



Εθνικό Μετσόβιο Πολυτεχνείο  
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών  
Υπολογιστών  
Εξάμηνο 7ο

## Τελική εργασία στο μάθημα Τεχνολογία Πολυμέσων

### Περιγραφή Εφαρμογής

Στα πλαίσια της εργασίας θα υλοποιηθεί μια έκδοση του κλασσικού παιχνιδιού «Ναυμαχία», όπου ο χρήστης θα παίζει με αντίπαλο τον υπολογιστή. Το παιχνίδι τελειώνει όταν βυθιστούν όλα τα πλοία ενός από τους παίκτες ή εναλλακτικά μετά από συγκεκριμένο αριθμό βολών

#### 1. Σχεδιασμός και υλοποίηση λογικής (55%)

Τυπικά, το παιχνίδι παίζεται χρησιμοποιώντας 4 ταμπλό , δύο για κάθε παίκτη. Στο ένα ταμπλό κάθε παίκτης τοποθετεί τα πλοία του και καταγράφει τις βολές του αντιπάλου και στο δεύτερο ταμπλό καταγράφει τις δικές του βολές.

Στην εφαρμογή θα υπάρχει ένας παίκτης που παίζει με αντίπαλο τον υπολογιστή. Η γραφική διεπαφή που θα υλοποιηθεί στη συνέχεια θα παρουσιάζει τα δύο ταμπλό του παίκτη, ενώ για τον αντίπαλο (υπολογιστής) δεν θα υπάρχει γραφική απεικόνιση των δικών του ταμπλό. Επιπλέον, θεωρούμε ότι τα ταμπλό που μοντελοποιούν το χώρο της ναυμαχίας αντιστοιχτούν σε ένα πλέγμα 10 γραμμών και 10 στηλών.

Πριν ξεκινήσει το παιχνίδι, έχουμε την τοποθέτηση των πλοίων στο βασικό ταμπλό κάθε παίκτη. Κάθε πλοίο καταλαμβάνει έναν αριθμό από συνεχόμενα κελία ενώ τα πλοία τοποθετούνται είτε κάθετα είτε οριζόντια και δεν μπορούν να καταλαμβάνουν διαγώνια κελία στο ταμπλό. Το πλήθος των κελιών που καταλαμβάνει κάθε πλοίο εξαρτάται από τον τύπο του. Κάθε κελί μπορεί να καταλαμβάνεται το πολύ από ένα πλοίο ενώ κάθε παίκτης έχει στη διάθεση του ένα πλοίο από την κάθε κατηγορία. Θεωρούμε 5 διαφορετικούς τύπους πλοίων τα χαρακτηριστικά των οποίων συνοψίζονται στον παρακάτω πίνακα.

Πίνακας 1 Χαρακτηριστικά διαθέσιμων πλοίων

Τύπος πλοίου	Πλήθος θέσεων	Πόντοι εύστοχης βολής	Bonus βύθισης
Carrier	5	350	1000
Battleship	4	250	500
Cruiser	3	100	250
Submarine	3	100	0
Destroyer	2	50	0

Κάθε πλοίο βρίσκεται σε μια από τις παρακάτω καταστάσεις:

- Ανέπαφο: Δεν έχει δεχθεί απολύτως καμία βολή.
- Χτυπημένο: Έχει δεχθεί τουλάχιστον μια βολή αλλά δεν έχει ακόμη χτυπηθεί σε όλες τις θέσεις και παραμένει στο παιχνίδι.
- Βυθισμένο: Έχει χτυπηθεί σε όλες τις θέσεις που καταλαμβάνει στο ταπλό, σε αυτή την περίπτωση βγαίνει εντελώς εκτός παιχνιδιού.

Η εφαρμογή θα πρέπει να σχεδιαστεί με γενικό τρόπο ώστε να μπορεί να υποστηρίζει διαφορετικές περιγραφές για την διάταξη των πλοίων του παίκτη και του υπολογιστή. Θεωρούμε πως υπάρχει ένας προκαθορισμένος φάκελος **“medialab”** που περιλαμβάνει μια σειρά από αρχεία περιγραφής που παρέχουν τα απαραίτητα στοιχεία. Το αρχείο με την περιγραφή της διάταξης των πλοίων για το παίκτη θα πρέπει να ονομάζεται **“player\_SCENARIO-ID.txt”** και το αντίστοιχο αρχείο για τον υπολογιστή **“enemy\_SCENARIO-ID.txt”**.

