

Εκφώνηση Απαλλακτικής Εργασίας

Ευφυείς Πράκτορες
Προπτυχιακό Πρόγραμμα Σπουδών
Τμήμα Πληροφορικής, Πανεπιστήμιο Πειραιά
Παναγιωτόπουλος Θεμιστοκλής
Απαλλακτική εργασία, ακαδημαϊκό έτος 2023

(Μια μόνο απο τις παρακάτω εργασίες πραγματοποιείται)

- 1 Εργασίες στην πόλη
- 2 Συγκέντρωση αγαθών
3. Ανάπτυξη generic planner
4. Χρήση στοιχείων γενετικών αλγορίθμων για την εξέλιξη πληθυσμού στο πρόβλημα «Συγκέντρωση Αγαθών».
5. ML-Agents (Machine Learning Agents) σύμφωνα με όσα είπαμε στο μάθημα
6. Affective Agents σύμφωνα με όσα είπαμε στο μάθημα

Η εργασία μπορεί να είναι έως 2 ατόμων. Ο κάθε φοιτητής πρέπει να ανεβάσει την εργασία σε μορφή word, open office ή pdf με όλα τα παραδοτέα στο gunet2 στον χώρο των εργασιών. Αν η εργασία είναι δύο ατόμων, πρέπει να φαίνονται τα στοιχεία του δεύτερου σε σχόλιο στο gunet2 και φυσικά μέσα στο κείμενο της εργασίας.

Η απαλλακτική ισχύει για όλες τις εξεταστικές περιόδους

1 Εργασίες στην πόλη

Περιγραφή

Μία ομάδα ευφυών εικονικών πρακτόρων κινούνται σε μία εικονική πόλη προσπαθώντας να βρεθούν στα κατάλληλα σημεία προκειμένου να εκτελέσουν συγκεκριμένα πλάνα ενεργειών.

Ανάλυση

Ο εικονικός κόσμος είναι επίπεδος και αποτελείται από $N \times M$ θέσεις (το λιγότερο 100×100). N, M ορίζονται ως παράμετροι στην αρχή της εφαρμογής.

Ο χώρος μέσα στον οποίο βρίσκονται οι πράκτορες έχει τη μορφή εξωτερικών χώρων πόλης. Συγκεκριμένες περιοχές του θεωρείται ότι ανήκουν σε συγκεκριμένα κτίρια.

Στην πόλη υπάρχουν δρόμοι, πάρκα, μονοπάτια καθώς και “σημαντικά” κτίρια (π.χ., τράπεζα, ταχυδρομείο, αγορά, κ.τ.λ.), σπίτια πρακτόρων και άλλα “ασήμαντα” κτίρια χωρίς ιδιαίτερη ταυτότητα αλλά με καθαρά “πολεοδομική” σημασία (π.χ. άλλες κατοικίες).

Στους δρόμους και τα μονοπάτια υπάρχουν σε τυχαία σημεία συνολικά G gold και E energy pots, G, E ορίζονται ως παράμετροι στην αρχή της εφαρμογής. Κάθε Energy Pot αξίζει X golds. Και αυτό ορίζεται στην αρχή της εφαρμογής.

Σε κάθε χρονική στιγμή, σε κάθε θέση μπορεί να υπάρχει (α) ένα τμήμα τοίχου συγκεκριμένου κτιρίου, ή (β) δρόμος, μέρος πάρκου, μονοπάτι, κ.λπ.

Οι πράκτορες έχουν τη δυνατότητα μετακίνησης μόνο γύρω από τα κτίρια και όχι στο εσωτερικό τους.

Το πλάνο του κάθε πράκτορα αποτελείται από ενέργειες μετακίνησης σε συγκεκριμένα κτίρια με συγκεκριμένη σειρά (π.χ. ταχυδρομείο, αγορά, τράπεζα).

Η ομάδα αποτελείται από έναν έως K πράκτορες, $4 \leq K \leq 10$. (Πρέπει να υπάρχουν και τουλάχιστον K σπίτια που μένουν οι πράκτορες). Το K ορίζεται παραμετρικά στην αρχή της εφαρμογής.

Οι πράκτορες έχουν δυνατότητες αντίληψης του περιβάλλοντος χώρου τους : πιο συγκεκριμένα, μπορούν να αντιληφθούν αν οι γειτονικές τους θέσεις περιέχουν κάποιον άλλο πράκτορα ή τμήμα τοίχου συγκεκριμένου κτιρίου, ή κάτι άλλο.

Όλοι οι πράκτορες μπορούν να επιλέξουν μια από τις παρακάτω ενέργειες που θα αρχίσουν να εκτελούν από το επόμενο clock tick :

1. exploration (τυχαία μετακίνηση σε κατευθύνσεις που θα του αποκαλύψουν όσο το δυνατόν περισσότερες θέσεις του άγνωστου χάρτη).

1. Μετακίνηση σε γειτονική θέση

2. Μετακίνηση σε θέση με Energy pot

3. Συλλογή Energy Pot από το έδαφος

4. Κατανάλωση Energy pot

5. Συλλογή Gold από το έδαφος

6. Ανταλλαγή γνώσης χάρτη με Golds πράκτορα σε γειτονική θέση (π.χ Εξαγορά χάρτη = 20 golds)

7. Αγορά / Πώληση Energy pots με πράκτορα σε γειτονική θέση (π.χ. 1 Energy pot = 10 gold)

Κάθε πράκτορας διαθέτει 100 Energy points. Κάθε clock tick ξοδεύει 1 Energy point. Αν το energy πέσει στο 0 ο πράκτορας πεθαίνει. Κάθε energy pot διαθέτει 20 Energy points.

Αρχικά οι πράκτορες δεν γνωρίζουν τον χάρτη της πόλης.

Εκτέλεση

Με την εκκίνηση γίνεται αρχικοποίηση της εφαρμογής. Προσδιορίζονται οι ελεύθερες παράμετροι :

- Ο εικονικός κόσμος είναι επίπεδος και αποτελείται από $N \times M$ θέσεις (το λιγότερο 100×100). N, M ορίζονται ως παράμετροι στην αρχή της εφαρμογής.
- Ενεργοποιείται ένας προσχεδιασμένος χάρτης με τις θέσεις των κτιρίων, δρόμων, μονοπατιών, κ.λπ.
- Στους δρόμους και τα μονοπάτια υπάρχουν σε τυχαία σημεία συνολικά G gold και E energy pots, G, E ορίζονται ως παράμετροι στην αρχή της εφαρμογής.
- Κάθε Energy Pot αξίζει X golds. Και αυτό ορίζεται στην αρχή της εφαρμογής.
- Η ομάδα αποτελείται από έναν έως K πράκτορες, $4 \leq K \leq 10$. (Πρέπει να υπάρχουν και τουλάχιστον K σπίτια που μένουν οι πράκτορες). Το K ορίζεται παραμετρικά στην αρχή της εφαρμογής.
- Εξαγορά χάρτη = Y golds. Και αυτό ορίζεται στην αρχή της εφαρμογής.

Κάθε πράκτορας ξεκινά από το σπίτι του και κινείται προσπαθώντας να εντοπίσει το πρώτο κτίριο που περιέχεται στο πλάνο του. Κατά την κίνησή τους, οι πράκτορες επιλέγουν κατευθύνσεις που θα του αποκαλύψουν όσο το δυνατόν περισσότερες θέσεις του άγνωστου χάρτη.

Όταν δύο πράκτορες συναντηθούν, ανταλλάσσουν όλη τη γνώση τους σχετικά με το μέρος του χάρτη που γνωρίζουν.

Όταν κάποιος πράκτορας αποκτήσει γνώση σχετικά με τη θέση του πρώτου κτιρίου στο πλάνο του σταματάει προσωρινά την εξερεύνηση και κινείται προσπαθώντας να φτάσει στο κτίριο. Στη συνέχεια συνεχίζει την εξερεύνηση έως ότου ανακαλύψει την θέση του επόμενου κτιρίου που πρέπει να επισκεφθεί, κ.ο.κ.

