

## Οντοκεντρικός Προγραμματισμός II –C++

Ακαδημαϊκό Έτος 2013-2014

### Εργαστηριακή Άσκηση

Στην παρούσα εργασία καλείστε να υλοποιήσετε την προσομοίωση της διαχείρισης στόλου ρομποτικών οχημάτων για την εξερεύνηση ενός πλανήτη και την εξόρυξη χρήσιμων πόρων. Ο στόλος των ρομποτικών οχημάτων αποτελείται από διάφορα είδη οχημάτων τα οποία κινούνται μέσα σε έναν προκαθορισμένο χώρο, ο οποίος και θα αποτελεί τον χάρτη του κόσμου προσομοίωσης.

#### ΠΕΡΙΒΑΛΛΟΝ ΠΛΑΝΗΤΗ

Στόχος της εξομοίωσης είναι να συγκεντρωθεί στην βάση μια προκαθορισμένη ποσότητα των τριών συστατικών: Παλλάδιο, Ιρίδιο, Λευκόχρυσος. Κάθε σημείο στον χάρτη αναπαριστάται από τις παρακάτω τιμές: Περιεκτικότητα Παλλάδιου, Περιεκτικότητα Ιριδίου, Περιεκτικότητα Λευκόχρυσου, Επικινδυνότητα Πρόσβασης (Τιμή από 0 έως 0.9, όσο μεγαλύτερη τόσο πιθανότερο να προκληθεί βλάβη σε όχημα) και Σημαία Κινδύνου (Boolean μεταβλητή που σηματοδοτεί αν η θέση έχει χαρακτηριστεί ως επικίνδυνη και θα πρέπει να αποφεύγεται).

#### ΟΧΗΜΑΤΑ

Τα οχήματα διαχωρίζονται σε τρεις βασικές κατηγορίες: Εξερεύνησης, Ανάλυσης και Διάσωσης.

Οι βασικές μεταβλητές ενός οχήματος είναι η Ταχύτητα, η Κατάσταση (δείχνει αν έχει βλάβη ή όχι) και η Ικανότητα Πρόσβασης (Τιμή από 0.1 έως 1 που δείχνει πόσο πιθανό είναι να μην δεχτεί βλάβη κατά την μετακίνηση). Τα Ρομπότ Ανάλυσης έχουν επιπλέον τις Μεταβλητές: Μέγιστο Φορτίο, Φορτίο Παλλάδιου, Φορτίο Ιριδίου, Φορτίο Λευκόχρυσου.

Οι δυο βασικές λειτουργίες ενός οχήματος είναι η συνάρτηση Μετακίνησης και η συνάρτηση Λειτουργίας.

Τα οχήματα μετακινούνται με τυχαίο τρόπο κάθε φορά αλλά μπορούν να αποφύγουν μετακίνηση σε τετράγωνο με Σημαία Κινδύνου. Η πιθανότητα να προκληθεί βλάβη στο όχημα κατά την μετακίνηση του είναι: Επικινδυνότητα Πρόσβασης Προορισμού \* (1 - Ικανότητα Πρόσβασης Οχήματος).

Η συνάρτηση Λειτουργίας έχει διαφορετική υλοποίηση για κάθε είδος οχήματος (προτείνεται πολυμορφισμός):

**Ρομπότ Εξερεύνησης.** Ελέγχουν αν το τετράγωνο/έδαφος που βρίσκονται έχει υψηλή τιμή Επικινδυνότητας Πρόσβασης (πχ >0,6) και τοποθετούν Σημαία Κινδύνου ώστε να το αποφύγουν άλλα οχήματα.

**Ρομπότ Ανάλυσης.** Πραγματοποιούν την εξόρυξη των συστατικών από το έδαφος. Κάθε εξόρυξη έχει πιθανότητα να προκαλέσει βλάβη στο όχημα οπότε εξόρυξη γίνεται μόνο σε τετράγωνο με αυξημένη περιεκτικότητα ώστε να ελαχιστοποιηθούν οι εξορύξεις. Γενικά μπορείτε να επιλέξετε όποιον τρόπο θέλετε για την απόφαση αν θα κάνει εξόρυξη ή όχι (πχ μπορείτε να λαμβάνετε υπόψη και τις ανάγκες της βάσης σε κάθε συστατικό). Όταν το όχημα φτάσει το μέγιστο φορτίο, επιστρέφει στην βάση για την αποθήκευση του.

