



CANDY CRUSH

ΕΥΡΙΠΙΔΗΣ ΜΠΑΛΤΖΗΣ 8196

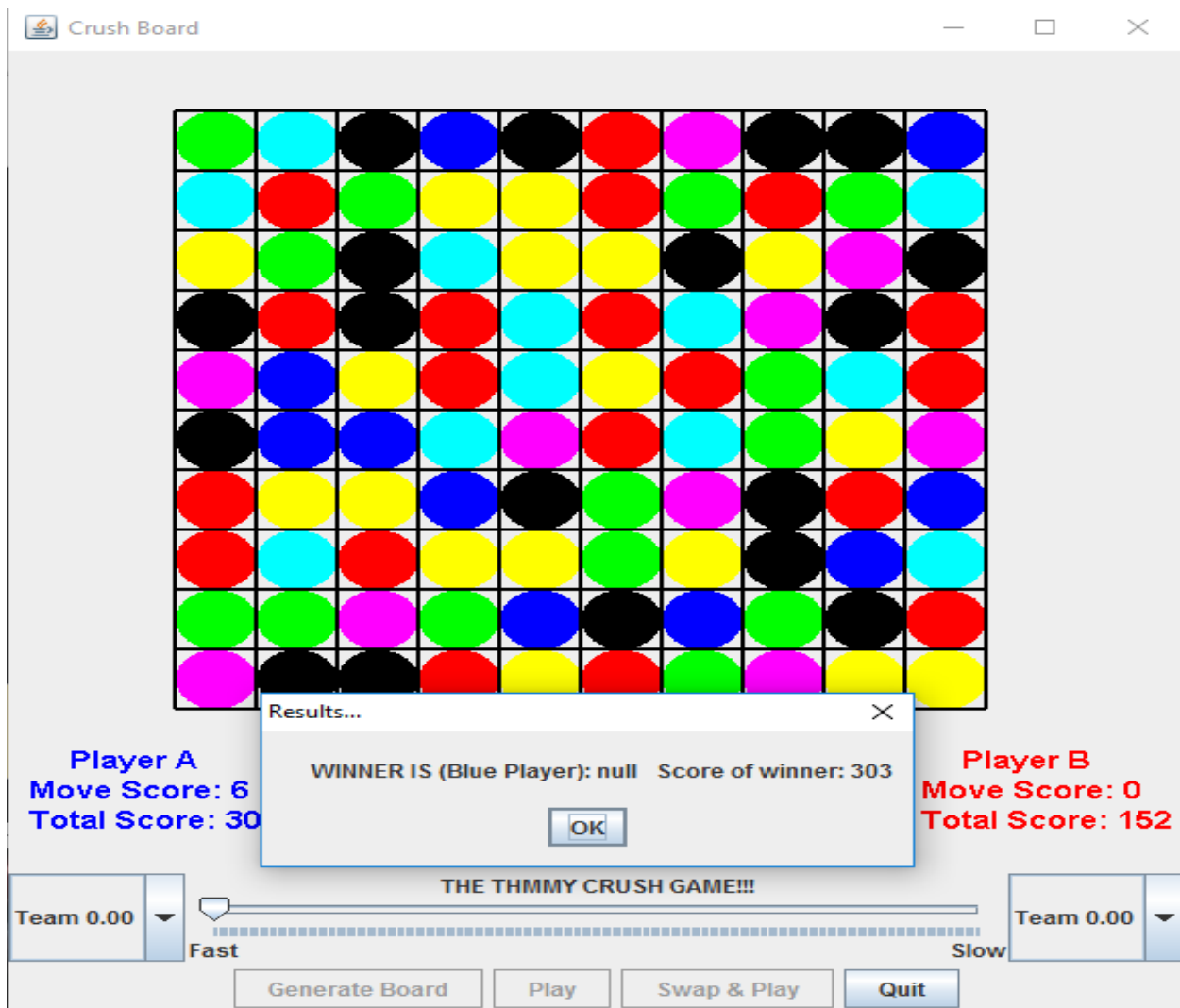
ΒΑΣΙΛΗΣ ΜΠΕΛΛΟΣ 8715

Στο **πρώτο** μέρος της εργασίας κληθήκαμε να συμπληρώσουμε τις κλάσεις Tile και RandomPlayer με όλες τις απαραίτητες συναρτήσεις, ώστε:

A) η κλάση Tile να δημιουργεί τα πλακίδια με τα ζαχαρωτά στο ταμπλό και

B) η κλάση RandomPlayer να δημιουργεί έναν καινούριο παίκτη, ο οποίος θα επιλέγει με τυχαίο τρόπο την επόμενη κίνησή του, θα υπολογίζει δηλαδή τις συντεταγμένες των ζαχαρωτών που θα ανταλλάξουν θέση κατά την κίνηση αυτή.