Όταν ένας πράκτορας μετακινηθεί σε θέση στην οποία βρίσκεται τμήμα τοίχου συγκεκριμένου κτιρίου θεωρείται πως έχει επισκεφθεί το κτίριο αυτό.

Όταν ένας πράκτορας επισκεφθεί το πρώτο κτίριο στο πλάνο του το διαγράφει από αυτό.

Όταν το πλάνο ενός πράκτορα είναι κενό, ο πράκτορας επιστρέφει στο σπίτι του αν έχει την απαραίτητη ενέργεια.

Ένας πράκτορας μπορεί προσωρινά να εγκαταλήψει το πλάνο του για να αναζητήσει energy pots τα οποία και θα καταναλώσει για να μην πεθάνει από πείνα (starvation).

Η εκτέλεση ολοκληρώνεται μόλις ο πρώτος πράκτορας επιστρέψει στο σπίτι του με ενέργεια μεγαλύτερη από το 0.

Προδιαγραφές

Ο εικονικός κόσμος θα περιγράφεται σε αρχείο κειμένου στον οποίο τα τμήματα τοίχων ασήμαντων κτιρίων θα σημειώνονται με "*", τα τμήματα τοίχων σημαντικών κτιρίων με ένα χαρακτήρα που θα προσδιορίζει μοναδικά το καθένα (π.χ., "T" για "Ταχυδρομείο", κ.τ.λ.), τα τμήματα τοίχων σπιτιών πρακτόρων με τα ψηφία "0", "1", ..., "9" και οι κενές θέσεις με κενό, δρόμοι, πάρκα, μονοπάτια έχουν επίσης κάποιο χαρακτηριστικό γράμμα) .

Τα πλάνα των πρακτόρων θα ορίζονται σε άλλο αρχείο κειμένου με κατάλληλη δομή (π.χ., "BEGINPLAN αριθμός πράκτορα n κτίριο 1... κτίριο m ENDPLAN")

Η διεπαφή της εφαρμογής θα είναι παραθυρική.

Η διεπαφή της εφαρμογής θα εμφανίζει μία κάτοψη του εικονικού κόσμου σε κάθε χρονική στιγμή, στην οποία θα εμφανίζεται η πόλη και οι πράκτορες.

Η διεπαφή της εφαρμογής θα παρέχει κατάλληλα μέσα για την επιλογή (και αποεπιλογή) ενός πράκτορα (π.χ., επιλογή πράκτορα μέσω πλήκτρων με ετικέτες "0", "1", ..., "9"), την οπτική απεικόνιση των μερών της πόλης για τα οποία ο επιλεγμένος πράκτορας έχει γνώση (π.χ., μέσω σκίασης περιοχών της κάτοψης ή τμηματικής απεικόνισής της), καθώς και την παρακολούθηση του πλάνου του.

Η διεπαφή της εφαρμογής θα παρέχει κατάλληλα μέσα για την φόρτωση αρχείου περιγραφής κόσμου και ορισμού πλάνων, καθώς και την έναρξη, την βηματική εκτέλεση, την παύση και τη ματαίωση της εκτέλεσης. Επιπλέον, η εφαρμογή θα επιτρέπει την επανέναρξη της εκτέλεσης μετά από ματαίωση χωρίς να είναι αναγκαία η εκ νέου φόρτωση αρχείου περιγραφής.

Η εφαρμογή θα καταγράφει σε αρχείο κειμένου διάφορα στατιστικά στοιχεία για την κάθε εκτέλεση, τα οποία θα περιέχουν τουλάχιστον το χρόνο εκτέλεσης, τον αριθμό των βημάτων του κάθε πράκτορα και τον αριθμό των ανταλλαγών γνώσης στις οποίες ο κάθε πράκτορας έλαβε μέρος.

Η εφαρμογή θα διαθέτει άμεσα εκτελέσιμο πρόγραμμα που δεν θα απαιτεί να κατέβουν διάφορα περιβάλλοντα για να εκτελεστεί.

Τεκμηρίωση

Η τεκμηρίωση της εφαρμογής θα περιλαμβάνει τα εξής:

1. Περιγραφή του προβλήματος
2. Περιγραφή της θεωρητικής βάσης της εφαρμογής, συμπεριλαμβανομένων των δομών δεδομένων και αναπαράστασης γνώσης που υιοθετήθηκαν, των αλγορίθμων και μεθοδολογιών που χρησιμοποιήθηκαν, καθώς και των προσαρμογών και μεταβολών που έγιναν στα παραπάνω προκειμένου να είναι δυνατή η εφαρμογή τους στο συγκεκριμένο πρόβλημα
3. Περιγραφή σημαντικών σχεδιαστικών αποφάσεων και στοιχείων υλοποίησης
4. Ολοκληρωμένη περιγραφή μίας παραδειγματικής εκτέλεσης και των αποτελεσμάτων της
5. Αναλυτική περιγραφή της διαδικασίας εγκατάστασης
6. Περιγραφή πρόσθετων δυνατοτήτων της εφαρμογής, εάν υπάρχουν

8. Αναλυτική περιγραφή ανοικτών θεμάτων, ανεπίλυτων προβλημάτων και πιθανοτήτων εμφάνισης σφαλμάτων κατά την εκτέλεση

Είναι σημαντικό να υπάρχουν αναλυτικά και επεξηγημένα screenshots από την εκτέλεση της εφαρμογής.

Παραδοτέα

1. Η εφαρμογή σε εκτελέσιμη μορφή (π.χ. αν είναι σε Unity πρέπει να έχει project και να έχει γίνει build)
2. Πηγαίος κώδικας για το σύνολο της εφαρμογής
3. Τεκμηρίωση όπως περιγράφεται προηγουμένως
4. Τουλάχιστον ένα παράδειγμα το οποίο θα συνοδεύεται από αντίστοιχο αρχείο στατιστικών
5. video + powerpoint

ΣΗΜΕΙΩΣΗ :

Η επιλογή γλωσσών και πλατφόρμας υλοποίησης είναι ελεύθερη.

2 Συγκέντρωση αγαθών

Περιγραφή

Δύο αντίπαλες ομάδες ευφυών εικονικών πρακτόρων με διαφορετικές ικανότητες προσπαθούν να συγκεντρώσουν συγκεκριμένα αγαθά από το κοινό περιβάλλον τους.

Ανάλυση

Ο εικονικός κόσμος είναι επίπεδος και αποτελείται από $N \times M$ θέσεις. (το λιγότερο 100×100). N, M ορίζονται ως παράμετροι στην αρχή της εφαρμογής.

Ο χώρος μέσα στον οποίο βρίσκονται οι πράκτορες έχει τη μορφή πεδιάδας στην οποία βρίσκονται διάφοροι πόροι (π.χ., ξυλεία, σίδηρος, χρυσός, σιτηρά, κ.τ.λ.) καθώς και τα χωριά στα οποία κατοικούν οι δύο ομάδες. Οι πράκτορες έχουν τη δυνατότητα μετακίνησης οπουδήποτε στην πεδιάδα.

Σε κάθε χρονική στιγμή, σε κάθε θέση μπορεί να υπάρχει (α) ένα μέρος συγκεκριμένου πόρου, (β) ένα μέρος συγκεκριμένου χωριού, ή (γ) τίποτα (κενός χώρος).

Κάθε ομάδα προσπαθεί να συγκεντρώσει στο χωριό της συγκεκριμένο αριθμό από συγκεκριμένα αγαθά (π.χ., 2 x χρυσό, 3 x σίδηρο, 10 x σιτηρά).

Κάθε πράκτορας μπορεί να μεταφέρει μία μονάδα από κάποιο αγαθό.

Ο κάθε πράκτορας της πρώτης ομάδας αναζητά αγαθά από κάποιο συγκεκριμένο πόρο.