**Ρομπότ Διάσωσης.** Ελέγχουν τα γειτονικά τετράγωνα και αν εντοπίσουν σε αυτά όχημα με βλάβη το επιδιορθώνουν.

Το πρόγραμμα δημιουργεί με τυχαίο τρόπο μια δυσδιάστατη αναπαράσταση κόσμου προσομοίωσης με την τοποθέτηση οχημάτων όλων των ειδών σε διάφορες θέσεις του κόσμου. Επίσης αρχικοποιεί τις θέσεις του περιβάλλοντος όσο αφορά την περιεκτικότητα τους σε πόρους και την δυσκολία πρόσβασης. Η προσομοίωση του κόσμου εξελίσσεται με βάση κάποιους βασικούς κανόνες που αναφέρονται παρακάτω. Σε όλη τη διάρκεια της προσομοίωσης παρουσιάζονται στοιχεία του στόλου των ρομποτικών οχημάτων.

### **ΚΑΝΟΝΕΣ ΠΡΟΣΟΜΟΙΩΣΗΣ**

1. Σε κάθε επανάληψη της προσομοίωσης εμφανίζονται σύντομα μηνύματα που περιγράφουν τι συμβαίνει πχ (το όχημα Α μετακινήθηκε, το όχημα Β έκανε εξόρυξη κτλ.)
2. Για κάθε όχημα καταγράφεται το πόσες θέσεις έχει μετακινηθεί από την αρχή της εξομοίωσης και πόσες βλάβες έχει υποστεί. Για τα οχήματα εξερεύνησης καταγράφεται επιπλέον πόσες σημαίες έχει τοποθετήσει ενώ για τα οχήματα διάσωσης πόσες φορές έχουν επιδιορθώσει κάποιο όχημα. Τέλος για τα οχήματα ανάλυσης καταγράφεται η ποσότητα που έχουν συλλέξει συνολικά.
3. Η βάση βρίσκεται σε συγκεκριμένη θέση του χάρτη και ξεκινάει με μηδενικά αποθέματα των τριών πόρων. Τα οχήματα ανάλυσης συλλέγουν τα συστατικά και όταν φτάσουν το μέγιστο φορτίο, μεταφέρονται αυτόματα στην βάση και τα αποθηκεύουν.
4. Το σύστημα πρέπει να μπορεί να εμφανίσει όταν το ζητήσει ο χρήστης συγκεκριμένες πληροφορίες για κάποιο όχημα ή κάποια θέση εδάφους.
5. Το σύστημα πρέπει να διατηρεί και να εμφανίζει αν το ζητήσει ο χρήστης γενικές πληροφορίες για τα οχήματα όπως τον συνολικό αριθμό βλαβών, τον συνολικό αριθμό σημαιών κινδύνου, συνολικό φορτίο που εξορύχτηκε από όλα τα οχήματα και άλλα (προτείνεται η χρήση static μεταβλητών και μεθόδων). Παρόμοια πρέπει να εμφανίζει γενικές πληροφορίες όσο αφορά τις διαθέσιμες θέσεις εδάφους όπως μέσος όρος επικινδυνότητας εδάφους, συνολική ποσότητα κάθε στοιχείου στον χάρτη και άλλα).
6. Θεωρούμε ότι όλα τα οχήματα κινούνται συνεχώς μέσα στον κόσμο.
7. Αν η κατάσταση του κόσμου είναι τέτοια ώστε όλα τα οχήματα να έχουν βλάβη η προσομοίωση τερματίζει με μήνυμα αποτυχίας. Αν η ποσότητα των συστατικών στην βάση φτάσει το προκαθορισμένο όριο, η προσομοίωση τερματίζει με μήνυμα επιτυχίας.
8. Αν ένα όχημα υποστεί βλάβη και δεν επιδιορθωθεί μέσα σε συγκεκριμένο χρόνο (πχ 15 γύρους) τότε αποσύρεται από την εξομοίωση.
9. Κατά τη διάρκεια της προσομοίωσης ο χρήστης μπορεί να εισάγει εξωτερικούς παράγοντες στο σύστημα όπως εισαγωγή νέου οχήματος, επεξεργασία μιας θέσης εδάφους, εισαγωγή κατάστασης βλάβης σε όχημα ή επιδιόρθωση του.

