

1^η Εργασία στον Αντικειμενοστραφή Προγραμματισμό

Περιγραφή

Σε αυτή την άσκηση καλείστε να υλοποιήσετε ένα πρόγραμμα χρησιμοποιώντας τη γλώσσα προγραμματισμού JAVA το οποίο θα προσομοιώνει έναν απλοποιημένο κλώνο του δημοφιλούς παιχνιδιού στρατηγικής, Battleship. Σκοπός της άσκησης είναι η εξοικείωση με την αναγνώριση κλάσεων, αντικειμένων.

Το παιχνίδι παίζεται σε ένα πλέγμα 9x9. Οι σειρές του πλέγματος αναγνωρίζονται με τους αριθμούς 1 έως 9 (πάνω προς τα κάτω) και οι στήλες αναγνωρίζονται από τους αριθμούς 1 έως 9 (αριστερά προς τα δεξιά).

Οι κανόνες του παιχνιδιού

Υπάρχουν δύο τύποι πλοίων: το **Patrol** και το **Cruiser**. Το **Patrol** καταλαμβάνει ένα μόνο κελί, το **Cruiser** καταλαμβάνει δύο κελιά, οριζόντια ή κάθετα.

Τρεις περιπολίες και τρία καταδρομικά θα τοποθετηθούν τυχαία στο πλέγμα. Ο χρήστης καλεί 10 διαφορετικά κελιά, προσπαθώντας να μαντέψει αν υπάρχει Patrol ή Cruiser σε αυτό.

Τα χτυπήματα και οι απώλειες καταγράφονται από το πρόγραμμα. Για κάθε χτύπημα Patrol ή Cruiser, κερδίζεται ένας βαθμός και εάν ένας Cruiser βυθιστεί, κερδίζονται δύο επιπλέον πόντοι.

Υλοποίηση

Inputs:

Το πρόγραμμα θα πρέπει να δέχεται σαν **είσοδο** έναν πίνακα:

target - ένας πίνακας που περιέχει 10 τιμές που είναι οι εικασίες που έκανε ο χρήστης. Το πρόγραμμα πρέπει να ζητάει από τον χρήστη 1 τιμή ξεχωριστά κάθε φορά μέχρι να συμπληρωθούν οι 10 εικασίες. Ο πίνακας αυτός θα πρέπει να γεμίζει, λοιπόν, με τιμές που θα δίνει ο χρήστης της μορφής «x,y» για την αναπαράσταση του αντίστοιχου πεδίου στο πλέγμα, όπου $x=1,...,9$ οι στήλες και $y=1,...,9$ οι γραμμές. Προσέξτε ότι τιμές τύπου «x0» ή «0x» όπου $x=1,...,9$ δεν είναι έγκυρες. Επομένως το πρόγραμμα πρέπει να ελέγχει ότι ο χρήστης δίνει σωστές τιμές.

Το πρόγραμμα θα πρέπει να τοποθετεί σε 9 κελιά του πλέγματος τους δύο τύπους πλοίων με τυχαίο τρόπο (συγκεκριμένα 3 πλοία τύπου Patrol και 3 τύπου Cruiser). Η τοποθέτηση θα πρέπει να γίνεται με την ίδια σύμβαση όπως παραπάνω(τιμές 1 έως 9 για τον άξονα x και τιμές 1 έως 9 στον άξονα y). Προσέξτε έτσι ώστε το πρόγραμμά σας να μην τοποθετεί πλοία δύο φορές στο ίδιο πεδίο, κατά την τυχαία τοποθέτηση των πλοίων στο πλέγμα. Επίσης προσέξτε το πλοίο τύπου Patrol να έχει κενό στο διπλανό του κελί κάθετα και οριζόντια.

Outputs:

Το πρόγραμμα θα πρέπει να επιστρέφει σαν **έξοδο** τον δισδιάστατο πίνακα board, ο οποίος είναι το γραφικό του παιχνιδιού:

board () - επιστρέφει την τελική κατάσταση του σκάφους, με βάση την τοποθέτηση του πλοίου και τα αποτελέσματα των εικασιών του χρήστη, ως μήτρα 9x9. Το ακόλουθο σύνολο χαρακτήρων θα αντιπροσωπεύει το γραφικό του παιχνιδιού:

“B”= ένα κενό διάστημα στο ταμπλό.

“S” = ένας χώρος που καταλαμβάνεται από ένα πλοίο.