Οι πράκτορες της δεύτερης ομάδας δεν έχουν προκαθορισμένους ρόλους. Έτσι, μεταφέρουν στο χωριό τους μονάδες αγαθών από οποιοδήποτε πόρο τύχει να συναντήσουν κατά την περιπλάνησή τους.

Η κάθε ομάδα αποτελείται από K πράκτορες. Το K ορίζεται ως παράμετρος στην αρχή της εφαρμογής.

Οι πράκτορες έχουν δυνατότητες αντίληψης του περιβάλλοντα χώρου τους: πιο συγκεκριμένα, μπορούν να αντιληφθούν αν οι γειτονικές τους θέσεις είναι κενές ή περιέχουν κάποιον άλλο πράκτορα ή τμήμα συγκεκριμένου πόρου.

Οι πράκτορες της πρώτης ομάδας μπορούν να αποφασίσουν για μια από τις παρακάτω ενέργειες που διαρκούν 1 clock tick η κάθε μία. Κάθε clock tick κοστίζει 1 Energy point σε κάθε πράκτορα.

1. Μετακίνηση σε γειτονική θέση
2. Εξαγορά γνώσης χάρτη από πράκτορα του ίδιου ή του άλλου χωριού σε γειτονική θέση. Ο άλλος πράκτορας πρέπει να συμφωνήσει στην αγοραπωλησία.
3. Συλλογή Energy pots
4. Εξαγορά Energy pot από άλλο πράκτορα σε γειτονική θέση. Ο άλλος πράκτορας πρέπει να συμφωνήσει στην αγοραπωλησία.
5. Συλλογή Gold
6. Συλλογή πόρων

Οι πράκτορες της πρώτης ομάδας μπορούν να αποφασίσουν για μια από τις παρακάτω ενέργειες που διαρκούν 1 clock tick η κάθε μία. Κάθε clock tick κοστίζει 1 Energy point σε κάθε πράκτορα.

1. Μετακίνηση σε θέση μέχρι 2 θέσεις μακριά από την παρούσα

2. Εξαγορά γνώσης χάρτη από πράκτορα του ίδιου ή του άλλου χωριού σε γειτονική θέση. Ο άλλος πράκτορας πρέπει να συμφωνήσει στην αγοραπωλησία.

3. Συλλογή Energy pots

4. Εξαγορά Energy pot από άλλο πράκτορα σε γειτονική θέση. Ο άλλος πράκτορας πρέπει να συμφωνήσει στην αγοραπωλησία.

5. Συλλογή Gold

6. Συλλογή πόρων

Αρχικά οι πράκτορες δεν γνωρίζουν τη δομή της πεδιάδας, δηλαδή ποιές θέσεις ανήκουν σε πόρους ή χωριά.

Νικήτρια θεωρείται όποια ομάδα συγκεντρώσει πρώτη τα απαιτούμενα αγαθά.

Εκτέλεση

Με την εκκίνηση γίνεται αρχικοποίηση της εφαρμογής. Προσδιορίζονται οι ελεύθερες παράμετροι :

- Ο εικονικός κόσμος είναι επίπεδος και αποτελείται από $N \times M$ θέσεις (το λιγότερο 100×100). N, M ορίζονται ως παράμετροι στην αρχή της εφαρμογής.
- Ενεργοποιείται ένας χάρτης και ορίζονται τυχαία οι θέσεις των πόρων.
- Στην πεδιάδα υπάρχουν σε τυχαία σημεία συνολικά G gold και E energy pots, G, E ορίζονται ως παράμετροι στην αρχή της εφαρμογής.
- Κάθε Energy Pot αξίζει X golds. Και αυτό ορίζεται στην αρχή της εφαρμογής.
- Κάθε χωριό αποτελείται από K πράκτορες, $4 \leq K \leq 10$. Το K ορίζεται παραμετρικά στην αρχή της εφαρμογής.
- Εξαγορά χάρτη = Y golds. Και αυτό ορίζεται στην αρχή της εφαρμογής.

Κάθε πράκτορας ξεκινά από το χωριό του και κινείται προσπαθώντας να εντοπίσει είτε συγκεκριμένο είτε οποιοδήποτε πόρο, ανάλογα με το σε ποιά ομάδα ανήκει. Κατά την κίνησή τους, οι πράκτορες επιλέγουν τυχαίες κατευθύνσεις αλλά δεν προσπαθούν να επισκεφθούν μέρη της πεδιάδας τα οποία έχουν ήδη επισκεφθεί.

Όταν δύο πράκτορες της πρώτης ομάδας συναντηθούν, ανταλλάσσουν όλη τη γνώση τους σχετικά με τα μέρη της πεδιάδας τα οποία έχουν επισκεφθεί ή εξαγοράζει την γνώση του χάρτη ο ένας από τον άλλο (άν έχει gold). Ένας πράκτορας μπορεί να αρνηθεί να ανταλλάξει τον χάρτη του (εξαρτάται πόσες άγνωστες θέσεις του εξασφαλίζει ο άλλος χάρτης)

Όταν κάποιος πράκτορας αποκτήσει γνώση σχετικά με τη θέση κάποιου πόρου ο οποίος τον ενδιαφέρει μπορεί να αποφασίσει να κινηθεί προσπαθώντας να φτάσει σε αυτόν.

Αν ένας πράκτορας που δε μεταφέρει τίποτα μετακινηθεί σε θέση η οποία ανήκει σε κάποιο πόρο που τον ενδιαφέρει φορτώνει τον πόρο αυτό (1 clock tick) και αρχίζει να μεταφέρει μία μονάδα από το αγαθό που παράγει ο συγκεκριμένος πόρος.

Ένας πράκτορας ο οποίος αρχίζει να μεταφέρει μία μονάδα από οποιαδήποτε αγαθό σταματάει να αναζητά πόρους και αρχίζει να κινείται προς το χωριό του.

Αν ένας πράκτορας που μεταφέρει μία μονάδα από κάποιο αγαθό μετακινηθεί σε θέση η οποία ανήκει στο χωριό του αφήνει την μονάδα του πόρου κάτω (1 clock tick), η μονάδα προστίθεται στα συσσωρευμένα αγαθά του χωριού ενώ ο πράκτορας σταματά να μεταφέρει οτιδήποτε και ξεκινά πάλι να αναζητά πόρους.

Αν ένας πράκτορας φτάσει να έχει 0 Energy points πεθαίνει. Έτσι ένας πράκτορας μπορεί να αποφασίσει να αναζητήσει Energy pot λόγω του ότι κινδυνεύει να πεθάνει από πείνα.

Η εκτέλεση ολοκληρώνεται μόλις κάποια ομάδα συγκεντρώσει τουλάχιστον τα απαιτούμενα αγαθά.

Προδιαγραφές

Ο εικονικός κόσμος θα περιγράφεται σε αρχείο κειμένου στον οποίο οι θέσεις που ανήκουν σε πόρους θα σημειώνονται με ένα χαρακτήρα που θα προσδιορίζει μοναδικά τον κάθε πόρο (π.χ., “W” για “Wood”, “G” για “Gold”, κ.τ.λ.), οι θέσεις που ανήκουν σε χωριά με τα ψηφία “1” ή “2” και οι κενές θέσεις με κενό.

Η διεπαφή της εφαρμογής θα είναι παραθυρική.

Η διεπαφή της εφαρμογής θα εμφανίζει μία κάτοψη του εικονικού κόσμου σε κάθε χρονική στιγμή, στην οποία θα εμφανίζονται τα χωριά, οι πόροι και οι πράκτορες.

Η διεπαφή της εφαρμογής θα παρέχει κατάλληλα μέσα για την επιλογή (και αποεπιλογή) ενός πράκτορα (π.χ. μέσω λίστας) και την οπτική απεικόνιση των μερών της πεδιάδας για τα οποία ο επιλεγμένος πράκτορας έχει γνώση (π.χ., μέσω σκίασης περιοχών της κάτοψης ή τμηματικής απεικόνισής της και εμφάνισης χαρακτήρων για πόρους τη θέση των οποίων γνωρίζει). Επίσης, η διεπαφή θα παρουσιάζει το τι μεταφέρει κάθε πράκτορας ο οποίος μεταφέρει κάτι καθώς και τα συσσωρευμένα αγαθά του κάθε χωριού.

