



## Ατομικές Εργασίες Ακαδημαϊκού Έτους 2021-2022

### Εργασία 2<sup>η</sup>: Criminal Network (CrimeNet)

Καλείστε να μοντελοποιήσετε σε αντικειμενοστρεφή κώδικα Java τις οντότητες που απαιτούνται για την υλοποίηση του κάτωθι απλοποιημένου και υποθετικού συστήματος παρακολούθησης της δραστηριότητας ενός δικτύου υπόπτων για εκληματικές πράξεις, καθώς και τις σχέσεις μεταξύ τους.

Κάθε παρακολουθούμενος ύποπτος (Suspect) του δικτύου CrimeNet χαρακτηρίζεται από το όνομά του, το κωδικοποιημένο όνομά του, την πόλη στην οποία δραστηριοποιείται καθώς και μια λίστα τηλεφωνικών αριθμών που χρησιμοποιεί. Ο κάθε ύποπτος συνδέεται με έναν ή περισσότερους από τους υπόλοιπους υπόπτους του δικτύου (πιθανοί συνεργάτες). Σύνδεση με έναν άλλο ύποπτο υπάρχει αν έχουν επικοινωνήσει τηλεφωνικά έστω και μία φορά.

Το σύστημα διατηρεί επίσης ένα κεντρικό αρχείο (Registry) με την καταγραφή όλων των επικοινωνιών (Communication) που μπορεί να είναι είτε τηλεφωνικές κλήσεις (PhoneCall) είτε μηνύματα (SMS). Το κεντρικό αρχείο διατηρεί επίσης μια λίστα με όλους τους υπόπτους.

Κάθε επικοινωνία χαρακτηρίζεται από τους δύο αριθμούς τηλεφώνου που εμπλέκονται και την ημερομηνία (ξεχωριστές ιδιότητες για το έτος, το μήνα και την ημέρα) που πραγματοποιήθηκε. Μια τηλεφωνική κλήση χαρακτηρίζεται επιπλέον από τη διάρκειά της (σε δευτερόλεπτα) και ένα SMS από το περιεχόμενο του μηνύματος.

- Όλες οι τηλεφωνικές επικοινωνίες πρέπει να διαθέτουν μέθοδο `printInfo()` εκτύπωσης όλων των ιδιοτήτων τους.

Κάθε ύποπτος παρέχει δημόσια τις εξής λειτουργίες-μεθόδους:

- Μέθοδο `addNumber(String number)` η οποία προσθέτει έναν αριθμό στη λίστα τηλεφωνικών αριθμών του υπόπτου.
- Μέθοδο η οποία προσθέτει έναν ύποπτο στη λίστα πιθανών συνεργατών ενός υπόπτου. Αυτό γίνεται αφού προηγηθεί έλεγχος για το αν πρόκειται για ύποπτο που είναι ήδη στη λίστα πιθανών συνεργατών.
- Μέθοδο `isConnectedTo(Suspect aSuspect)` η οποία δέχεται ως παράμετρο έναν άλλο ύποπτο και επιστρέφει `true` ή `false` ανάλογα με το αν είναι συνδεδεμένοι.
- Μέθοδο `getCommonPartners(Suspect aSuspect)` η οποία δέχεται ως παράμετρο έναν άλλο ύποπτο και **επιστρέφει** μια λίστα με τους κοινούς πιθανούς συνεργάτες των δύο υπόπτων.
- Μέθοδο η οποία τυπώνει τους πιθανούς συνεργάτες (ονόματα και κωδικά ονόματα) του υπόπτου.

Το κεντρικό αρχείο παρέχει δημόσια τις εξής λειτουργίες-μεθόδους:

