**ΟΔΗΓΙΕΣ**

* *Η εργασία γίνεται σε ομάδες 2 ατόμων.*
* *Η παράδοση της εργασίας είναι έως τις* ***20/06/2022****.*
* *Ο βαθμός της εργασίας μπορεί να κρατηθεί μέχρι και την εξεταστική του Σεπτεμβρίου 2022.*
* *Τα προγράμματά σας θα περιέχουν οπωσδήποτε σχόλια! Τα σχόλια αποτελούν μέρος του κώδικα και βαθμολογούνται.*
* *Η ΑΝΤΙΓΡΑΦΗ ΑΝΙΧΝΕΥΕΤΑΙ ΚΑΙ ΤΙΜΩΡΕΙΤΑΙ.*

**Α. (6 μονάδες)** Να αναπτυχθεί ένα πρόγραμμα το οποίο διαχειρίζεται τα σιδηροδρομικά δρομολόγια του ΟΣΕ. Θα ορίσετε τις κλάσεις **Train** (τρένο), **Route** (διαδρομή), **IndirectRoute** (δρομολόγιο με στάση για μετεπιβίβαση) και **Ticket** (εισητήριο)**.**

Η 1η κλάση **Train** (τρένο), έχει τα παρακάτω πεδία, κατασκευαστές και μεθόδους:

**Πεδία**

* **id**: Κωδικός του τρενου (ακέραιος).
* **model:** Τύπος του τρένου π.χ. SilverArrow, ηλεκτρικό, κ.α. (string).
* **maxPassenger:** Τον μέγιστο αριθμό επιβατών (ακέραιος).

**Κατασκευαστές**

* **χωρίς ορίσματα**: αρχικοποιεί όλα τα πεδία σε ουδέτερες τιμές ή κενά.
* **με ορίσματα**: έχει ως ορίσματα τιμές για όλα τα πεδία.

**Μέθοδοι**

* Συναρτήσεις **set** και **get** για όλα τα πεδία.
* **read()**: ζητάει από το χρήστη, ελέγχει και αποθηκεύει τα στοιχεία για το αντικείμενο που την καλεί.
* **String toString()**: επιστρέφει σε μορφή αλφαριθμητικού όλα τα πεδία της κλάσης.

Η 2η κλάση **Route** (διαδρομή) έχει τα παρακάτω πεδία, κατασκευαστές και μεθόδους:

**Πεδία**

* **id**: Κωδικός της διαδρομής (ακέραιος).
* **aeroplane:** Ο κωδικός του τρένου (ακέραιος).
* **departure:** Ο τόπος αναχώρησης (string).
* **arrival:** Ο τόπος προορισμού (string).
* **tickets:** ένα ArrayList με τα ticket (εισιτήρια) που αγοράστηκαν.

**Κατασκευαστές**

* **χωρίς ορίσματα**: αρχικοποιεί όλα τα πεδία σε ουδέτερες τιμές ή κενά.
* **με ορίσματα**: έχει ως ορίσματα τιμές για όλα τα πεδία.

**Μέθοδοι**

* Συναρτήσεις **set** και **get** για όλα τα πεδία.
* **addTicket()** για την εισαγωγή νέου εισιτηρίου στο δρομολόγιο του τρένου. Σε αυτή θα ελέγχετε και θα χειρίζεστε τις εξής εξαιρέσεις, εμφανίζοντας κατάλληλο μήνυμα: **α)** πρέπει να υπάρχουν διαθέσιμες θέσεις στο τρένο που εκτελεί τη διαδρομή, **β)** αν ο κωδικός δρομολογίου του εισιτηρίου είναι ίδιος με τον κωδικό του δρομολογίου και **γ)** η αφετηρία και ο προορισμός πρέπει να ταιριάζει με αυτές του δρομολογίου.
* **finalize()** η οποία θα εκτυπώνει όλα τα στοιχεία του δρομολογίου και θα ελέγχει αν έχει τουλάχιστον έναν επιβάτη για να μπορεί να πραγματοποιηθεί (θα εμφανίζει στην οθόνη σχετικό μήνυμα αν θα πραγματοποιηθεί η πτήση ή αν όχι καθώς το τρένο είναι άδειο από επιβάτες).
* **String toString()**: επιστρέφει σε μορφή αλφαριθμητικού όλα τα πεδία της κλάσης.

Η 3η κλάση **IndirectRoute** (δρομολόγιο με στάση για μετεπιβίβαση) έχει τα παρακάτω πεδία, κατασκευαστές και μεθόδους:

**Κληρονομικότητα**

* Θα κληρονομεί τη route – αποτελεί διαδρομή στην οποία θα προστίθεται και ένας ενδιάμεσος προορισμός όπου θα είναι δυνατή η επιβίβαση νέων επιβατών και η αποβίβαση υφιστάμενων. Προφανώς και ο πελάτης που αποβιβάζεται στην ενδιάμεση στάση

**Πεδία**

* **middlestop:** Ο τόπος της ενδιάμεσης στάσης (string).

