



ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

Τμήμα Πληροφορικής

ΕΠΛ 133 – Αντικειμενοστραφής Προγραμματισμός

ΑΣΚΗΣΗ 1 – A small AI chat Agent

Διδάσκων: Καθ. Μάριος Δικαιάκος

Υπεύθυνος Άσκησης: Πύρρος Μπράτσкас

Ημερομηνία Ανάθεσης: 27 Ιανουαρίου 2023

Ημερομηνία Παράδοσης: 10 Φεβρουαρίου 2023 13:00

(ο κώδικας να υποβληθεί μέσω του Moodle)

<http://www.cs.ucy.ac.cy/courses/EPL133>

I. Στόχος

Στην εργασία αυτή θα ασχοληθούμε με αντικειμενοστραφή σχεδίαση προβλημάτων, σχεδίαση κλάσεων, μεθόδους, αντικείμενα, βρόγχους, ελέγχους, πίνακες, μορφοποιημένη είσοδο/έξοδο στην οθόνη. Η άσκηση αποτελείται από πολλαπλά μέρη. **Υλοποιήστε όλα τα μέρη σε αρχεία Java, όπου η κάθε κλάση να αντιπροσωπεύει ένα ξεχωριστό αντικείμενο.** Θα ξεκινήσουμε με την περιγραφή της εργασίας και στην συνέχεια θα παρουσιάσουμε τα ζητούμενα.

II. Περιγραφή

Εισαγωγή

Τους δυο τελευταίους μήνες, υπάρχει μια μεγάλη συζήτηση στην κοινότητα της Πληροφορικής, και όχι μόνο, σχετικά με τις δυνατότητες που προσφέρει το εργαλείο τεχνητής νοημοσύνης chatGPT.

Το ChatGPT (Generative Pre-trained Transformer - Παραγωγικός Προεκπαιδευμένος Μετασχηματιστής) είναι μια εφαρμογή διαλόγου-απαντήσεων με τεχνητή νοημοσύνη (chatbot) που κυκλοφόρησε από την εταιρία OpenAI τον Νοέμβριο του 2022. Βασίζεται στην οικογένεια μεγάλων γλωσσικών μοντέλων GPT-3.5 της OpenAI και είναι βελτιστοποιημένο τόσο με εποπτευόμενες όσο και με ενισχυτικές τεχνικές εκμάθησης.

Το ChatGPT κυκλοφόρησε ως πρωτότυπο στις 30 Νοεμβρίου 2022 και γρήγορα τράβηξε την προσοχή για τις λεπτομερείς απαντήσεις που παρείχε σε πολλά γνωστικά πεδία. Η ανομοιόμορφη πραγματική του ακρίβεια αναγνωρίστηκε ως σημαντικό μειονέκτημα.

Σε αυτή την εργασία θα δημιουργήσετε τρεις κλάσεις οι οποίες θα υλοποιούν ένα μικρό chatbot. Ακολουθεί η περιγραφή των κλάσεων αυτών.

III. Ζητούμενα της άσκησης

Σε αυτήν την ενότητα περιγράφονται οι κλάσεις που πρέπει να υλοποιήσετε. Σημειώστε ότι μπορείτε να προσθέσετε και άλλες μεθόδους ή άλλες κλάσεις εκτός από αυτές που σας ζητούνται από την εκφώνηση. Εννοείται ότι η δημιουργία άλλων μεθόδων/κλάσεων πρέπει να είναι δικαιολογημένη και σύμφωνη με τις αρχές του αντικειμενοστρεφούς προγραμματισμού.

Η κλάση `Agent` μοντελοποιεί ένα πράκτορα ο οποίος αναπαριστά ένα φοιτητή που σπουδάζει. Οι πράκτορες μπορούν να «επικοινωνούν» μεταξύ τους δημιουργώντας μια συνομιλία.