“M”= μια απώλεια (λάθος εικασία).

“H” = χτύπημα (σωστή εικασία).

Ο χρήστης θα μπορεί να ζητάει από το πρόγραμμα να του εμφανίσει τον **συνολικό αριθμό επιτυχιών** που έκανε ο χρήστης (σωστές εικασίες), είτε σε Patrols (e.g. hits) είτε σε Cruisers (e.g. sunks) και τον **συνολικό αριθμό πόντων** (e.g. points) που κέρδισε ο χρήστης (1 για κάθε χτύπημα, 2 για κάθε βυθισμένο Cruiser) για κάθε παιχνίδι.

Πώς θα παίζει το παιχνίδι ο χρήστης

Ο χρήστης θα δίνει 10 «μαντεψιές» στο πρόγραμμα σαν **input** και το πρόγραμμα σας θα τυπώνει τα αντίστοιχα **outputs** για κάθε παιχνίδι, όπως περιγράφονται παραπάνω. Κάθε φορά μετά το τέλος ενός παιχνιδιού ο χρήστης θα πρέπει να ερωτάται αν θέλει να παίζει καινούργιο παιχνίδι ή να τερματίζεται το πρόγραμμα.

Υποδείξεις:

- Το παιχνίδι θα παίζεται μέσω αντικειμένων της **public class Battleship**, ο constructor της οποία θα δέχεται σαν παραμέτρους τον πίνακα με τις «μαντεψιές» του χρήστη και τον πίνακα τοποθέτησης των πλοίων στο πλέγμα.
- Κάντε αποδοτική χρήση κλάσεων και αντικειμένων. Υποστηρίζοντας την αντικειμενοστραφή φύση της Java υλοποιήστε τις παραπάνω λειτουργίες χρησιμοποιώντας όσο το δυνατόν περισσότερα αντικείμενα.

Εκτέλεση προγράμματος

Η έξοδος του προγράμματος θα πρέπει να είναι η παρακάτω:

```
Oi manteipsies sou einai:  
[11, 12, 13, 14, 15, 16, 17, 18, 19, 21]  
O pinakas topothetisis twn ploiwn einai:  
[86, 87, 49, 36, 37, 18, 32, 33, 56]
```

```
O pinakas sou einai:  
M M M M M M H M  
M B B B B B B B  
B S S B B S S B B  
B B B B B B B S  
B B B B B S B B B  
B B B B B B B B  
B B B B B B B B  
B B B B B S S B B  
B B B B B B B B  
Hits:1  
Your sunks are:0  
Your points are:1  
Do you want to continue y/n?
```

ΑΞΙΟΛΟΓΗΣΗ

Για την αξιολόγηση του προγράμματος σας, θα ληφθούν οι πιο κάτω παράμετροι:

- **Ορθότητα:** ο πρόγραμμα σας πρέπει τρέχει ορθά για οποιαδήποτε είσοδο
- **Κατανοητά σχόλια:** Γράφετε κατανοητά σχόλια που να εξηγούν την λειτουργία της κάθε κλάσης/πεδίου/μεθόδου.
- **Σωστή χρήση προσδιοριστών πρόσβασης**
- **Σωστή χρήση κατασκευαστών**

ΟΔΗΓΙΕΣ

- Η κάθε κλάση θα πρέπει να υλοποιηθεί σε ξεχωριστό αρχείο JAVA.
- Στο πάνω μέρος της κάθε κλάσης να γράψετε με σχόλια το ονοματεπώνυμο και τον αριθμό ταυτότητας σας.
- Βεβαιωθείτε ότι τα προγράμματα σας είναι ορθά και τρέχουν.
- Οι άσκηση σας θα πρέπει να παραδοθεί σαν ένα zip αρχείο (που να περιλαμβάνει όλες τις κλάσεις που θα υλοποιήσετε και το έγγραφο τεκμηρίωσης) σε ηλεκτρονική μορφή στο eclass στην ημερομηνία παράδοσης.
- Το zip αρχείο θα είναι πρέπει να γίνει μέσω export από το eclipse σαν archive file.

Τμήμα Πληροφορικής & Τηλεπικοινωνιών, Πανεπιστήμιο Θεσσαλίας

- Το zip αρχείο θα ονομάζεται με τον εξής τρόπο:
ep1.onoma.epwnymo.omada_x<αριθμός μητρώου>.zip, όπου x η ομάδα
εργαστηρίου στην οποία ανήκετε.