Θα πρέπει επίσης να κατασκευαστούν μέθοδοι που ελέγχουν αν η τοποθέτηση τηρεί τους παρακάτω περιορισμούς:

- Ένα πλοίο δεν μπορεί να βγαίνει εκτός των ορίων του ταμπλό.
- Ένα πλοίο δεν μπορεί να τοποθετηθεί σε κελί που ήδη έχει άλλο πλοίο.
- Ένα πλοίο δεν μπορεί να εφάπτεται κάθετα ή οριζόντια με κανένα άλλο πλοίο, έστω και για ένα κελί. Αυτό σημαίνει ότι τα πλοία έχουν μεταξύ τους απόσταση τουλάχιστον ένα ελεύθερο κελί.
- Δεν μπορούν να υπάρχουν περισσότερα από ένα πλοία για κάθε τύπο.

Για κάθε μία από τις παραπάνω περιπτώσεις μπορείτε να ορίσετε τον αντίστοιχο τύπο εξαίρεσης: **OversizeException**, **OverlapTilesException**, **AdjacentTilesException**, **InvalidCountException**.

Στη σελίδα του εργαστηρίου μαζί με την εκφώνηση μπορείτε να βρείτε το αρχείο **“examples.rar”** που περιλαμβάνει ολοκληρωμένα παραδείγματα περιγραφής, καθώς και σημειώσεις για την σύνταξη των σχετικών αρχείων περιγραφής.

Μετά την αρχικοποίηση με την κατάλληλη τοποθέτηση των πλοίων το παιχνίδι συνεχίζεται με μια σειρά από γύρους. Αρχικά, η εφαρμογή θα επιλέγει τυχαία αν την πρώτη κίνηση σε κάθε γύρο την κάνει ο παίκτης ή ο υπολογιστής. Η επιλογή γίνεται πριν το πρώτο γύρο και διατηρείται για όλη την διάρκεια του παιχνιδιού, ενώ θα πρέπει να ενημερώνεται κατάλληλα ο χρήστης με κάποιο μήνυμα στην γραφική διεπαφή.

Ο παίκτης θα ορίζει το κελί «στόχο» μέσω της γραφικής διεπαφής. Η εφαρμογή με βάση την διάταξη των πλοίων του αντιπάλου, που θα διατηρεί εσωτερικά, θα αναγνωρίζει αν η βολή ήταν επιτυχής ή όχι. Στην περίπτωση επιτυχούς βολής ο χρήστης κερδίζει τους πόντους εύστοχης βολής που αντιστοιχούν στο τύπο του πλοίου ενώ αν η βολή έχει ως αποτέλεσμα τη βύθιση του πλοίου τότε επιπρόσθετα κερδίζει και τους πόντους που

αντιστοιχούν στο bonus βύθισης. Ακόμη, μετά από κάθε βολή θα πρέπει να ενημερώνεται κατάλληλα η γραφική διεπαφή.

Ο υπολογιστής αρχικά θα πραγματοποιεί τυχαίες βολές αλλά όταν έχει κάποια εύστοχη θα προσπαθεί να κάνει βολές σε γειτονικά κελιά, όπως θα έκανε ο χρήστης. Μετά από κάθε βολή του υπολογιστή θα πρέπει να ανανεώνεται κατάλληλα η τρέχουσα κατάσταση και η γραφική διεπαφή.

Τέλος, θεωρούμε πως κάθε παίκτης έχει στη διαθέση του μέχρι 40 βολές. Το παιχνίδι ολοκληρώνεται είτε όταν έχουν βυθιστεί όλα τα πλοία ενός παίκτη είτε όταν έχουν συμπληρωθεί όλες οι διαθέσιμες βολές για τους παίκτες. Στη πρώτη περίπτωση, νικητής είναι ο παίκτης που κατάφερε να βυθίσει όλα τα πλοία του αντιπάλου. Στη δεύτερη περίπτωση, νικητής είναι ο παίκτης που έχει συγκεντρώσει την μεγαλύτερη βαθμολογία.