Η διεπαφή της εφαρμογής θα παρέχει κατάλληλα μέσα για την φόρτωση αρχείου περιγραφής, τον καθορισμό των απαιτούμενων αγαθών ανά χωριό καθώς επίσης και την έναρξη, την βηματική εκτέλεση, την παύση και τη ματαίωση της εκτέλεσης. Επιπλέον, η εφαρμογή θα επιτρέπει την επανέναρξη της εκτέλεσης μετά από ματαίωση χωρίς να είναι αναγκαία η εκ νέου φόρτωση αρχείου περιγραφής.

Η εφαρμογή θα καταγράφει σε αρχείο κειμένου διάφορα στατιστικά στοιχεία για την κάθε εκτέλεση, τα οποία θα περιέχουν τουλάχιστον το χρόνο εκτέλεσης, τον αριθμό των βημάτων του κάθε πράκτορα, τον αριθμό των ανταλλαγών γνώσης στις οποίες ο κάθε πράκτορας έλαβε μέρος και τη νικήτρια ομάδα.

Η εφαρμογή θα διαθέτει άμεσα εκτελέσιμο πρόγραμμα που δεν θα απαιτεί να κατέβουν διάφορα περιβάλλοντα για να εκτελεστεί.

Τεκμηρίωση

Η τεκμηρίωση της εφαρμογής θα περιλαμβάνει τα εξής:

1. Περιγραφή του προβλήματος
2. Περιγραφή της θεωρητικής βάσης της εφαρμογής, συμπεριλαμβανομένων των δομών δεδομένων και αναπαράστασης γνώσης που υιοθετήθηκαν, των αλγορίθμων και μεθοδολογιών που χρησιμοποιήθηκαν, καθώς και των προσαρμογών και μεταβολών που έγιναν στα παραπάνω προκειμένου να είναι δυνατή η εφαρμογή τους στο συγκεκριμένο πρόβλημα.
3. Περιγραφή σημαντικών σχεδιαστικών αποφάσεων και στοιχείων υλοποίησης
4. Ολοκληρωμένη περιγραφή μίας παραδειγματικής εκτέλεσης και των αποτελεσμάτων της
5. Αναλυτική περιγραφή της διαδικασίας εγκατάστασης
6. Περιγραφή πρόσθετων δυνατοτήτων της εφαρμογής, εάν υπάρχουν
7. Αναλυτική περιγραφή της συμβολής του κάθε μέλους της ομάδας

8. Αναλυτική περιγραφή ανοικτών θεμάτων, ανεπίλυτων προβλημάτων και πιθανοτήτων εμφάνισης σφαλμάτων κατά την εκτέλεση

Είναι σημαντικό να υπάρχουν αναλυτικά και επεξηγημένα screenshots από την εκτέλεση της εφαρμογής

Παραδοτέα

1. Η εφαρμογή σε εκτελέσιμη μορφή (π.χ. αν είναι σε Unity πρέπει να έχει project και να έχει γίνει build)
2. Πηγαίος κώδικας για το σύνολο της εφαρμογής
3. Τεκμηρίωση όπως περιγράφεται προηγουμένως
4. Τουλάχιστον ένα παράδειγμα το οποίο θα συνοδεύεται από αντίστοιχο αρχείο στατιστικών για συγκεκριμένα απαιτούμενα αγαθά ανά χωριό.
5. video + powerpoint

ΣΗΜΕΙΩΣΗ :

Η επιλογή γλωσσών και πλατφόρμας υλοποίησης είναι ελεύθερη.

3 Ανάπτυξη generic planner

Περιγραφή

Ένας γεννήτορας σχεδίων (plan generation, planner) παράγει μια ακολουθία ενεργειών ώστε, όταν εκτελεστούν οι ενέργειες ο κόσμος του πράκτορα να βρεθεί από μια αρχική σε μια τελική κατάσταση. Ο planner θεωρείται generic αν δεν αλλάζουμε σε τίποτα τον κώδικα του για την λύση διαφορετικών προβλημάτων.

Κάθε πρόβλημα που προσπαθούμε να επιλύσουμε διαφοροποιείται από τα άλλα μόνο από την διαφορετική περιγραφή του αρχικού, του τελικού, αλλά και οποιουδήποτε ενδιάμεσου στιγμιότυπου του κόσμου. Δηλαδή για κάθε πρόβλημα έχουμε μεν τον ίδιο τρόπο αναπαράστασης των καταστάσεων του κόσμου, αλλά (πιθανώς) διαφορετική αναπαράσταση από την αναπράσταση σε άλλα προβλήματα.

Επίσης για κάθε πρόβλημα διαθέτουμε ένα διαφορετικό σενεργειών που μπορούμε να εκτελέσουμε.

1. Αναζητείστε κώδικα ενός generic planner στο διαδίκτυο, ή αναπτύξτε τον δικό σας κώδικα.
2. Τρέξτε τον planner για 2 διαφορετικά προβλήματα : το Blocks World και το Water jug

https://en.wikipedia.org/wiki/Blocks_world

https://en.wikipedia.org/wiki/Water_pouring_puzzle

Τεκμηρίωση

Η τεκμηρίωση της εφαρμογής θα περιλαμβάνει τα εξής:

1. Περιγραφή του προβλήματος
2. Περιγραφή της θεωρητικής βάσης της εφαρμογής, συμπεριλαμβανομένων των δομών δεδομένων και αναπαράστασης γνώσης που υιοθετήθηκαν, των αλγορίθμων και μεθοδολογιών που χρησιμοποιήθηκαν, καθώς και των προσαρμογών και μεταβολών που έγιναν στα παραπάνω προκειμένου να είναι δυνατή η εφαρμογή τους στο συγκεκριμένο πρόβλημα
3. Περιγραφή σημαντικών σχεδιαστικών αποφάσεων και στοιχείων υλοποίησης
4. Ολοκληρωμένη περιγραφή μίας παραδειγματικής εκτέλεσης και των αποτελεσμάτων της τόσο για το Blocks World όσο και για το Water jug.
5. Αναλυτική περιγραφή της διαδικασίας εγκατάστασης
6. Περιγραφή πρόσθετων δυνατοτήτων της εφαρμογής, εάν υπάρχουν
8. Αναλυτική περιγραφή ανοικτών θεμάτων, ανεπίλυτων προβλημάτων και πιθανοτήτων εμφάνισης σφαλμάτων κατά την εκτέλεση

Είναι σημαντικό να υπάρχουν αναλυτικά και επεξηγημένα screenshots από την εκτέλεση της εφαρμογής.

Η εφαρμογή θα διαθέτει άμεσα εκτελέσιμο πρόγραμμα που δεν θα απαιτεί να κατέβουν διάφορα περιβάλλοντα για να εκτελεστεί.

Παραδοτέα

1. Η εφαρμογή σε εκτελέσιμη μορφή (π.χ. αν είναι σε Unity πρέπει να έχει project και να έχει γίνει build)

2. Πηγαίος κώδικας για το σύνολο της εφαρμογής
3. Τεκμηρίωση του planner και των αναπαραστάσεων των καταστάσεων και των ενεργειών για τα 2 προβλήματα
4. Τα αποτελέσματα του planner για τα 2 προβλήματα
5. video + powerpoint

ΣΗΜΕΙΩΣΗ :

Η επιλογή γλωσσών και πλατφόρμας υλοποίησης είναι ελεύθερη.

4. Χρήση στοιχείων γενετικών αλγορίθμων για την εξέλιξη πληθυσμού στο πρόβλημα «Συγκέντρωση Αγαθών».