**Κατασκευαστές**

* **χωρίς ορίσματα**: αρχικοποιεί όλα τα πεδία σε ουδέτερες τιμές ή κενά (και της μητρικής κλάσης flight με τη χρήση της super).
* **με ορίσματα**: έχει ως ορίσματα τιμές για όλα τα πεδία (και της μητρικής κλάσης flight με τη χρήση της super).

**Μέθοδοι**

* Συναρτήσεις **set** και **get** για όλα το νέο πεδίο middlestop.
* **addTicket()** για την εισαγωγή νέου εισιτηρίου στο δρομολόγιο από την ενδιάμεση στάση. Σε αυτή θα ελέγχετε και θα χειρίζεστε τις εξής εξαιρέσεις, εμφανίζοντας κατάλληλο μήνυμα: **α)** πρέπει να υπάρχουν διαθέσιμες θέσεις στο τρένο που εκτελεί την διαδρομή, **β)** αν ο κωδικός δρομολογίου του εισιτηρίου είναι ίδιος με τον κωδικό του δρομολογίου και **γ)** η αφετηρία και ο προορισμός πρέπει να ταιριάζει με αυτές του δρομολογίου.
* **finalize()** η οποία θα εκτυπώνει όλα τα στοιχεία του δρομολογίου και θα ελέγχει αν έχει τουλάχιστον έναν επιβάτη για να μπορεί να πραγματοποιηθεί (θα εμφανίζει στην οθόνη σχετικό μήνυμα αν θα πραγματοποιηθεί το δρομολόγιο ή αν όχι καθώς το τρένο είναι άδειο από επιβάτες στην ενδιάμεση στάση).
* **String toString()**: επιστρέφει σε μορφή αλφαριθμητικού όλα τα πεδία της κλάσης.

Η 4η κλάση **Ticket** (εισιτήρια) έχει τα παρακάτω πεδία, κατασκευαστές και μεθόδους:

**Πεδία**

* **id**: Κωδικός του δρομολογίου (ακέραιος).
* **passenger:** Το όνομα του επιβάτη (string).
* **departure:** Ο τόπος αναχώρησης (string).
* **arrival:** Ο τόπος προορισμού (string).
* **distance:** Η απόσταση του δρομολογίου σε χιλιόμετρα (float).
* **cost:** Το κόστος του δρομολογίου (float).

**Κατασκευαστές**

* **χωρίς ορίσματα**: αρχικοποιεί όλα τα πεδία σε ουδέτερες τιμές ή κενά.
* **με ορίσματα**: έχει ως ορίσματα τιμές για όλα τα πεδία.

**Μέθοδοι**

* Συναρτήσεις **set** και **get** για όλα τα πεδία.
* **read()**: ζητάει από το χρήστη, ελέγχει και αποθηκεύει τα στοιχεία για το αντικείμενο που την καλεί. Χειριστείτε με εξαίρεση την περίπτωση που θα δοθεί μηδενική ή αρνητική τιμή στο κόστος και τυπώστε το κατάλληλο μήνυμα. Το κόστος θα υπολογίζεται με βάση τη χιλιομετρική απόσταση και θα είναι ίσο με 0,25€ / χιλιόμετρο.
* **String toString()**: επιστρέφει σε μορφή αλφαριθμητικού όλα τα πεδία της κλάσης.

Στη συνάρτηση *main* θα κάνετε τα παρακάτω:

1. Θα δημιουργήσετε ένα ArrayList με 3 δρομολόγια:
   * 1 χωρίς ενδιάμεση στάση και όνομα d1,
   * 1 με ενδιάμεση στάση και όνομα d2, και
   * 1 χωρίς ενδιάμεση στάση και όνομα d3.
2. Θα εισάγετε 1 επιβάτη στο 1o δρομολόγιο d1.
3. Θα εισάγετε 1 επιβάτη στο 2o δρομολόγιο d2 που θα έχει χιλιομετρική απόστααση ίση με μηδέν.
4. Στο 3ο δρομολόγιο d3 δεν θα εισάγετε επιβάτη.
5. Θα εκτελέσετε τη **finalize** σε όλες τις πτήσεις ώστε να ενεργοποιηθούν τα σχετικά exceptions.

**Β. (4 μονάδες)** Να αναπτύξετε ένα πρόγραμμα το οποίο θα προσομοιώνει τη λειτουργία ενός καταστήματος το οποίο διαθέτει υπαλλήλους (employee) που εξυπηρετούν πελάτες (customers) με βάση το νούμερο αναμονής τους και εξυπηρέτηση αποκλειστικά με ραντεβού. Οι υπάλληλοι και οι πελάτες θα είναι δύο παράλληλα *threads*.