- Μέθοδο `addSuspect(Suspect aSuspect)` καταχώρησης ενός υπόπτου.
- Μέθοδο `addCommunication(Communication aCommunication)` καταχώρησης μιας νέας επικοινωνίας. Αν η επικοινωνία αφορά τους τηλεφωνικούς αριθμούς `Num1` και `Num2`, τότε στους υπόπτους που διαθέτουν τους αντίστοιχους αριθμούς ενημερώνεται η λίστα των συνεργατών τους.
- Μέθοδο `getSuspectWithMostPartners()` η οποία να επιστρέφει τον ύποπτο με τους περισσότερους πιθανούς συνεργάτες.
- Μέθοδο `getLongestPhoneCallBetween(String number1, String number2)` η οποία να δέχεται ως παράμετρο δύο αριθμούς και να επιστρέφει την τηλεφωνική κλήση μεταξύ τους με τη μεγαλύτερη διάρκεια.
- Μέθοδο `getMessagesBetween(String number1, String number2)` η οποία να δέχεται ως παράμετρο δύο αριθμούς και να επιστρέφει όλα τα μηνύματα μεταξύ τους που περιέχουν οποιαδήποτε από τις λέξεις "Bomb", "Attack", "Explosives", "Gun".

Κατασκευαστές (constructors) μπορείτε να προσθέσετε ανάλογα με τις ανάγκες του προγράμματος που θα δημιουργήσετε.

**Να χρησιμοποιήσετε την κλάση Main που δίνεται** μαζί με την εκφώνηση στην οποία κατασκευάζονται αντικείμενα και εκτελούνται οι κατάλληλες εντολές για τον έλεγχο των λειτουργιών του προγράμματος.



Οι έλεγχοι πρέπει να αποδώσουν τα εξής αποτελέσματα (δεν υπάρχει πρόβλημα αν υπάρχει μικρή απόκλιση στον τρόπο εκτύπωσης των αποτελεσμάτων)

Test1: Bob Robson, Frozen Bear

Test2:

This phone call has the following info  
Between 00496955444444 --- 00478484777777  
on 2021/10/16  
Duration: 240

Test3:

This SMS has the following info  
Between 00478484777777 --- 00446999888888  
on 2021/10/14  
Text: Gun Received from Rusty Knife  
This SMS has the following info  
Between 00478484777777 --- 00446999888888  
on 2021/10/15  
Text: Metro Attack ready

Test4:

true  
true

Test5:

Danny Rust, Rusty Knife

Τα προγράμματα θα αξιολογηθούν με βάση τα παρακάτω κριτήρια:

- Αντικειμενοστρέφεια – Περιλαμβάνει το πρόγραμμα τις απαιτούμενες κλάσεις και σχέσεις μεταξύ τους για τη μοντελοποίηση των οντοτήτων που περιγράφονται;
- Λειτουργικότητα – Έχουν υλοποιηθεί σωστά οι μέθοδοι που ζητούνται; Εκτελείται ορθά η μέθοδος main();
- Γενική εντύπωση – Είναι το πρόγραμμα σωστά δομημένο?
- Σφάλματα – μεταγλωττίζεται το πρόγραμμα? Υπάρχουν προφανή σφάλματα?
- Εσωτερική τεκμηρίωση – Είναι εύκολα κατανοητός ο κώδικας? Χρησιμοποιείτε κατάλληλα ονόματα κλάσεων/ιδιοτήτων/μεθόδων? Υπάρχουν σχόλια (στα σημεία που απαιτούνται)?
- Αναγνωσιμότητα – Χρησιμοποιείται στοίχιση που να ανακλά τη δομή του κώδικα? Είναι εύκολη η ανάγνωση του κώδικα? Υπάρχουν κενά που να διαχωρίζουν τα κύρια τμήματα του κώδικα?

Δειγματοληπτικά θα ζητηθεί από ορισμένους φοιτητές/φοιτήτριες που υπέβαλλαν εργασίες να εξηγήσουν τα προγράμματά τους στους διδάσκοντες.

Σημείωση περί αντιγραφής: Η συγκεκριμένη εργασία εμπίπτει στις ατομικές εργασίες του μαθήματος. Κατά συνέπεια θεωρείται ότι είναι προϊόν ατομικής προσπάθειας. Για τον λόγο αυτό, εργασίες που θα παρουσιάζουν ενδείξεις αντιγραφής τμημάτων κώδικα θα μηδενίζονται (για όλους τους εμπλεκόμενους)

### Οδηγίες Υποβολής

Υποβάλλετε συμπιεσμένο ολόκληρο το Eclipse project στην αντίστοιχη εργασία του eClass. Το όνομα του Eclipse Project θα πρέπει να έχει τη μορφή:

**Epwnymo\_Onoma\_ArithmosErgasias**

π.χ. **Papadopoulos\_Nikolaos\_2**