Περιγραφή

Δύο αντίπαλες ομάδες ευφυών εικονικών πρακτόρων με διαφορετικές ικανότητες προσπαθούν να συγκεντρώσουν συγκεκριμένα αγαθά από το κοινό περιβάλλον τους. Η ιδέα είναι να χρησιμοποιηθούν τα στοιχεία διασταύρωσης και μετάλλαξης από τους γενετικούς αλγορίθμους για να γίνει πιο ενδιαφέρον το simulation.

Ανάλυση

Ο εικονικός κόσμος είναι επίπεδος και αποτελείται από $N \times M$ θέσεις. (το λιγότερο 100×100). N, M ορίζονται ως παράμετροι στην αρχή της εφαρμογής.

Ο χώρος μέσα στον οποίο βρίσκονται οι πράκτορες έχει τη μορφή πεδιάδας στην οποία βρίσκονται διάφοροι πόροι (ξυλεία, πέτρα, χρυσός) καθώς και τα χωριά στα οποία κατοικούν οι δύο ομάδες. Οι πράκτορες έχουν τη δυνατότητα μετακίνησης οπουδήποτε στην πεδιάδα.

Σε κάθε χρονική στιγμή, σε κάθε θέση μπορεί να υπάρχει (α) ένα μέρος συγκεκριμένου πόρου, (β) ένα μέρος συγκεκριμένου χωριού, ή (γ) τίποτα (κενός χώρος).

Κάθε ομάδα προσπαθεί να συγκεντρώσει στο χωριό της συγκεκριμένο αριθμό από συγκεκριμένα αγαθά (Α ξυλεία, Β πέτρα, C χρυσό). Το $A+B+C$ πρέπει να είναι το ίδιο για τα 2 χωριά.

Κάθε πράκτορας διαθέτει τα εξής χαρακτηριστικά :

0 μετακίνηση 1 θέση

1 μετακίνηση 2 θέσεις

00 εξειδικεύεται στην μεταφορά ξυλείας

01 εξειδικεύεται στην μεταφορά πέτρας

10 εξειδικεύεται στην μεταφορά χρυσού

11 μεταφέρει όλους τους πόρους

00 μπορεί να μεταφέρει μόνο μια μονάδα πόρου κάθε στιγμή

01 μπορεί να μεταφέρει δυό μονάδες πόρου κάθε στιγμή

10 μπορεί να μεταφέρει τρεις μονάδες πόρου κάθε στιγμή

11 μπορεί να μεταφέρει τέσσερις μονάδες πόρου κάθε στιγμή

00 ξεκινά με 10 Gold

01 ξεκινά με 20 Gold

10 ξεκινά με 40 Gold

11 ξεκινά με 80 Gold

00 ξεκινά με 1 Energy pot

01 ξεκινά με 2 Energy pots

10 ξεκινά με 3 Energy pots

11 ξεκινά με 4 Energy Pots

00 Διαθέτει 50 Energy points

01 ξεκινά με 100 Energy points

10 ξεκινά με 200 Energy points

11 ξεκινά με 400 Energy Points

Έτσι το χρωμόσωμα κάθε πράκτορα αποτελείται από 11 θέσεις.

Κάθε πράκτορας φέρει τα χαρακτηριστικά που υποστηρίζει το χρωμόσωμά του.

Η κάθε ομάδα αποτελείται από K πράκτορες. Το K πρέπει να ορίζεται ως παράμετρος στην αρχή της εφαρμογής.

Οι πράκτορες έχουν δυνατότητες αντίληψης του περιβάλλοντα χώρου τους: πιο συγκεκριμένα, μπορούν να αντιληφθούν αν οι γειτονικές τους θέσεις είναι κενές ή περιέχουν κάποιον άλλο πράκτορα ή τμήμα συγκεκριμένου πόρου.

Οι πράκτορες μπορούν να αποφασίσουν για μια από τις παρακάτω ενέργειες που διαρκούν 1 clock tick η κάθε μία. Κάθε clock tick κοστίζει 1 Energy point σε κάθε πράκτορα.

1. Μετακίνηση σε γειτονική θέση

2. Εξαγορά γνώσης χάρτη από πράκτορα του ίδιου ή του άλλου χωριού σε γειτονική θέση. Ο άλλος πράκτορας πρέπει να συμφωνήσει στην αγοραπωλησία.

3. Εξαγορά Energy pot από άλλο πράκτορα σε γειτονική θέση. Ο άλλος πράκτορας πρέπει να συμφωνήσει στην αγοραπωλησία.

4. Συλλογή Energy pots

5. Συλλογή Gold

6. Συλλογή πόρων

7. Διασταύρωση με άλλο πράκτορα γειτονικής θέσης (μέγιστη πιθανότητα διασταύρωσης D% σε κάθε συνάντηση). Η διασταύρωση είναι υποχρεωτική αν πέσει ο τυχαίος αριθμός είναι μικρότερος από την πιθανότητα διασταύρωσης. Σ' αυτή την περίπτωση τα χρωμοσώματα κόβονται σε τυχαία θέση και το ένα παιδί πέρνει το πρώτο μέρος του πρώτου γονέα και το δεύτερο μέρος του χρωμοσώματος του δεύτερου γονέα. Αντίστροφα για το δεύτερο παιδί. Τα παιδιά επίσης κληρονομούν το καθένα τις γνώσεις και τα υπάρχοντα των γονέων τους. Οι γνώσεις του χάρτη, οι πόροι, τα golds, τα Energy pots ενώνονται και μοιράζονται τυχαία στα παιδιά. Πόροι και υπάρχοντα που δεν ταιριάζουν ή ξεπερνάνε τα όρια ενός πράκτορα ρίχνονται στο έδαφος, Οι γονείς πεθαίνουν και αντικαθιστώνται ως πράκτορες από τα παιδιά. Αν οι γονείς ανήκαν σε διαφορετικά χωριά το ένα παιδί εντάσσεται στο ένα χωριό και το άλλο παιδί στο άλλο χωριό (τυχαία).

8. Μετάλλαξη. Σε κάθε χρονική στιγμή υπάρχει μια μέγιστη πιθανότητα μετάλλαξης M%. Αυτή πρέπει να είναι αρκετά μικρή για να μην αλλάξει ραγδαία όλος ο πληθυσμός. Εφόσον διαθέτουμε K πράκτορες για κάθε χωριό με 11 θέσεις για κάθε χρωμόσωμα έχουμε συνολικά $2 \times 11 \times K$ πιθανές θέσεις που μπορούν να υποστούν μετάλλαξη. Δημιουργείται ένας πίνακας $2 \times 11 \times K$ και για κάθε θέση τρέχει ένας τυχαίος αριθμός, αν ο αριθμός είναι μικρότερος από την πιθανότητα μετάλλαξης M% το αντίστοιχο bit μετατρέπεται από 0 σε 1 ή από 1 σε 0.

Αρχικά οι πράκτορες δεν γνωρίζουν τη δομή της πεδιάδας, δηλαδή ποιές θέσεις ανήκουν σε πόρους ή χωριά.

Ένας πράκτορας με 0 Energy points πεθαίνει και δεν αντικαθίσταται.

Νικήτρια θεωρείται όποια ομάδα συγκεντρώσει πρώτη τα απαιτούμενα αγαθά.

Εκτέλεση

Με την εκκίνηση γίνεται αρχικοποίηση της εφαρμογής. Προσδιορίζονται οι ελεύθερες παράμετροι :

- Ο εικονικός κόσμος είναι επίπεδος και αποτελείται από $N \times M$ θέσεις (το λιγότερο 100×100). N, M ορίζονται ως παράμετροι στην αρχή της εφαρμογής.
- Ενεργοποιείται ένας χάρτης και ορίζονται τυχαία οι θέσεις των πόρων.
- Στην πεδιάδα υπάρχουν σε τυχαία σημεία συνολικά G gold και E energy pots, G, E ορίζονται ως παράμετροι στην αρχή της εφαρμογής.
- Κάθε Energy Pot αξίζει X golds. Και αυτό ορίζεται στην αρχή της εφαρμογής.
- Κάθε χωριό αποτελείται από K πράκτορες, $4 \leq K \leq 10$. Το K ορίζεται παραμετρικά στην αρχή της εφαρμογής.
- Εξαγορά χάρτη = Y golds. Και αυτό ορίζεται στην αρχή της εφαρμογής.
- Αρχικοποιείται τυχαία ο πληθυσμός των χρωμοσωμάτων των $2 \times K$ πρακτόρων των 2 χωριών.