## **B.2. Δημιουργία γραφικής διεπαφής (25%)**

Θα πρέπει να σχεδιάσετε και να υλοποιήσετε την κατάλληλη γραφική διεπαφή χρήστη (Graphical User Interface - GUI) χρησιμοποιώντας το πλαίσιο JavaFX [1][2].

**Σημείωση:** Στη συνέχεια παρουσιάζονται οι βασικές προδιαγραφές για την γραφική διεπαφή, για όλες τις λεπτομέρειες της τελικής υλοποίησης μπορείτε να κάνετε όποιες επιλογές θέλετε σχετικά με την εμφάνιση και τη γενικότερη αλληλεπίδραση του χρήστη με την εφαρμογή, χωρίς καμία επίπτωση στην τελική βαθμολογία. Για παράδειγμα, μπορείτε να επιλέξετε μια απλή απεικόνιση για τα διάφορα στοιχεία χρησιμοποιώντας διαφορετικά χρώματα και κείμενο ή να συνδυάσετε εικόνες με διάφορα χαρακτηριστικά από το JavaFX ώστε να δημιουργήσετε ένα αποτέλεσμα που να αντιστοιχεί σε μια σύγχρονη εφαρμογή. Σε κάθε περίπτωση, δεν υπάρχει λόγος να κάνετε υπερβολικά πολύπλοκο το συγκεκριμένο μέρος της εργασίας.

Για την δημιουργία του γραφικού περιβάλλοντος της εφαρμογής θα πρέπει να ακολουθήσετε τις παρακάτω γενικές οδηγίες:

1. Δημιουργήστε το κεντρικό «παράθυρο» της εφαρμογής με τίτλο “MediaLab Battleship” και ορίστε τις κατάλληλες διαστάσεις.
2. Χωρίστε το παράθυρο σε τρία βασικά μέρη.
3. Στο πάνω μέρος της οθόνης θα εμφανίζονται συγκεντρωτικές πληροφορίες:
  - a. Πλήθος ενεργών πλοίων κάθε παίκτη
  - b. Συνολικοί πόντοι κάθε χρήστη
  - c. Ποσοστό επιτυχών βολών κάθε χρήστη
4. Στο μεσαίο τμήμα που θα αποτελείται από δύο επιμέρους τμήματα:
  - a. Το πρώτο, που θα βρίσκεται αριστερά, θα αντιστοιχεί στη γραφική παρουσίαση του ταμπλό του παίκτη που καταγράφονται οι θέσεις που βρίσκονται τα πλοία του και οι θέσεις που έχει κάνει βολές ο αντίπαλος. Θα πρέπει να περιλαμβάνει την αρίθμηση των γραμμών και στηλών και

να παρουσιάζει με ευδιάκριτο τρόπο τα σημεία που έχει κάνει βολές ο αντίπαλος και το αποτέλεσμα της κάθε βολής. Υπάρχει πλήρης ελευθερία για τον τρόπο παρουσίασης των συγκεκριμένων πληροφοριών καθώς και πιθανών επιπρόσθετων πληροφοριών της επιλογής σας. Τα περιεχόμενα θα πρέπει να ενημερώνονται αυτόματα μετά από κάθε βολή του αντιπάλου και ανάλογα με το αποτέλεσμα της.

- b. Το δεύτερο, που θα βρίσκεται δεξιά, θα αντιστοιχεί στη γραφική παρουσίαση του ταμπλό του παίκτη που καταγράφει τις βολές του. Θα πρέπει να περιλαμβάνει την αρίθμηση των γραμμών και στηλών και να απεικονίζει με ευδιάκριτο τρόπο τα σημεία που ο παίκτης έχει επιχειρήσει βολές και το αποτέλεσμα της κάθε βολής. Υπάρχει πλήρης ελευθερία για τον τρόπο παρουσίασης των πληροφοριών. Τα περιεχόμενα θα πρέπει να ενημερώνονται αυτόματα μετά από κάθε βολή του χρήστη και ανάλογα με το αποτέλεσμα της.

