

## Προγραμματιστική άσκηση #2

**Προθεσμία παράδοσης:** Παρασκευή 20 Ιουνίου (στο Εργαστήριο Η/Υ). Μετά τη λήξη της προθεσμίας, δεν θα γίνονται δεκτές εργασίες.

**Η άσκηση μπορεί να υλοποιηθεί από έναν ή δύο φοιτητές.** Οι φοιτητές μπορούν να συζητήσουν μεταξύ τους θέματα που αφορούν την άσκηση αλλά δεν επιτρέπεται να αντιγράψει η μία ομάδα την λύση (ή μέρη της λύσης) της άλλης ομάδας. Για απορίες να συμβουλευέστε τον διδάσκοντα και τον υπεύθυνο του εργαστηρίου σας.

**Η άσκηση θα υπολογιστεί ως το 5% του τελικού βαθμού.**

**Στη σελίδα του μαθήματος διατίθεται πρότυπο εκτελέσιμο πρόγραμμα που υλοποιεί την τμήμα της άσκηση. Χρησιμοποιείστε το για να κατανοήσετε την λειτουργία των κλάσεων της άσκησης.**

**Θέμα:** α) Μερική μοντελοποίηση του παιχνιδιού «σκάκι»  
β) Υλοποίηση γραφικού περιβάλλοντος επικοινωνίας

### A. Μερική μοντελοποίηση του παιχνιδιού «σκάκι»

Σκοπός της παρούσας άσκησης είναι η μερική μοντελοποίηση και ανάπτυξη του παιχνιδιού «σκάκι». Οδηγίες για το πως παίζεται το σκάκι μπορείτε να βρείτε στο διαδίκτυο. Το σημαντικό για την άσκηση είναι να γνωρίζεται το ποίες είναι οι νομικές κινήσεις καθώς και την αρχική τοποθέτηση των πιονιών στην σκακιέρα.

Το πρόγραμμα σας δεν θα παίζει σκάκι με έναν παίκτη. Το μόνο που θα κάνει είναι να παρέχει την σκακιέρα έτσι ώστε δυο παίκτες να μπορούν να την χρησιμοποιήσουν για να παίξουν μεταξύ τους. Μετά από κάθε κίνηση το πρόγραμμα ζωγραφίζει την σκακιέρα με τις νέες θέσεις των πιονιών. Η σκακιέρα ζωγραφίζεται στο τερματικό «ως κείμενο» με την χρήση `System.out.println()` εντολών.

Η παρούσα άσκηση αποσκοπεί στην εξάσκηση με την κληρονομικότητα καθώς και την εξοικείωση σας με πιο σύνθετα προγράμματα.

Μία προτεινόμενη υλοποίηση περιλαμβάνει τις παρακάτω κλάσεις:

- **Pioni** Υλοποιεί ένα γενικό πiónι. Είναι αφηρημένη κλάση. Για κάθε πiónι ορίζονται: το χρώμα του, οι θέση του στην σκακιέρα, και ο τύπος του (στρατιώτης, βασιλιάς, βασίλισσα, πύργος, αξιωματικός, άλογο). Η κλάση Pioni περιλαμβάνει επίσης τις μεθόδους:
  - `public abstract boolean isLegalMove(char x, int y, ChessBoard b)`; Ελέγχει το εάν η κίνηση προς την θέση που προσδιορίζει η παράμετρος είναι νόμιμη. Η παράμετρος `b` προσδιορίζει την σκακιέρα στην οποία βρίσκεται το πiónι έτσι ώστε να έχει μέσω του `b` πρόσβαση και στη θέση των άλλων πιονιών.
  - `public String print()` Τυπώνει μια αναπαράσταση του πιονιού χρησιμοποιώντας ένα μόνο γράμμα (p:στρατιώτης, k:βασιλιάς, q:βασίλισσα, r:πύργος, n: άλογο, b, στρατηγός). Η σύμβαση είναι ότι τα λευκά πiónια τυπώνονται ως κεφαλαία γράμματα ενώ τα μαύρα πiónια ως πεζά γράμματα.
  - `public void setPosition(char x, int y)` Θέτει τη θέση του πιονιού με βάση τις παραμέτρους.
- **Vasilias, Vasilissa, Pyrgos, Alogo, Stratigos, Stratiotis** Κληρονομούν από το Pioni και υλοποιούν την μέθοδο `isLegalMove()`. Η υλοποίηση της μεθόδου είναι το πιο σημαντικό τμήμα του κώδικα.
- **ChessBoard** Υλοποιεί μια σκακιέρα. Υποστηρίζει τις μεθόδους:
  - `public void placePioniAt(Pioni p, char xPos, int yPos)` Τοποθετεί ένα πiónι στη σκακιέρα
  - `public void loadBoard()` Βάζει τα πiónια στην αρχική τους θέση.
  - `public Pioni getPioniAt(char xPos, int yPos)` Επιστρέφει το πiónι στην συγκεκριμένη θέση
  - `public void move(char xOrig, int yOrig, char xDest, int yDest)` Μετακινεί ένα πiónι, υποθέτοντας ότι οι παράμετροι προσδιορίζουν μια νόμιμη κίνηση.
  - `public void printBoard()` Τυπώνει την σκακιέρα με τα περιεχόμενά της.
- **ChessEngine** Η μηχανή η οποία συντονίζει το παιχνίδι του σκακιού ανάμεσα στους 2 παίκτες. Υποστηρίζει τις μεθόδους:
  - `public static void printNotation()` Τυπώνει την ορολογία που χρησιμοποιείται στην απεικόνιση της σκακιέρας
  - `public void playChess()` Ξεκινάει το παιχνίδι.