Η κλάση `Agent`

Αυτή η κλάση θα έχει κάποια χαρακτηριστικά, με τις τιμές για αυτά τα χαρακτηριστικά να επιλέγονται τυχαία. Εναπόκειται σε εσάς να επιλέξετε μια καλή αναπαράσταση δεδομένων. Δεδομένου ότι όλα τα χαρακτηριστικά είναι ιδιωτικά (private), δεν μας ενδιαφέρει ποιο τρόπο χρησιμοποιείτε για να αποθηκεύσετε τα δεδομένα, αλλά αυτό που έχει σημασία είναι ότι ένα στιγμιότυπο της κλάσης ΠΡΕΠΕΙ να αναπαριστά σωστά όλα αυτά τα δεδομένα.

Για παράδειγμα, το φύλο ενός πράκτορα (agent) θα μπορούσε να αναπαρασταθεί με διάφορους τρόπους, όπως αυτό που φαίνεται πιο κάτω:

1. Ένα `boolean`, `isFemale`, το οποίο είναι `true` ή `false` με βάση το φύλο.
2. Ένας χαρακτήρας όπου αποθηκεύεται η συντομογραφία όπως `M` ή `F`.
3. Μια συμβολοσειρά όπου αποθηκεύεται ολόκληρη τη συμβολοσειρά "αρσενικό" ή "θηλυκό".
4. κ.λπ.

Για καθένα από τα παρακάτω, θα πρέπει να σκεφτείτε ποιος είναι ο καλύτερος τρόπος για να τα αποθηκεύσετε. Θα πρέπει να βασίσετε αυτές τις αποφάσεις στο πώς θα χρησιμοποιήσετε αυτές τις μεταβλητές αργότερα.

- `gender` - Επιλέγεται τυχαία (άνδρας ή γυναίκα)
- `birthday` - Επιλέγεται τυχαία
- `name` - Εξαρτάται από το φύλο του πράκτορα και προέρχεται από το `WordList.java` (περιγράφεται στη σελίδα 2)
- `city of birth` - Επιλέγεται τυχαία από το `WordList.java`
- `city where they live now` - Επιλέγεται τυχαία από το `WordList.java`
- `currently mayor` - Επιλέγεται τυχαία από το `WordList.java`

Εκτός από αυτές τις ιδιότητες, πρέπει να παρέχετε διάφορες μεθόδους:

- `getCityNow()` : Returns the city the Agent currently lives in
- `getCityBorn()` : Returns the city the Agent was born in
- `getName()` : Returns the name of the Agent
- `getGender()` : Returns the gender of the Agent
- `getMajor()` : Returns the major of the Agent
- `getBirthday()` : Returns a String representing the date the Agent was born (e.g. 20th of July, 1985)

Επιπλέον, αφού πρόκειται για πράκτορες, πρέπει να τους «δώσετε φωνή» και να τους κάνετε να πουν πράγματα. Για αυτό, πρέπει να παρέχετε αυτές τις δημόσιες μεθόδους:

- `sayHello()` - `println(this.name + " says: Hello my dear")`
- `sayCityBorn()` - `println(this.name + " says: I am from " + this.cityBorn)`
- `howOldAreYou()` - This one has to be based on real time, not hard coded. Use `Calendar.getInstance().get(Calendar.YEAR)`
- `sayGender()`
- `sayCityNow()`
- `sayMajor()`

Για να δοκιμάσετε τον κώδικά σας, πρέπει να συμπεριλάβετε μια δημόσια μέθοδο `whoAreYou()` που εκτυπώνει το περιεχόμενο όλων των λιστών σας. Η μέθοδος `whoAreYou()` θα εκτυπώσει μια βιογραφία του πράκτορά σας. Αναμένουμε να μάθουμε το όνομά τους, το φύλο, την ηλικία, την τρέχουσα πόλη, όπου γεννήθηκαν και την τρέχουσα ειδικότητά τους.