Η 1η κλάση **Appointments** (ραντεβού) έχει τα παρακάτω πεδία, κατασκευαστές και μεθόδους:

**Πεδία**

* **id**: O αριθμός του ραντεβού – θα πρέπει να είναι μοναδικός για κάθε ραντεβού και να αυξάνεται αυτόματα κατά ένα – δηλαδή αν έχουμε 3 ραντεβού, τότε θα έχουν id 1, 2 και 3 (ακέραιος αριθμός).
* **employee:** το όνομα του γιατρού (string).
* **customer:** το όνομα του πολίτη (string).

**Κατασκευαστές**

* **χωρίς ορίσματα**: αρχικοποιεί όλα τα πεδία σε ουδέτερες τιμές ή κενά.
* **με ορίσματα**: έχει ως ορίσματα τιμές για όλα τα πεδία.

**Μέθοδοι**

* Συναρτήσεις **set** και **get** για όλα τα πεδία.
* **read()**: ζητάει από το χρήστη, ελέγχει και αποθηκεύει τα στοιχεία για το αντικείμενο που την καλεί.
* **String toString()**: επιστρέφει σε μορφή αλφαριθμητικού όλα τα πεδία της κλάσης.

Η 2η κλάση **Employee** (υπάληλλος) έχει τα παρακάτω πεδία, κατασκευαστές και μεθόδους:

**Πεδία**

* **id**: O αριθμός μητρώου του υπαλλήλου – θα πρέπει να είναι μοναδικός για κάθε υπάλληλο και να αυξάνεται αυτόματα κατά ένα – δηλαδή αν έχουμε 3 υπαλλήλους, τότε θα έχουν id 1, 2 και 3 (ακέραιος αριθμός).
* **name**: Το όνομα του υπαλλήλου (string).

**Κατασκευαστές**

* **χωρίς ορίσματα**: αρχικοποιεί όλα τα πεδία σε ουδέτερες τιμές ή κενά.
* **με ορίσματα**: έχει ως ορίσματα τιμές για όλα τα πεδία.

**Μέθοδοι**

* Συναρτήσεις **set** και **get** για όλα τα πεδία.
* **read()**: ζητάει από το χρήστη, ελέγχει και αποθηκεύει τα στοιχεία για το αντικείμενο που την καλεί.
* **run():** για να προσομοιώσετε το χρόνο εξυπηρέτησης ενός πελάτη απο υπάλληλο θα βάζετε το thread που δημιουργείτε σε ύπνο για τυχαίο χρόνο μεταξύ 5 και 10 δευτερολέπτων (με τη sleep για την αναμονή και με τη random για τον τυχαίο χρόνο).
* **toString()**: επιστρέφει σε μορφή αλφαριθμητικού όλα τα πεδία της κλάσης.

Η 3η κλάση **Customer** (πελάτης) έχει τα παρακάτω πεδία, κατασκευαστές και μεθόδους:

**Πεδία**

* **name**: Το όνομα του πελάτη (string).

**Κατασκευαστές**

* **χωρίς ορίσματα**: αρχικοποιεί όλα τα πεδία σε ουδέτερες τιμές ή κενά.
* **με ορίσματα**: έχει ως ορίσματα τιμές για όλα τα πεδία.

**Μέθοδοι**

* Συναρτήσεις **set** και **get** για όλα τα πεδία.
* **read()**: ζητάει από το χρήστη, ελέγχει και αποθηκεύει τα στοιχεία για το αντικείμενο που την καλεί.
* **run():** ο υπάλλος θα πρέπει να καθυστερήσει λίγο να πάρει αριθμό ραντεβού βάζοντας το thread σε κατάσταση sleep για τυχαίο διάστημα μεταξύ 2 και 5 δευτερολέπτων (μικρότερο σε διάρκεια από το προηγούμενο, ώστε να μη συσσωρεύονται οι πελάτες και η εξυπηρέτηση είναι μικρότερη σε ρυθμό από τη συσσώρευση),
* **String toString()**: επιστρέφει σε μορφή αλφαριθμητικού όλα τα πεδία της κλάσης.

Στη συνάρτηση *main* θα κάνετε τα παρακάτω:

* Θα δημιουργήσετε ένα ArrayList με τα Appointments (ραντεβού)
* Θα δημιουργήσετε 20 customers (πελάτες) με ονόματα C1, C2, … C20
* Θα δημιουργήσετε 3 employees (υπαλλήλους) με ονόματα E1, E2, E3
* Θα χρησιμοποιήσετε threads για να προσομοιώσετε τη διαδικασία λήψης αριθμού ραντεβού από εισερχόμενους πελάτες και την εξυπηρέτησή τους από τους διαθέσιμους υπαλλήλους, και θα υπάρχει κατάλληλος συγχρονισμός, ώστε να δίνετε αποκλειστική πρόσβαση σε όσα δεδομένα μέλη απαιτείται.
* Με τη χρήση του ArrayList θα τυπώνετε όλα τα στοιχεία των ραντεβού.