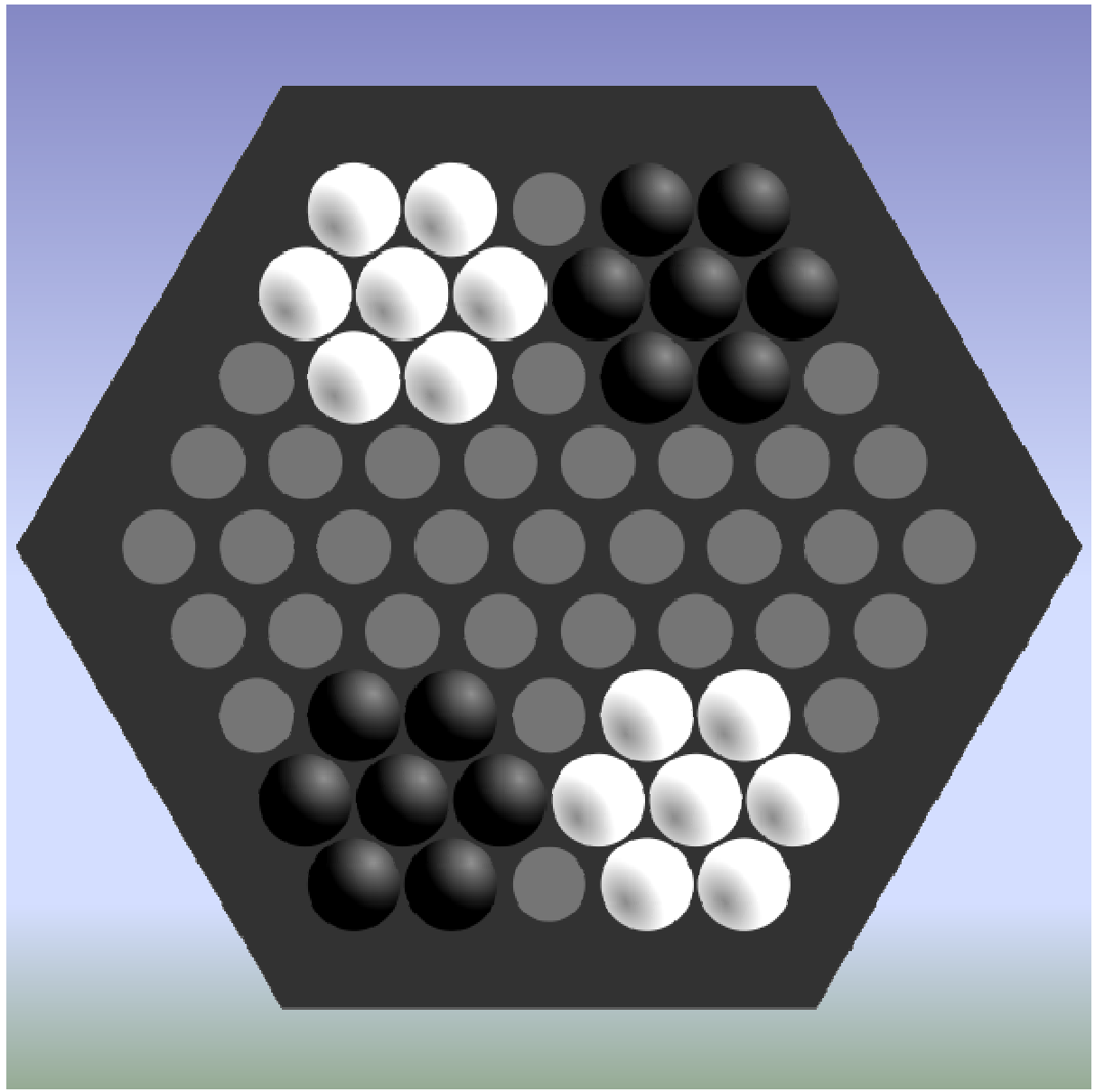


Δομές Δεδομένων – Εργασία Α΄

ABALONE

*Belgian daisy starting position



Περιγραφή του προβλήματος:

Σκοπός του προβλήματος ήταν η δημιουργία της κλάσης «Cell», η οποία αναπαριστά ένα κελί στο ταμπλό του παιχνιδιού.

Η συγκεκριμένη κλάση, σε συνδυασμό με τις υπόλοιπες κλάσεις που μας δοθήκαν θα αναπαριστά το στήσιμο των πουλιών σε σχηματισμό «βελγικής μαργαρίτας».

Περιγραφή του αλγορίθμου:

Για την επίλυση του προβλήματος δημιουργήσαμε την κλάση «Cell».

Αρχικά, δηλώσαμε τις μεταβλητές που θα χρησιμοποιηθούν, τις *x*, *y* και *color* δηλαδή.

Στη συνέχεια, υλοποιήσαμε τις μεθόδους «setX», «setY», «setColor» οι οποίες χρησιμοποιούνται για την αλλαγή τιμών στις αντίστοιχες μεταβλητές.

Επιπλέον, δημιουργήσαμε τις μεθόδους «getX», «getY», «getColor» για την επιστροφή των αντίστοιχων μεταβλητών.

Κατόπιν, υλοποιήσαμε τον constructor της κλάσης, ο οποίος θέτει στο *x* και στο *y* της κλάσης τις τιμές των ορισμάτων και την τιμή *null* στη μεταβλητή *color*.

Επιπρόσθετα, δημιουργήσαμε τη μέθοδο «getNeighbors», η οποία δέχεται ως όρισμα έναν πίνακα τύπου «Cell 11x11» και επιστρέφει έναν πίνακα 6 κελιών, που περιέχει τα κελιά του πίνακα που είναι γειτονικά στο κελί της κλάσης.

Συγκεκριμένα :

| | |
|---|---------------------|
| 0 | Επάνω Αριστερό Κελί |
| 1 | Επάνω Δεξί Κελί |
| 2 | Δεξί Κελί |
| 3 | Κάτω Δεξί Κελί |
| 4 | Κάτω Αριστερό Κελί |
| 5 | Αριστερό Κελί |

Ο παραπάνω πίνακας περιγράφει το πώς βρίσκονται τοποθετημένα τα κελιά στον πίνακα 6 κελιών.

Θα πρέπει να αναφερθεί επίσης πως για να επιτύχουμε την ακεραιότητα των τιμών *x* και *y* στην συγκεκριμένη μέθοδο, χρησιμοποιήσαμε 2 νέες μεταβλητές, τις *temp1* και *temp2*.

Τέλος, υλοποιήσαμε τη μέθοδο `isInBoard`. Η μέθοδος ελέγχει εάν το κελί βρίσκεται εντός ή εκτός του πίνακα και επιστρέφει τις τιμές `true` αν βρίσκεται εντός ή την τιμή `false` αν βρίσκεται εκτός.

Καθώς όλα τα βήματα της Ά Εργασίας ολοκληρώθηκαν, η Java εφαρμογή πλέον «τρέχει» κανονικά, επιτρέποντας σου να παίξεις μία παρτίδα Abalone με τον εαυτό σου!!!