Output example from calling `whoAreYou()`:

```
Jean says: "Bonjour, my name is Jean!"
Jean says: "I am from Baie-Comeau."
Jean says: "I am 101 years old."
Jean says: "I am majoring in Computer Science."
Jean says: "I live in Montreal."
```

Τέλος, θα πρέπει να παρέχετε μια στατική μέθοδο `generateAgent` που επιστρέφει έναν "τυχαίο" πράκτορα. Αυτό θα πρέπει να το κάνει δημιουργώντας πρώτα τέσσερα διαφορετικά αντικείμενα `WordList` με βάση τα αρχεία για πόλεις, ειδικότητες, γυναικεία ονόματα και ανδρικά ονόματα. Αυτά τα αρχεία βρίσκονται στο Moodle ή μπορείτε και εσείς να τα δημιουργήσετε μόνοι σας. Εάν τα δημιουργήσετε εσείς, τότε τα αρχεία πρέπει να έχουν ακριβώς το ίδιο όνομα με τα αρχεία που σας δίνονται.

Στη συνέχεια, χρησιμοποιείτε αυτές τις τέσσερις λίστες λέξεων, `WordList`, για να δημιουργήσετε έναν τυχαίο πράκτορα:

1. Επιλέξτε το φύλο τυχαία
2. Επιλέξτε τυχαία γενέθλια (υπόδειξη: πρέπει να επιλέξετε 3 τυχαίους ακέραιους)
3. Επιλέξτε ένα όνομα τυχαία, κοιτάζοντας είτε τα θηλυκά είτε τα αρσενικά ονόματα (χρησιμοποιώντας τη Λίστα λέξεων) με βάση το φύλο του πράκτορα
4. Επιλέξτε ένα δήμαρχο (mayor), μια πόλη που γεννήθηκε (ο πράκτορας) και μια πόλη που βρίσκεται αυτή τη στιγμή (ο πράκτορας) τυχαία χρησιμοποιώντας την κατάλληλη Λίστα λέξεων.

Η κλάση `WordList`

Όπως έχετε ήδη παρατηρήσει, η κλάση `Agent` χρησιμοποιεί μια άλλη κλάση, την `WordList` την οποία πρέπει να την ορίσετε. Η κλάση αυτή έχει μόνο ένα χαρακτηριστικό, μια λίστα λέξεων (`ArrayList`) με `Strings`. Επιπλέον, πρέπει να έχει και δυο μεθόδους:

1. Έναν κατασκευαστή που λαμβάνει ως παράμετρο ένα όνομα αρχείου. Ο κατασκευαστής θα πρέπει να εισάγει τις λέξεις από το αρχείο στη λίστα `ArrayList`. Σημειώστε ότι κάθε γραμμή του αρχείου πρέπει να θεωρείται ως μία "λέξη".
2. Μια μέθοδο `getRandomWord` που δεν παίρνει τίποτα ως παράμετρο και επιστρέφει μια τυχαία συμβολοσειρά από τη λίστα των λέξεων. Σημειώστε ότι μπορείτε να χρησιμοποιήσετε την κλάση `Random` για να το κάνετε αυτό.