- `public void nextMove(char xOrig, int yOrig, char xDest, int yDest)`. Καλεί τον παίκτη να κάνει την επόμενη κίνηση. Ελέγχει ότι η κίνηση είναι νόμιμη. Εάν όντως είναι νόμιμη την εκτελεί και καλεί τον άλλο παίκτη να παίξει. Εάν δεν είναι νόμιμη την αγνοεί και καλεί τον ίδιο παίκτη να παίξει ξανά.
- **Utilities** Μετατρέπει τις συντεταγμένες με μορφή γραμμάτων (A . . . Z) σε συντεταγμένες με μορφή αριθμών (1 . . . 8). Υποστηρίζει τις μεθόδους:
  - `public static char int2Char(int xCharPos)`
  - `public static int char2Int(char xCharPos)`

**Να γράφει κώδικας Java που υλοποιεί το παιχνίδι σκάκι όπως περιγράφηκε παραπάνω.** Ελέγξτε την λειτουργία του προγράμματος σας. Η έξοδος που θα παραχθεί θα πρέπει να είναι αντίστοιχη με αυτή του πρότυπου εκτελέσιμου προγράμματος.

**[Προαιρετικά]** Να σχεδιαστεί και να υλοποιηθεί εφαρμογή η οποία μέσω κατάλληλου γραφικού περιβάλλοντος επικοινωνίας (GUI) επιτρέπει σε δύο παίκτες να παίξουν σκάκι σε μία σκακιέρα που ελέγχει ο υπολογιστής. Ο υπολογιστής γνωρίζει και ελέγχει το ποιος παίκτης έχει σειρά για την επόμενη κίνηση, ελέγχει την νομιμότητα της κίνησης, αφαιρεί από την σκακιέρα πιόνια τα οποία έχουν «κερδηθεί», μετράει τον χρόνο που είναι διαθέσιμος στο παίκτη για την επόμενη κίνηση του, κλπ. Οι παραπάνω λειτουργικότητες, να υλοποιηθεί σταδιακά (όση μπορείτε και προλαβαίνετε), με τις βασικότερες λειτουργίες πρώτες. Στο διαδίκτυο είναι διαθέσιμα πολλά προγράμματα σκακιού από τα οποία μπορείτε να πάρετε ιδέες για την μορφή του GUI και τις διαθέσιμες λειτουργίες.

## **B. Υλοποίηση γραφικού περιβάλλοντος επικοινωνίας**

Να σχεδιαστεί και να υλοποιηθεί απλή «παραθυρική» εφαρμογή «υπενθύμισης γεγονότων / εργασιών». Η εφαρμογή πρέπει να παρέχει την δυνατότητα (κατ' ελάχιστο) να εισαχθεί ένα γεγονός στην λίστα των γεγονότων που εμφανίζεται στην οθόνη και να διαγραφεί όταν αυτό έχει ολοκληρωθεί. Επιπλέον δυνατότητες μπορεί να περιλαμβάνουν: την κατηγοριοποίηση σε απλά γεγονότα (events / θα συμβούν απλώς σε κάποια ημερομηνία /ώρα, πχ. γενέθλια) ή εργασίες (to-do items, έχουν προθεσμία εκτέλεσης, πχ. παράδοση προγραμματιστικής εργασίας) και τον έλεγχο για το εάν έχει παρέλθει η σχετική ημερομηνία, την προβολή των εργασιών/ γεγονότων που έχουν ολοκληρωθεί, την δυνατότητα ταξινόμησης των γεγονότων/εργασιών μέσω προτεραιοτήτων, την προβολή σε διαφορετικά παράθυρα των εργασιών και των γεγονότων, κλπ.

### **Στα προγράμματα-projects (2) που θα αναπτύξετε:**

- **Η κλάσεις να είναι κατάλληλα τεκμηριωμένες**, δηλαδή να περιέχουν σχόλια που να εξηγούν τη χρήση κάθε πεδίου τους, και τη λειτουργία κάθε μεθόδου τους.
- **Να συμπληρωθεί** το αρχείο (README.TXT) που αυτόματα δημιουργεί το BlueJ με τα στοιχεία του προγραμματιστή ή τα στοιχεία των μελών της ομάδας [δύο άτομα] (όνομα, αριθμό μητρώου, και ημερομηνία) .

**Το παραδοτέο σας** για την άσκηση αυτή είναι **φάκελος** που αποτελείται από:

1. **δισκέτα** που περιέχει τα έργα (projects) που υλοποιήσατε με το BlueJ. Από τη δισκέτα ο υπεύθυνος του εργαστηρίου θα πρέπει να μπορεί να δει τον κώδικα που γράψατε και να τον εκτελέσει. Στο αρχείο με τα σχετικά με το έργο σχόλια, καθώς και σε κάθε κλάση, θα πρέπει να αναγράφεται το όνομα και ο αριθμός μητρώου των μελών της ομάδας.
2. **εκτύπωση** του κώδικα κάθε κλάσης .

**Ο φάκελος, η δισκέτα, και οι εκτυπώσεις να αναγράφουν το όνομα και τον αριθμό μητρώου των μελών της ομάδας.**