5. Στο κάτω μέρος της οθόνης:

- a. Θα υπάρχει η κατάλληλη φόρμα που θα επιτρέπει στον παίκτη να ορίζει το στόχο για την επόμενη βολή του και να την εκτελεί. Μετά την εκτέλεση της βολής θα πρέπει να ανανεώνονται τα κατάλληλα πεδία της γραφικής διεπαφής.

6. Προσθέστε ένα menu bar που θα περιλαμβάνει:

- a. Menu “Application” με επιλογές:
  - i. Start: Έναρξη νέας εκτέλεσης με βάση το επιλεγμένο σενάριο. Αν υπάρχει κάποια ενεργή εκτέλεση θα πρέπει να διακόπτεται και να αρχίζει μια νέα με κατάλληλη αρχικοποίηση όλων των παραμέτρων και των πληροφοριών που υπάρχουν στην γραφική διεπαφή.
  - ii. Load: Μέσω ενός popup παραθύρου ο χρήστης θα προσδιορίζει το “SCENARIO-ID”. Στη συνέχεια η εφαρμογή θα αναζητά τα αντίστοιχα αρχεία (player\_SCENARIO-ID.txt, enemy\_SCENARIO-ID.txt). Αν οι περιγραφές είναι σωστές (δεν προκύπτουν εξαιρέσεις) θα γίνεται η κατάλληλη αρχικοποίηση της εφαρμογής, διαφορετικά θα εμφανίζεται το κατάλληλο μήνυμα σφάλματος.
  - iii. Exit: Τερματισμός εφαρμογής.
- b. Menu “Details” με επιλογές:
  - i. Enemy Ships: Μέσω ενός popup παραθύρου θα παρουσιάζει την κατάσταση για όλα τα πλοία του αντιπάλου.
  - ii. Player Shots: Μέσω ενός popup παραθύρου θα παρουσιάζει για τις 5 τελευταίες βολές του χρήστη τις παρακάτω πληροφορίες: συντεταγμένες στόχου, αποτέλεσμα βολής και τύπο πλοίου σε περίπτωση εύστοχης βολής.

- iii. Enemy Shots: Μέσω ενός popup παραθύρου θα παρουσιάζει για τις 5 τελευταίες βολές του αντιπάλου τις παρακάτω πληροφορίες: συντεταγμένες στόχου, αποτέλεσμα βολής και τύπο πλοίου σε περίπτωση εύστοχης βολής.

### B.3. Λοιπές απαιτήσεις (20%)

- Η υλοποίηση θα πρέπει να ακολουθεί τις αρχές σχεδίασης του αντικειμενοστραφούς προγραμματισμού (OOP design principles).
- Σε μια κλάση της επιλογής σας θα πρέπει κάθε public μέθοδος που περιέχει να είναι τεκμηριωμένη σύμφωνα με τις προδιαγραφές του εργαλείου javadoc [3].

**Σημείωση:** Για οτιδήποτε δεν είναι ξεκάθαρο από την εκφώνηση μπορείτε να κάνετε τις δικές σας παραδοχές και υποθέσεις. Γενικότερα, στην εκφώνηση περιγράφονται οι βασικές απαιτήσεις που πρέπει να ακολουθεί η εφαρμογή σας, ωστόσο μπορείτε να κάνετε δικές σας σχεδιαστικές παραδοχές προσπαθώντας να κάνετε πιο ρεαλιστική την εφαρμογή, χωρίς παράλληλα να κάνετε πολύπλοκη την υλοποίηση.

### Παραδοτέα

- Το project (του IDE της επιλογής σας) με την υλοποίηση της εφαρμογής.
- Μια σύντομη αναφορά που θα περιέχει το σχεδιασμό της υλοποίησης, θα αναφέρετε όποια λειτουργικότητα δεν έχετε υλοποιήσει καθώς και όποιες επιπρόσθετες παραδοχές έχετε κάνει. Σε καμία περίπτωση δεν θα πρέπει να συμπεριλάβετε τμήματα κώδικα.

### Αναφορές

[1] <https://docs.oracle.com/javase/8/javafx/get-started-tutorial/jfx-overview.htm>

[2] [https://docs.oracle.com/javafx/2/get\\_started/jfxpub-get\\_started.htm](https://docs.oracle.com/javafx/2/get_started/jfxpub-get_started.htm)

[3] <https://www.oracle.com/technical-resources/articles/java/javadoc-tool.html>