

## Πανεπιστήμιο Πελοποννήσου Πληροφορικής και Τηλεπικοινωνιών

Μότση Πολυξένη  
ΑΜ: 2022201800125

Μπενακίδης Ιωάννης  
ΑΜ: 2022201800128

### Έχουν υλοποιηθεί όλα τα ζητούμενα της άσκησης.

Το πρόγραμμα ξεκινάει από την κλάση "Game". Αρχικά, δηλώνουμε τα αντικείμενα των κλάσεων:

- **Κλάση Menu:** Περιέχει ότι χρειαζόμαστε για την ορθή εμφάνιση του παιχνιδιού. Για παράδειγμα, τους κανόνες.
- **Κλάσεις GiverEntity & ReceiverEntity:** Περιέχουν τις πληροφορίες για την τράπεζα NGO και για την Mafia αντίστοιχα. Επιπρόσθετα, τα πεδία των κλάσεων αυτών βρίσκονται στην αφηρημένη κλάση GenInfo.
- **Κλάση Board:** Περιέχει τις μεθόδους addSquare, getSquare και getInSquare, οι οποίες αναλύονται σε σχόλια στον κώδικα.

Τώρα, με τα αντικείμενα boardfile και codefile ανοίγουμε δυο αρχεία που θα μας βοηθήσουν για την δημιουργία των τετραγώνων. Τα δυο αυτά αρχεία είναι:

- **BoardRules.txt:** Περιέχει το κείμενο που πρέπει να εκτυπώνει το κάθε κουτί.
- **RuleCode.txt:** Περιέχει τις πληροφορίες που αντυπροσοπεύει το κάθε κουτί σε μία μορφή για την διευκόλυνσή μας, οι οποίες αποθηκεύονται στην **κλάση acts**. Για παράδειγμα στον κωδικό 3000,0,0,1,1,0,1 αντιστοιχούν ως εξής:

1. **Money:** Χρήματα που αφαιρούνται ή προστίθενται.  
Για παράδειγμα, σε αυτή την περίπτωση θα αφαιρεθούν 3000 δολάρια από τον παίκτη και θα καταχωρηθούν στα λεφτά της Mafia, ενώ στην περίπτωση που ο αριθμός είναι -1000 τότε τα λεφτά αφαιρούνται από την τράπεζα και καταχωρούνται στον παίκτη.
2. **Move:** Είναι οι θέσεις που μετακινείται ο παίκτης είτε μπροστά είτε πίσω, το οποίο καθορίζεται από το αν ο αριθμός είναι θετικός ή αρνητικός αντίστοιχα.
3. **Stay:** Είναι ο αριθμός των γύρων που ο παίκτης μένει ακίνητος.
4. **Roll dice:** Είναι ο αριθμός των ζαριών που έχει να πετάξει.
5. **Alive:** Αν ο παίκτης είναι ζωντανός ή όχι.
6. **Extra life:** Αν ο παίκτης αποκτάει ή δεν αποκτάει σωσίβιο.

Money	Move	Stay	Roll dice	Alive	Extra life	Mafia
3000	0	0	1	1	0	1

Επομένως το παραπάνω μεταφράζεται ως: Δώσε 3000 δολάρια στην Mafia και ξαναρίξε το ζάρι.

## Load Game

Στην switch στο case 2 καλούμε το Saves.txt ώστε να “κατεβάσουμε” τα δεδομένα προηγούμενου παιχνιδιού.

Το Saves.txt είναι και αυτό σε μορφή για την καλύτερη και βολική εκμετάλλευση των στοιχείων που έχουν αποθηκευτεί. Για παράδειγμα,

### 2\_SAVE

Ξένιος 6666 2 alive

Μπόμπος 101 26 alive

ΜπενΔεΤεν 1000 20 alive

Mafia 22233\$

NGO 9000\$

<Ξένιος,6666,0,2,22233,9000,1,1,0,0>

<Μπόμπος,101,0,26,22233,9000,1,0,0,0>

<ΜπενΔεΤεν,1000,0,20,22233,9000,1,0,0,0>

end2

Κάτω από τον αριθμό του παιχνιδιού,

έχει το όνομα του πάικτη, τα λεφτά του, την θέση του αντίστοιχα αλλά και το αν είναι ζωντανός ή όχι.

Επιπλέον, φαίνονται τα λεφτά που διαθέτει η Mafia αλλά και η τράπεζα NGO.