Κάθε πράκτορας ξεκινά από το χωριό του και κινείται προσπαθώντας να εντοπίσει είτε συγκεκριμένο είτε οποιοδήποτε πόρο, ανάλογα με το σε ποιά ομάδα ανήκει. Κατά την κίνησή τους, οι πράκτορες επιλέγουν τυχαίες κατευθύνσεις αλλά δεν προσπαθούν να επισκεφθούν μέρη της πεδιάδας τα οποία έχουν ήδη επισκεφθεί.

Όταν δύο πράκτορες της πρώτης ομάδας συναντηθούν, εξαγοράζει την γνώση του χάρτη ο ένας από τον άλλο (άν έχει gold). Ένας πράκτορας μπορεί να αρνηθεί να ανταλλάξει τον χάρτη του (εξαρτάται πόσες άγνωστες θέσεις του εξασφαλίζει ο άλλος χάρτης)

Όταν κάποιος πράκτορας αποκτήσει γνώση σχετικά με τη θέση κάποιου πόρου ο οποίος τον ενδιαφέρει μπορεί να αποφασίσει να κινηθεί προσπαθώντας να φτάσει σε αυτόν.

Αν ένας πράκτορας που δε μεταφέρει τίποτα μετακινηθεί σε θέση η οποία ανήκει σε κάποιο πόρο που τον ενδιαφέρει φορτώνει τον πόρο αυτό (1 clock tick) και αρχίζει να μεταφέρει μία μονάδα από το αγαθό που παράγει ο συγκεκριμένος πόρος.

Ένας πράκτορας ο οποίος αρχίζει να μεταφέρει μία μονάδα από οποιαδήποτε αγαθό σταματάει να αναζητά πόρους και αρχίζει να κινείται προς το χωριό του.

Αν ένας πράκτορας που μεταφέρει μία μονάδα από κάποιο αγαθό μετακινηθεί σε θέση η οποία ανήκει στο χωριό του αφήνει την μονάδα του πόρου κάτω (1 clock tick), η μονάδα προστίθεται στα συσσωρευμένα αγαθά του χωριού ενώ ο πράκτορας σταματά να μεταφέρει οτιδήποτε και ξεκινά πάλι να αναζητά πόρους.

Αν ένας πράκτορας φτάσει να έχει 0 Energy points πεθαίνει και δεν αντικαθίσταται. Έτσι ένας πράκτορας μπορεί να αποφασίσει να αναζητήσει Energy pot λόγω του ότι κινδυνεύει να πεθάνει από πείνα.

Η εκτέλεση ολοκληρώνεται μόλις κάποια ομάδα συγκεντρώσει τουλάχιστον τα απαιτούμενα αγαθά.

Προδιαγραφές

Ο εικονικός κόσμος θα περιγράφεται σε αρχείο κειμένου στον οποίο οι θέσεις που ανήκουν σε πόρους θα σημειώνονται με ένα χαρακτήρα που θα προσδιορίζει μοναδικά τον κάθε πόρο (π.χ., “W” για “Wood”, “G” για “Gold”, κ.τ.λ.), οι θέσεις που ανήκουν σε χωριά με τα ψηφία “1” ή “2” και οι κενές θέσεις με κενό.

Η διεπαφή της εφαρμογής θα είναι παραθυρική.

Η διεπαφή της εφαρμογής θα εμφανίζει μία κάτοψη του εικονικού κόσμου σε κάθε χρονική στιγμή, στην οποία θα εμφανίζονται τα χωριά, οι πόροι και οι πράκτορες.

Η διεπαφή της εφαρμογής θα παρέχει κατάλληλα μέσα για την επιλογή (και αποεπιλογή) ενός πράκτορα (π.χ. μέσω λίστας) και την οπτική απεικόνιση των μερών της πεδιάδας για τα οποία ο επιλεγμένος πράκτορας έχει γνώση (π.χ., μέσω σκίασης περιοχών της κάτοψης ή τμηματικής απεικόνισής της και εμφάνισης χαρακτήρων για πόρους τη θέση των οποίων γνωρίζει). Επίσης, η διεπαφή θα παρουσιάζει το τι μεταφέρει κάθε πράκτορας ο οποίος μεταφέρει κάτι καθώς και τα συσσωρευμένα αγαθά του κάθε χωριού.

Η διεπαφή της εφαρμογής θα παρέχει κατάλληλα μέσα για την φόρτωση αρχείου περιγραφής, τον καθορισμό των απαιτούμενων αγαθών ανά χωριό καθώς επίσης και την έναρξη, την βηματική εκτέλεση, την παύση και τη ματαίωση της εκτέλεσης. Επιπλέον, η εφαρμογή θα επιτρέπει την επανέναρξη της εκτέλεσης μετά από ματαίωση χωρίς να είναι αναγκαία η εκ νέου φόρτωση αρχείου περιγραφής.

Η εφαρμογή θα καταγράφει σε αρχείο κειμένου διάφορα στατιστικά στοιχεία για την κάθε εκτέλεση, τα οποία θα περιέχουν τουλάχιστον το χρόνο εκτέλεσης, τον αριθμό των βημάτων του κάθε πράκτορα, τις ενέργειες που ο κάθε πράκτορας αποφάσισε και έκανε και τη νικήτρια ομάδα.

Θα ήταν ενδιαφέρον σαυτή την προσομοίωση να φαίνεται με βάση μια αξιολόγηση αποτελεσματικότητας ποια ομάδα κερδίζει σε κάθε χρονική στιγμή (π.χ. μια γραφική παράσταση που θα εξελίσσεται στον χρόνο με το σύνολο των πόρων που κάθε χωριό έχει μαζέψει και το σύνολο των πρακτόρων που παραμένουν ζωντανοί)

Η εφαρμογή θα διαθέτει άμεσα εκτελέσιμο πρόγραμμα που δεν θα απαιτεί να κατέβουν διάφορα περιβάλλοντα για να εκτελεστεί.

Τεκμηρίωση

Η τεκμηρίωση της εφαρμογής θα περιλαμβάνει τα εξής:

1. Περιγραφή του προβλήματος
2. Περιγραφή της θεωρητικής βάσης της εφαρμογής, συμπεριλαμβανομένων των δομών δεδομένων και αναπαράστασης γνώσης που υιοθετήθηκαν, των αλγορίθμων και μεθοδολογιών που χρησιμοποιήθηκαν, καθώς και των προσαρμογών και μεταβολών που έγιναν στα παραπάνω προκειμένου να είναι δυνατή η εφαρμογή τους στο συγκεκριμένο πρόβλημα.
3. Περιγραφή σημαντικών σχεδιαστικών αποφάσεων και στοιχείων υλοποίησης
4. Ολοκληρωμένη περιγραφή μίας παραδειγματικής εκτέλεσης και των αποτελεσμάτων της
5. Αναλυτική περιγραφή της διαδικασίας εγκατάστασης
6. Περιγραφή πρόσθετων δυνατοτήτων της εφαρμογής, εάν υπάρχουν
7. Αναλυτική περιγραφή της συμβολής του κάθε μέλους της ομάδας

8. Αναλυτική περιγραφή ανοικτών θεμάτων, ανεπίλυτων προβλημάτων και πιθανοτήτων εμφάνισης σφαλμάτων κατά την εκτέλεση

Είναι σημαντικό να υπάρχουν αναλυτικά και επεξηγημένα screenshots από την εκτέλεση της εφαρμογής

Παραδοτέα

1. Η εφαρμογή σε εκτελέσιμη μορφή (π.χ. αν είναι σε Unity πρέπει να έχει project και να έχει γίνει build)
2. Πηγαίος κώδικας για το σύνολο της εφαρμογής
3. Τεκμηρίωση όπως περιγράφεται προηγουμένως
4. Τουλάχιστον ένα παράδειγμα το οποίο θα συνοδεύεται από αντίστοιχο αρχείο στατιστικών για συγκεκριμένα απαιτούμενα αγαθά ανά χωριό.
5. video + powerpoint

ΣΗΜΕΙΩΣΗ :

Η επιλογή γλωσσών και πλατφόρμας υλοποίησης είναι ελεύθερη.