Η κλάση `DiscussionDirector`

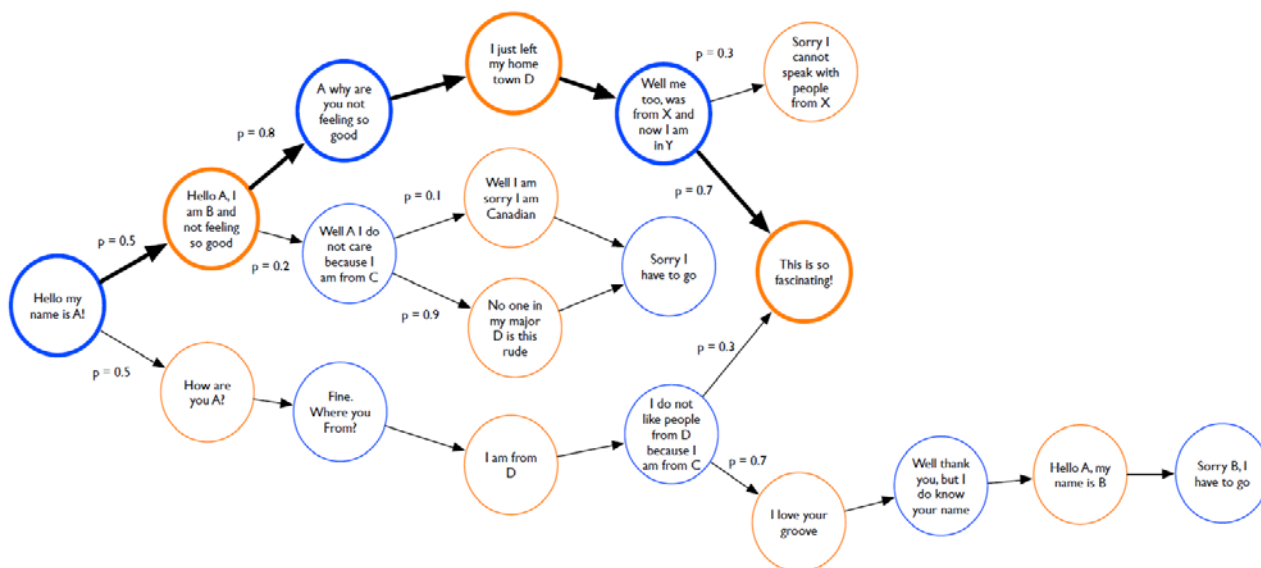
Αυτή η κλάση θα υλοποιεί μια συνομιλία μεταξύ δυο `Agent`. Για αυτό θα πρέπει να δημιουργήσετε δύο στιγμιότυπα της `Agent` και θα τα κάνετε να μιλήσουν μεταξύ τους. Για να το πετύχετε αυτό, ίσως χρειαστεί να δημιουργήσετε πρόσθετες μεθόδους στο `Agent.java`, όπως τη `sayHelloTo` (όνομα συμβολοσειράς) που θα έδιναν την ακόλουθη έξοδο:

Paul says: "Hi there, my name is Paul!"

Alice says: "Hello Paul, nice to meet you, my name is Alice, how are you?"

Για να ξεκινήσετε μια συνομιλία, θα πρέπει να δημιουργήσετε τη μέθοδο `discuss()` στην κλάση `DiscussionDirector`. Αυτή η μέθοδος θα λάβει δύο πράκτορες σαν παραμέτρους και θα εκτυπώσει μια συζήτηση μεταξύ τους.

Η συνομιλία που θα δημιουργήσετε θα πρέπει να ακολουθήσει τη δομή που παρουσιάζεται στον παρακάτω γράφο, ο οποίος αντιπροσωπεύει τις πιθανές συνομιλίες που μπορούν να έχουν οι δύο πράκτορες, ο μπλε κύκλος είναι ο πράκτορας Jean και ο πορτοκαλί κύκλος είναι η πράκτορας Lea.



Ξεκινήστε από το αριστερό τμήμα του γράφου με τον κύκλο να λέει "Γεια, με λένε A!" (χρησιμοποιώντας τις συγκεκριμένες μεθόδους του πράκτορα A.). Στη συνέχεια επιλέξτε τυχαία με πιθανότητα 50% εάν θα απαντήσετε

με "Γεια, Α, εγώ είμαι Β και δεν νιώθω τόσο καλά» ή «Πώς είσαι Α;». Στη συνέχεια, συνεχίστε να ακολουθείτε το διάγραμμα ροής μέχρι να φτάσετε στη δεξιά πλευρά του γραφήματος, λαμβάνοντας αποφάσεις με βάση την πιθανότητα σε κάθε διάσπαση του γράφου. Για τη μοντελοποίηση αυτού του γράφου, πρέπει να επιλέξετε μια λογική δομή δεδομένων!