(τα πάνω στοιχεία είναι σε αυτή την μορφή ώστε αν γίνει η φόρτωση του παιχνιδιού, να γνωρίζει ο χρήστης τα βασικά πράγματα που γίνονται στο παιχνίδι)

Στην συνέχεια, παρατηρούμε την μορφή που βρίκαμε και στο **RuleCode.txt**, πάλι για την βέλτιστη διαχείριση των στοιχείων. Όμως χρησιμοποιούμε λίγο διαφορετικά την σειρά των στοιχείων, για παράδειγμα στον κωδικό <ΜπενΔεΤεν,10000,0,38,0,9000,1,0,0,0> αντιστοιχούν ως εξής:

1. ΜπενΔεΤεν : **Player Name**
2. 1000 : **Money**
3. 0 : **Expenses**
4. 38 : **Current Square**
5. 0 : **Mafia Money**
6. 9000 : **NGO Money**
7. 1 : **Alive (1) or Not (0)**
8. 0 : **Extra life**
9. 0 : **Turns with no movment**
10. 0 : **Extra rolls**

Player Name	Money	Expenses	Current Square	Mafia Money	NGO Money	Alive (1) or Not (0)	Extra life	Turns with no movment	Extra rolls
ΜπενΔεΤεν	1000	0	38	22233	9000	1	0	0	0

## Actions που υλοποιήθηκαν

Όλα τα Actions ξεκινάνε από την κλάση Square στην μέθοδο act() και ελέγχουμε κάθε πληροφορία με την σειρά. Δηλαδή, αν για παράδειγμα βρισκόμαστε στο τετράγωνο 9, όπου πληρώνουμε 3000\$ στην Mafia και ξανα ρίχνουμε το ζάρι, τότε μπορούμε στην If με την PayAction και μετά με την RollDiceAction.

- **DeadAction:** Αν alive=0 τότε καλούμε την μέθοδο act και κάνουμε έλεγχο αν έχει σωσίβιο στην κατοχή του ο παίκτης. Αν δεν έχει, τότε πεθαίνει και βγαίνει εκτός παιχνιδιού, αλλιώς τον προειδοποιούμε ότι είχε σωσίβιο και σώθηκε.
- **ReceiveAction\*:** Προσθέτουμε τα 1000\$ στον παίκτη.
- **PayAction\*:** Αφαιρούμε τα λεφτά που πρέπει να δώσει ο παίκτης είτε στην Mafia είτε σε άλλα πράγματα, όπως το φαγητό στο τετράγωνο 1.
- **LifeJacketSea:** Μετατρέπουμε σε true το πεδίο extra\_life του παίκτη.
- **RollDiceAction:** Λέμε στον χρήστη να πατήσει το 1 για να ξανα ρίξει το ζάρι. Αν βρεθεί στο τετράγωνο 22 τότε θα μετακινηθεί τόσα βήματα πίσω, αλλιώς τόσα βήματα μπροστά.
- **GoToAction:** Μετακινούμε τον παίκτη στην θέση που πρέπει να πάει.
- **StayAction:** Αλλάζουμε την πληροφορία στον χρήστη, ότι χάνει την σειρά του.

\* Θα θέλαμε να σας ενημερώσουμε ότι δεν μπορούσαμε να βρούμε άλλον τρόπο έτσι ώστε να έχουμε πρόσβαση είτε στην NGO είτε στην Mafia από τις κλάσεις ReceiveAction και PayAction επειδή κάνουν override την μέθοδο act() που περνάει ως παράμετρο την κλάση Refugee και μόνο. Επομένως αυτές οι λειτουργίες γίνονται στην κλάση Board, στην μέθοδο getInSquare() με παραμέτρους τον Refugee και τις άλλες δυο τράπεζες. Αν σε κάποιο τετράγωνο με την μέθοδο getMoney() της κλάσης Square πάρουμε έναν θετικό αριθμό τότε η Mafia θα πάρει αυτά τα λεφτά. Τα λεφτά τα παίρνουμε την αφαίρεση από τα λεφτά που είχε ο παίκτης ΠPIN γίνει η PayAction με τα λεφτά που έχει META την PayAction. Αν όμως έχουμε αρνητικό αριθμό, δηλαδή το -1000, τότε απλά αφαιρούμε τα 1000 από την NGO τράπεζα. Η μεταβλητή flag μας βοηθάει στην περίπτωση που γίνεται RollDice για να μη ξαναφύγουν 1000\$.

## Τετράγωνο 26, με τις δύο επιλογές

Η λειτουργία αυτή γίνεται στην κλάση Square στην μέθοδο act(), που γίνεται ο έλεγχος για το αν πρέπει να πληρώσει ή όχι. Επειδή πρέπει πρώτα να πληρώσει 1000\$ αυτό μας δίνει το πλεονέκτημα να μπει μέσα στην if. Επομένως, κάνουμε έλεγχο αν είμαστε στο τετράγωνο 26. Αν είμαστε, τότε τον ρωτάμε αν θέλει να επιλέξει το Α ή το Β. Αν επιλέξει το Α τότε χάνει έναν γύρο και στην συνέχεια πληρώνει τα 1000\$ που πρέπει να πληρώσει συν τα 1500\$, αλλιώς χάνει 2 γύρους και πληρώνει μόνο τα 1000\$.