπλό δεν μικρύνει, τότε επαναλαμβάνεται ο έλεγχος στον επόμενο γύρο. Το παιχνίδι συνεχίζεται όσο οι διαστάσεις N, M του ταμπλό είναι μεγαλύτερες του τέσσερα. Στο τέλος του παιχνιδιού τυπώνεται το ταμπλό και οι βαθμοί του κάθε παίκτη. Νικητής είναι ο παίκτης που συγκέντρωσε τους περισσότερους πόντους.