Το πραγματικό κείμενο για κάθε γραμμή δεν είναι σημαντικό, αλλά πρέπει να έχετε την ίδια δομή με αυτό του γράφου. Προσπαθήστε να κάνετε το κείμενο να έχει νόημα και φυσικά όχι προσβλητικό.

Για παράδειγμα, μια κλήση της `discuss()` μπορεί να ακολουθεί την έντονη γραμμή στο γράφο και να δίνει την ακόλουθη έξοδο:

Jean says: "Bonjour, my name is Jean!"

Lea says: "Welcome Jean, nice to meet you, my name is Lea and I am not feeling so well."

Jean says: "Hoo Alice, why are you feeling so low?"

Lea says: "Well I just left my home town, Natasquan."

Jean says: "Me too, Natasquan is a fantastic town, I just left my home town Calgary as well."

Lea says: "This is fascinating, but really I have to go, see you Jean!"

Εάν θέλετε να είστε δημιουργικοί, εκτός από την εφαρμογή του παραπάνω γράφου, μπορείτε να επιλέξετε να υλοποιήσετε και άλλα μοντέλα συνομιλιών.

IV. Γενικές Οδηγίες

Είναι απαραίτητο να ακολουθήσετε την παρακάτω οργάνωση του προγράμματος:

- Project name: `Homeworks`
- Package name: `username.hw1` όπου `username` είναι αυτό που χρησιμοποιείται για να συνδεθείτε στις μηχανές του εργαστηρίου.
- Class names: Όπως σας ζητείται από την εκφώνηση
- Method signatures: Όπως σας ζητείται από την εκφώνηση.
- Javadoc documentation για όλες τις κλάσεις και μεθόδους.

Το πρόγραμμά σας θα πρέπει να περιλαμβάνει εύστοχα και περιεκτικά σχόλια, να έχει καλή στοίχιση και το όνομα κάθε μεταβλητής, σταθεράς, ή μεθόδου να είναι ενδεικτικό του ρόλου της. **Να χρησιμοποιήσετε το λογισμικό τεκμηρίωσης javadoc** έτσι ώστε να μπορούμε να μετατρέψουμε τα σχόλια του προγράμματός σας σε HTML αρχεία και να τα δούμε με ένα browser. Η συστηματική αντιμετώπιση της λύσης ενός προβλήματος περιλαμβάνει στο παρόν στάδιο τη διάσπαση του προβλήματος σε μικρότερα ανεξάρτητα προβλήματα που κατά κανόνα κωδικοποιούμε σε ξεχωριστές συναρτήσεις. Για αυτό τον λόγο σας καλούμε να κάνετε χρήση μεθόδων και άλλων τεχνικών αντικειμενοστραφούς προγραμματισμού. Επίσης, σας θυμίζουμε ότι κατά την διάρκεια της εκτέλεσης του προγράμματος σας, αυτό θα πρέπει να δίνει τα κατάλληλα μηνύματα σε περίπτωση λάθους. Τέλος το πρόγραμμά σας θα πρέπει να μεταγλωττίζεται στις μηχανές του εργαστηρίου.

Επίσης να ακολουθήσετε τα πιο κάτω βήματα όταν υποβάλετε την άσκηση σας στο Moodle:

1. **Ανεβάστε στο Moodle όλα τα αρχεία πηγαίου κώδικα java.**

V. Κριτήρια αξιολόγησης

Κλάση <code>Agent</code>	25
Κλάση <code>WordList</code>	20
Κλάση <code>DiscussionDirector</code> (συν τυχών άλλες κλάσεις)	40
Programming style	15
ΣΥΝΟΛΟ	100

Καλή επιτυχία.