Μετά την υλοποίηση της κλάσης αυτής θα πρέπει να τρέξουμε το παιχνίδι επιλέγοντας και οι 2 παίκτες να είναι της μορφής RandomPlayer και περιμένουμε ως αποτέλεσμα τη διεξαγωγή ενός παιχνιδιού μεταξύ των τυχαίων παικτών που δημιουργεί η κλάση που υλοποιήσαμε, όπως φαίνεται στην παρακάτω εικόνα.



Class Tile

Παρακάτω παρουσιάζεται η υλοποίηση της ζητούμενης κλάσης και περιγράφονται οι μεταβλητές, μέθοδοι και συναρτήσεις που την αποτελούν.

Αρχικά, οι μεταβλητές ενός αντικειμένου της κλάσης αυτές είναι:

- `protected int id`: περιέχει την ταυτότητα του εκάστοτε πλακιδίου στο ταμπλό
- `private int x`: περιέχει την τετμημένη του εκάστοτε πλακιδίου
- `private int y`: περιέχει την τεταγμένη του εκάστοτε πλακιδίου
- `private int color`: περιέχει το χρώμα του εκάστοτε πλακιδίου
- `private boolean mark`: περιέχει τη λογική τιμή 1 ή 0, ανάλογα με το εάν πρόκειται (ή όχι αντίστοιχα) να διαγραφεί στην επόμενη κίνηση εφόσον δημιουργεί τριάδα, τετράδα ή πεντάδα ίδιου χρώματος πλακιδίων στο ταμπλό

Στη συνέχεια αναλύονται οι συναρτήσεις και μέθοδοι της κλάσης.

- **Constructor `Tile(int id, int xTile, int yTile, int colorTile, Boolean markTile):`** : ο constructor της κλάσης, ο οποίος αντιστοιχεί τις τιμές των ορισμάτων με τις αντίστοιχες μεταβλητές της κλάσης, όπως φαίνεται παρακάτω:

```
id = idTile;  
x = xTile;  
y = yTile;  
color = colorTile;  
mark = markTile;
```

- **Όλες οι συναρτήσεις της μορφής `typos_metavlhths getOnomaMetavlhths()`**: οι συναρτήσεις αυτές επιστρέφουν τις αντίστοιχες τιμές των μεταβλητών που περιέχονται στο όνομα της συνάρτησης (π.χ. η συνάρτηση `int getId()` επιστρέφει την τιμή της μεταβλητής `id` του αντικειμένου της κλάσης που την καλεί).
- **Όλες οι συναρτήσεις της μορφής `void setOnomaMetavlhths(typos_metavlhths orisma)`**: οι συναρτήσεις αυτές δέχονται ως όρισμά τους μια μεταβλητή τύπου ανάλογου του τύπου της μεταβλητής που περιέχεται στο όνομα της συνάρτησης και περνάνε την τιμή του ορίσματος στην αντίστοιχη μεταβλητή του αντικειμένου της κλάσης που τις καλεί (π.χ. η συνάρτηση `setId(int pid)` περνάει την τιμή του ορίσματος `pid` στη μεταβλητή `id` του αντικειμένου της κλάσης που την καλεί).

Class RandomPlayer

Παρακάτω παρουσιάζεται η υλοποίηση της ζητούμενης κλάσης και περιγράφονται οι μεταβλητές, μέθοδοι και συναρτήσεις που την αποτελούν.

Αρχικά, οι μεταβλητές ενός αντικειμένου της κλάσης αυτές είναι:

- **int id:** η μεταβλητή αυτή παίρνει τις τιμές 1 ή 2 και καθορίζει αν ο παίκτης που δημιουργήθηκε είναι ο μπλε ή ο κόκκινος.
- **String name:** περιέχει το όνομα του εκάστοτε παίκτη
- **int score:** περιέχει το συνολικό σκορ του παίκτη, το οποίο προκύπτει από το άθροισμα όλων των τιμών εντός των πλακιδίων που ανήκουν στον παίκτη

Στη συνέχεια αναλύονται οι συναρτήσεις και μέθοδοι της κλάσης.