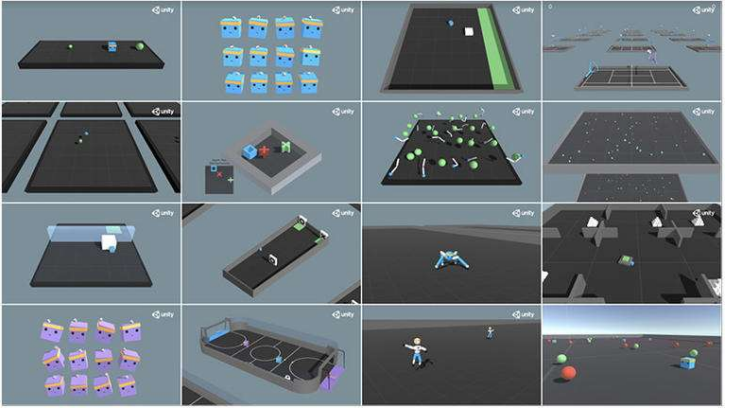
5. ML-Agents (Machine Learning Agents)

Περιγραφή

Οι πράκτορες Μηχανικής Μάθησης της Unity (Unity ML-Agents), αποτελούν μια ενδιαφέρουσα εναλλακτική εκπαίδευσης ευφυών Πρακτόρων μέσα στο περιβάλλον της Unity. Πραγματοποιώντας μια εργασία στα πλαίσια αυτά εξοικειώνεται κανείς με τις τεχνικές της μηχανικής μάθησης όπως αυτές υλοποιούνται στο πλαίσιο της πλατφόρμας της Unity

<https://unity.com/products/machine-learning-agents>

«Unity Machine Learning Agents, Train and embed intelligent agents by leveraging state-of-the-art deep learning technology»

	
Create an intelligent game experience	Create realistic and complex AI environments to train models

Υπάρχουν πολλά έτοιμα παραδείγματα πολύ ενδιαφέροντων εφαρμογών ML-Agents τόσο μέσα στην Unity, όσο και σε σχετικά videos στο YouTube :

<https://github.com/Unity-Technologies/ml-agents>

<https://www.youtube.com/watch?v=zPFU30tbyKs>

<https://towardsdatascience.com/an-introduction-to-unity-ml-agents-6238452fcf4c>

<https://gamedevacademy.org/unity-machine-learning-agents-tutorials/>

Υπάρχει επίσης μεγάλη ερευνητική δραστηριότητα που έχει δημοσιευθεί ως επιστημονικά άρθρα, ή μεταπτυχιακές διατριβές :

<https://arxiv.org/pdf/1809.02627.pdf>

<https://skemman.is/bitstream/1946/37111/1/An%20Evaluation%20of%20Unity%20ML-Agents%20Toolkit%20for%20Learning%20Boss%20-%20MSc%20Thesis.pdf>

Έχουν ήδη σε αρχίσει να εμφανίζονται και σχετικά ξενόγλωσσα βιβλία :

«Learn Unity ML-Agents – Fundamentals of Unity Machine Learning: Incorporate new powerful ML algorithms such as Deep Reinforcement Learning for games», Transform games into environments using machine learning and Deep learning with Tensorflow, Keras, and Unity

https://books.google.gr/books?hl=el&lr=&id=OMNiDwAAQBAJ&oi=fnd&pg=PP1&dq=Unity+Machine+Learning+Agents&ots=CjFPRA-zd-&sig=Os4n85rWeFL3HUgl5GXn-pQsAKQ&redir_esc=y#v=onepage&q=Unity%20Machine%20Learning%20Agents&f=false

Στην εργασία αυτή ζητείται να αξιοποιήσετε την σχετική τεχνολογία αναπτύσσοντας μια νέα εφαρμογή ή αξιοποιώντας μια υπάρχουσα και επεμβαίνοντας σε αυτή για να αναπτύξετε κάτι που ταιριάζει στα δικά σας ενδιαφέροντα.

Τεκμηρίωση

Η τεκμηρίωση της εφαρμογής θα περιλαμβάνει τα εξής:

1. Περιγραφή του προβλήματος
 2. Περιγραφή της θεωρητικής βάσης της εφαρμογής, συμπεριλαμβανομένων των δομών δεδομένων και αναπαράστασης γνώσης που υιοθετήθηκαν, των αλγορίθμων και μεθοδολογιών που χρησιμοποιήθηκαν, καθώς και των προσαρμογών και μεταβολών που έγιναν στα παραπάνω προκειμένου να είναι δυνατή η εφαρμογή τους στο συγκεκριμένο πρόβλημα.
 3. Περιγραφή σημαντικών σχεδιαστικών αποφάσεων και στοιχείων υλοποίησης
 4. Ολοκληρωμένη περιγραφή μίας παραδειγματικής εκτέλεσης και των αποτελεσμάτων της
 5. Αναλυτική περιγραφή της διαδικασίας εγκατάστασης
 6. Περιγραφή πρόσθετων δυνατοτήτων της εφαρμογής, εάν υπάρχουν
 7. Αναλυτική περιγραφή της συμβολής του κάθε μέλους της ομάδας
 8. Αναλυτική περιγραφή ανοικτών θεμάτων, ανεπίλυτων προβλημάτων και πιθανοτήτων εμφάνισης σφαλμάτων κατά την εκτέλεση
- Είναι σημαντικό να υπάρχουν αναλυτικά και επεξηγημένα screenshots από την εκτέλεση της εφαρμογής

Παραδοτέα

1. Η εφαρμογή σε εκτελέσιμη μορφή (πρέπει να έχει project και να έχει γίνει build)
2. Πηγαίος κώδικας για το σύνολο της εφαρμογής
3. Τεκμηρίωση όπως περιγράφεται προηγουμένως
4. Τουλάχιστον ένα παράδειγμα το οποίο θα συνοδεύεται από αντίστοιχο αρχείο στατιστικών για συγκεκριμένα απαιτούμενα αγαθά ανά χωριό.
5. video + powerpoint

