Τεχνητή Νοημοσύνη

Εργασία 1
Θέμα: Cross-bridge
ΑΜ: 3160074, 3170014, 3180230

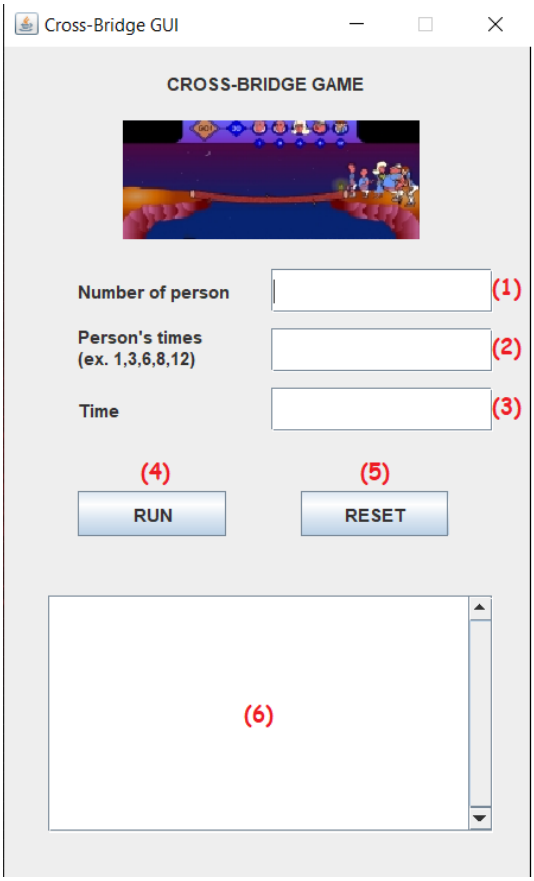
8/12/2020

1. Περιγραφή τρόπου χρήσης

- **Compile** : javac CrossFrame.java State.java Pair.java Searcher.java
- **Run** : java CrossFrame

Μετά το Run, θα εμφανιστεί στην οθόνη μας το CrossBridge GUI στο οποίο μπορούμε να δοκιμάσουμε οποιοδήποτε παράδειγμα.

Επεξήγηση GUI :



(1): Εισαγάγουμε το πλήθος των ατόμων που θέλουμε (π.χ.: 5).

(2): Εισαγάγουμε τους χρόνους των ατόμων χωρισμένους με κόμμα (",") (π.χ.: 1,3,6,8,12).

(3): Εισαγάγουμε τον μέγιστο χρόνο διάσχισης της γέφυρας (π.χ.: 30).

(4): Τρέχουμε το πρόγραμμα.

(5): Μηδενίζει τα στοιχεία για την δοκιμή ενός άλλου παραδείγματος.

(6): Εμφανίζει τα αποτελέσματα.

2. Δυνατότητες του προγράμματος

- Υπολογίζει το βέλτιστο μονοπάτι που πρέπει να ακολουθήσει, δηλαδή το ελάχιστο κόστος για τη διάσχιση της γέφυρας από τα άτομα, στα πλαίσια του δοσμένου χρόνου.
- Αναγνωρίζει αν το input είναι λανθασμένο και ενημερώνει κατάλληλα.
- Εάν, βάση του δοσμένου χρόνου δεν υπάρχει βέλτιστο μονοπάτι ενημερώνει αντίστοιχα.
- Ακόμη και για άτομα με ίδιους χρόνους το πρόγραμμα, μπορεί και βρίσκει το βέλτιστο μονοπάτι.

3. Αρχιτεκτονική Προγράμματος

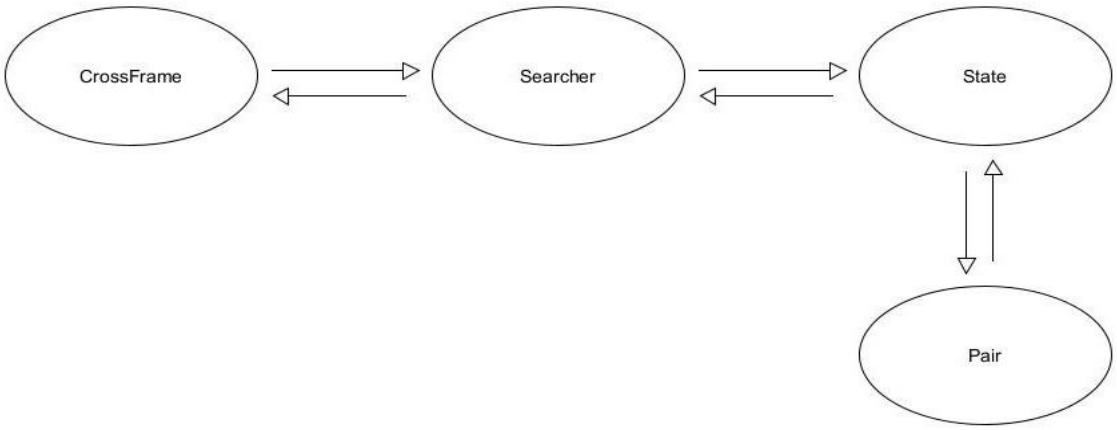
- **Class Pair** - δημιουργεί instances ζευγαριών που θα διασχίζουν την γέφυρα κάθε φορά. Τα επιστρέφει στη State.

- **Class Searcher** – Καλείται από τη CrossFrame. Η βασική της λειτουργία είναι να ελέγχει εάν είμαστε σε τερματική κατάσταση από τα παιδιά που της έχουν επιστραφεί από την κλάση State. Στο τέλος αποθηκεύει την τελική κατάσταση στο Path.

- **Class State** – Καλείται από την Searcher και χρησιμοποιεί την Pair. Βασική λειτουργία η εύρεση των παιδιών της κάθε κατάστασης. Επιστρέφει στην Searcher τα παιδιά.

- **Class CrossFrame** - Αποτελείται από την main() μας και δημιουργεί το GUI για την εύχρηστη χρήση του προγράμματος. Τέλος, εκτυπώνει τα αποτελέσματα με την print().

Η επικοινωνία των κλάσεων φαίνεται στην παρακάτω εικόνα:



4. Μέθοδοι AI που χρησιμοποιήθηκαν

Χρησιμοποιείται ο αλγόριθμος αναζήτησης A* με μια heuristic συνάρτηση (int heuristic ()), η οποία βρίσκεται στην State.java. Ουσιαστικά η A* αναζητά στην ταξινομημένη λίστα την επόμενη κατάσταση και για όσο υπάρχουν paths to explore. Αν ο χρόνος τελειώσει τότε σταματά την αναζήτηση. Σε κάθε άλλη περίπτωση είναι ένα βήμα πιο κοντά στο να χτίσει το μονοπάτι.

5. Παραδείγματα (CPU: i7700HQ, RAM: 12GB, JAVA: 14.0.1)