- **Constructor RandomPlayer(Integer pid):** ο constructor της κλάσης, ο οποίος δέχεται ως όρισμα του έναν integer και περνάει την τιμή του στη μεταβλητή id του αντικειμένου που δημιουργείται.
- **Όλες οι συναρτήσεις της μορφής `typos_metavlhths getOnomaMetavlhths()`:** οι συναρτήσεις αυτές επιστρέφουν τις αντίστοιχες τιμές των μεταβλητών που περιέχονται στο όνομα της συνάρτησης (π.χ. η συνάρτηση `int getId()` επιστρέφει την τιμή της μεταβλητής id του αντικειμένου της κλάσης που την καλεί).
- **Όλες οι συναρτήσεις της μορφής `void setOnomaMetavlhths(typos_metavlhths orisma)`:** οι συναρτήσεις αυτές δέχονται ως όρισμά τους μια μεταβλητή τύπου ανάλογου του τύπου της μεταβλητής που περιέχεται στο όνομα της συνάρτησης και περνάνε την τιμή του ορίσματος στην αντίστοιχη μεταβλητή του αντικειμένου της κλάσης που τις καλεί (π.χ. η συνάρτηση `setId(int pid)` περνάει την τιμή του ορίσματος pid στη μεταβλητή id του αντικειμένου της κλάσης που την καλεί).
- **Συνάρτηση `int[] getNextMove(ArrayList<int[]> availableMoves, Board board)`:** η συνάρτηση αυτή δέχεται σαν όρισμα ένα αντικείμενο της κλάσης Board, δηλ. δέχεται ως όρισμα ολόκληρο το ταμπλό, καθώς και το δυναμικό πίνακα availableMoves, που περιέχει όλες τις διαθέσιμες κινήσεις πάνω στο ταμπλό τη δεδομένη στιγμή, και καλείται να επιστρέψει ένα μονοδιάστατο πίνακα 4 θέσεων, που θα περιέχει τις συντεταγμένες του πλακιδίου που θα επιλέξει ο παίκτης να χρησιμοποιήσει για την επόμενη κίνησή του, καθώς και τις συντεταγμένες του πλακιδίου με το οποίο θα γίνει η ανταλλαγή θέσεων. Η επιλογή του κάθε πλακιδίου γίνεται τυχαία με τη χρήση ενός αντικειμένου της κλάσης Random και της αντίστοιχης μεθόδου `nextInt(bound)`. Η μέθοδος αυτή παράγει με τυχαίο

τρόπο έναν αριθμό στο διάστημα $[0, \text{bound})$, όπου bound είναι μια μεταβλητή της μορφής `int`. Στη περίπτωση μας ως bound χρησιμοποιείται το μέγεθος του δυναμικού πίνακα `availableMoves.size()` και το αποτέλεσμα της όλης διαδικασίας περνιέται στη μεταβλητή `index`.

Στη συνέχεια, ο δυναμικός πίνακας `availableMoves`, μαζί με το δείκτη `index`, χρησιμοποιούνται ως ορίσματα στη στατική συνάρτηση `getRandomMove` της κλάσης `CrushUtilities`, η οποία επιστρέφει ένα διάνυσμα 1×3 , που περιέχει τις συντεταγμένες ενός τυχαίου πλακιδίου του ταμπλό, και μια μεταβλητή τύπου `int`, που προσδιορίζει το πλακίδιο με το οποίο θα γλίνει η ανταλλαγή θέσεων μέσω ενός αριθμού που δείχνει την κατεύθυνση της κίνησης του πλακιδίου. Ο κωδικός των κατευθύνσεων περιέχεται στις στατικές μεταβλητές της κλάσης `CrushUtilities` και περιγράφεται από τον παρακάτω πίνακα:

LEFT = 0	(Αριστερά)
DOWN = 1	(Κάτω)
RIGHT = 2	(Δεξιά)
UP = 3	(Πάνω)

Το επιστρεφόμενο διάνυσμα της `getRandomMove` χρησιμοποιείται στη συνάρτηση και ανάλογα με την κατεύθυνση που έχει επιλεγεί, υπολογίζονται και οι συντεταγμένες του πλακιδίου με το οποίο θα γίνει η ανταλλαγή θέσεων, ενώ επίσης γίνεται έλεγχος για το αν η επιλεγμένη κατεύθυνση κίνησης πρόκειται να υπερβεί τα όρια του ταμπλό για το επόμενο πλακίδιο. Παρακάτω παρουσιάζεται ο πίνακας επιλογής πλακιδίου ανάλογα με την κατεύθυνση της κίνησης:

Κατεύθυνση κίνησης	Τετμημένη πλακιδίου	Τεταγμένη πλακιδίου
LEFT	$x-1$	y
DOWN	x	$y-1$
RIGHT	$x+1$	y
UP	x	$y+1$