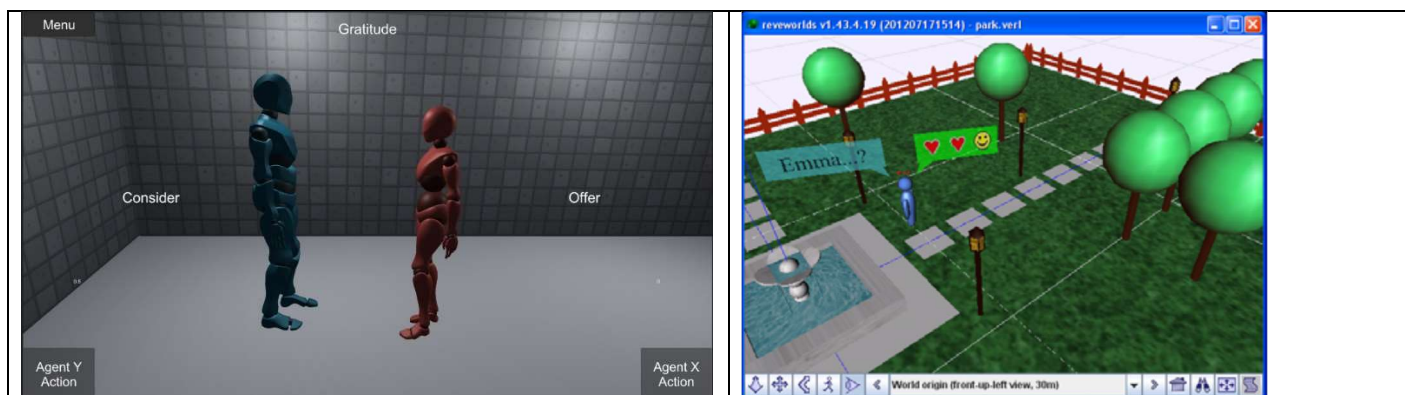
6. Affective Agents

Περιγραφή

Η εμφάνιση και ραγδαία ανάπτυξη του τομέα των τρισδιάστατων γραφικών (3D graphics) επιτρέπει την δημιουργία αληθοφανών χαρακτήρων οι οποίοι μπορούν να κινούνται και να ομιλούν. Με τη χρήση νέων τεχνολογιών, όπως η καταγραφή κίνησης (motion capture), η κίνηση στον τρισδιάστατο χώρο (3D animation) και η αναγνώριση φωνής (speech recognition), η κίνηση των τρισδιάστατων χαρακτήρων και η χρήση διαλόγων φυσικής γλώσσας εκ μέρους τους έχει καταστεί δυνατή.

Με την ανάπτυξη των παραπάνω τεχνολογιών, έχει αναδειχθεί και η σημασία της ανάπτυξης ενός υπολογιστικού μοντέλου που θα αναπαριστά όσο το δυνατόν πιστότερα την ανθρώπινη προσωπικότητα με σκοπό την ενσωμάτωση και τη χρήση του από τρισδιάστατους χαρακτήρες σε ψηφιακά παίγνια (digital games), σε προγράμματα Τεχνητής Νοημοσύνης (AI), καθώς και σε εφαρμογές Εικονικής Πραγματικότητας (Virtual Reality) και αναπαράστασης της πραγματικότητας (simulations).¹

Η ανθρώπινη προσωπικότητα, σαν μια ιδιαίτερα πολύπλοκη και όχι πλήρως αντικειμενικά αξιολογήσιμη δομή, δεν προσφέρεται για μια απόλυτα πιστή και έγκυρη ψηφιακή αναπαράσταση. Τόσο όσον αφορά τον ορισμό, όσο και τη δομή και την ερμηνεία, οι θεωρίες για την ανθρώπινη προσωπικότητα παρουσιάζουν μεγάλη ποικιλία. Η πλήρης ψηφιοποίηση ενός από τα υπάρχοντα μοντέλα προσωπικότητας αναμφίβολα ενέχει και πολλές ad-hoc παραδοχές κατά την ανάπτυξη του μοντέλου, οι οποίες θα πρέπει να καλύψουν ασάφειες αλλά και να επιλύσουν πρακτικά προβλήματα που προκύπτουν κατά τη μεταφορά του μοντέλου από τη θεωρία στην πρακτική εφαρμογή του στα πλαίσια (και με τους περιορισμούς) ενός ηλεκτρονικού προγράμματος.



Στην παρούσα εργασία είναι αναγκαίο να επιχειρηθεί η παρουσίαση και ανάπτυξη ενός απλού μοντέλου προσωπικότητας αληθοφανών πρακτόρων, το οποίο επιτρέπει την αλληλεπίδραση ανάμεσα σε πράκτορες που μπορούν να είναι ενσωματωμένοι σε τρισδιάστατους εικονικούς χαρακτήρες. Το μοντέλο αυτό περιλαμβάνει τις έννοιες της Προσωπικότητας, των Συναισθημάτων και των Πράξεων. Για να επιτευχθεί η όσο το δυνατόν ευρύτερη χρήση και γενίκευση της χρησιμότητας του μοντέλου, οι Πράξεις μπορούν να οριστούν από τον προγραμματιστή που το χρησιμοποιεί, ούτως ώστε να μπορούν να ορίζονται οι κατάλληλες πράξεις για κάθε περιβάλλον εφαρμογής όπου θα χρησιμοποιείται το μοντέλο.

- Για τις ανάγκες αυτές, παρέχεται στο σύνολό της η μεταπτυχιακή εργασία του φοιτητή Ιγνάτιου Καρέκου, με τίτλο «Αλληλεπίδραση μεταξύ εικονικών χαρακτήρων με τη χρήση αληθοφανών πρακτόρων»

- Η ερευνητική εργασία : «Emotion-driven behaviour for REVE intelligent virtual agents», T. Panayiotopoulos, E. Papaggeli, N. Avradinis, G. Anastassakis
- Και το εγχειρίδιο της εφαρμογής LoveHate που υπάρχει ως asset στο Unity. Το αντίστοιχο asset έχει αγοραστεί και είναι διαθέσιμο.

Τα ανωτέρω υπάρχουν στο χώρο του μαθήματος στο Teams στον φάκελλο «Emotional Agent».

Χρησιμοποιώντας τα ανωτέρω, το ζητούμενο είναι να αναπτυχθεί μια προσομοίωση με Ευφυείς Πράκτορες που διαθέτουν προσωπικότητα, συναισθήματα και εκτελούν ενέργειες παρόμοια με αυτή της ερευνητικής εργασίας που προσφέρεται, ή άλλης κοινωνικής αλληλεπίδρασης συναισθηματικών ευφυών πρακτόρων στα πλαίσια κάποιας μεταξύ τους κοινωνικής συνεύρεσης (π.χ. κάποιο πάρτι, παρέα, κλπ). Μπορείτε να χρησιμοποιήσετε το Terrain που αναπτύξατε σε προηγούμενα μαθήματα (Εικονική Πραγματικότητα, Τεχνολογίες Ανάπτυξης παιχνιδιών)

Τεκμηρίωση

Η τεκμηρίωση της εφαρμογής θα περιλαμβάνει τα εξής:

1. Περιγραφή του προβλήματος
2. Περιγραφή της θεωρητικής βάσης της εφαρμογής, συμπεριλαμβανομένων των δομών δεδομένων και αναπαράστασης γνώσης που υιοθετήθηκαν, των αλγορίθμων και μεθοδολογιών που χρησιμοποιήθηκαν, καθώς και των προσαρμογών και μεταβολών που έγιναν στα παραπάνω προκειμένου να είναι δυνατή η εφαρμογή τους στο συγκεκριμένο πρόβλημα.
3. Περιγραφή σημαντικών σχεδιαστικών αποφάσεων και στοιχείων υλοποίησης
4. Ολοκληρωμένη περιγραφή μίας παραδειγματικής εκτέλεσης και των αποτελεσμάτων της
5. Αναλυτική περιγραφή της διαδικασίας εγκατάστασης
6. Περιγραφή πρόσθετων δυνατοτήτων της εφαρμογής, εάν υπάρχουν
7. Αναλυτική περιγραφή της συμβολής του κάθε μέλους της ομάδας
8. Αναλυτική περιγραφή ανοικτών θεμάτων, ανεπίλυτων προβλημάτων και πιθανοτήτων εμφάνισης σφαλμάτων κατά την εκτέλεση

Είναι σημαντικό να υπάρχουν αναλυτικά και επεξηγημένα screenshots από την εκτέλεση της εφαρμογής

Παραδοτέα

1. Η εφαρμογή σε εκτελέσιμη μορφή (πρέπει να έχει project και να έχει γίνει build)
2. Πηγαίος κώδικας για το σύνολο της εφαρμογής
3. Τεκμηρίωση όπως περιγράφεται προηγουμένως
4. Τουλάχιστον ένα παράδειγμα το οποίο θα συνοδεύεται από αντίστοιχο αρχείο στατιστικών για συγκεκριμένα απαιτούμενα αγαθά ανά χωριό.
5. video + powerpoint