### **Διεπαφή με τον χρήστη**

Η προσομοίωση θα υλοποιηθεί με διεπαφή κονσόλας. Ο χρήστης θα έχει την δυνατότητα να κάνει παύση της προσομοίωσης για να δει την συνολική κατάσταση του κόσμου ή/και πληροφορίες για κάποιο όχημα ή θέση του κόσμου, να εισάγει εξωτερικούς παράγοντες και να τερματίσει την προσομοίωση.

### **Γραφική Διεπαφή με χρήση της βιβλιοθήκης GLUT ( Προαιρετικό!!!)**

Η υλοποίηση γραφικής παραθυρικής διεπαφή για την οπτικοποίηση της εξέλιξης της προσομοίωσης είναι προαιρετική. Για την γραφική διεπαφή μπορεί να χρησιμοποιηθεί η βιβλιοθήκη GLUT (<http://www.lighthouse3d.com/opengl/glut/>). Για την οπτικοποίηση μπορείτε να κάνετε όποιες συμβάσεις επιθυμείτε: π.χ. να χρησιμοποιήσετε διαφορετικά χρώματα και σχήματα για κάθε οργανισμό ή να χρησιμοποιήσετε εικόνες bitmap.

## ΟΔΗΓΙΕΣ

### 1. Βαθμολόγηση:

- a. Για να υπάρξει προσμέτρηση της προγραμματιστικής εργασίας στον τελικό βαθμό, ο βαθμός της γραπτής εξέτασης θα πρέπει να είναι μεγαλύτερος ή ίσος του 5.
  - b. Η προγραμματιστική εργασία είναι υποχρεωτική και πρέπει να υποβληθεί για την προσέλευση στην γραπτή εξέταση της εξεταστικής του Ιανουαρίου. Αν η εργασία υποβληθεί την περίοδο του Σεπτεμβρίου συμμετέχετε μόνο στην γραπτή εξέταση του Σεπτεμβρίου.
  - c. Η προγραμματιστική εργασία συμμετέχει με βάρος 30% στην τελική βαθμολόγηση του μαθήματος. Σε περίπτωση που υλοποιηθεί το προαιρετικό ερώτημα της γραφικής διεπαφής συμμετέχει με βάρος 40%. Σε περίπτωση που η εργασία υποβληθεί τον Σεπτέμβριο η εργασία συμμετέχει με βάρος 15% και 20% αντίστοιχα.
  - d. Η βαθμολογία στην προγραμματιστική άσκηση δεν κατοχυρώνεται για επόμενη ακαδημαϊκή χρονιά.
2. Το πρόγραμμά σας θα πρέπει να υλοποιεί όλο το σύνολο των κλάσεων και διεπαφών για την υποστήριξη των παραπάνω λειτουργικών απαιτήσεων
  3. Το πρόγραμμα πρέπει να δημιουργηθεί χρησιμοποιώντας αντικειμενοστραφή σχεδιασμό (ιεραρχία κλάσεων, κληρονομικότητα, πολυμορφισμός κ.λ.π.).
  4. Η εργασία γίνεται σε ομάδες μέχρι των 3 ατόμων
  5. Απορίες υποβάλλονται στο φόρουμ του μαθήματος
  6. Η παράδοση της εργασίας γίνεται μέσω του **eclass.upatras.gr**
  7. Ημερομηνία Παράδοσης **10/01/2014**
  8. **Παραδοτέα:** Η άσκηση θα παραδίδεται υπό την μορφή ενός ZIP αρχείου, με όνομα <AM1>\_<AM2>\_<AM3>.zip (πχ. 1821\_1822\_1822.zip) όπου AMx οι αριθμοί μητρώου των μελών της ομάδας, και το οποίο θα περιέχει τα ακόλουθα:
    1. Πηγαίο κώδικα των κλάσεων που υλοποιούν την άσκηση (όχι τα εκτελέσιμα αρχεία!!)
    2. Σύντομη επεξήγηση του προγράμματος σας (.pdf).
    3. Διάγραμμα κλάσεων. Ένα σχεδιάγραμμα που δείχνει τη δομή και τις σχέσεις μεταξύ των κλάσεων του σχεδιασμού σας
    4. Ένα αρχείο txt με τις παραμέτρους εκτέλεσης του προγράμματος.
    5. Ένα αρχείο περιγραφής της μεταγλώττισης και σύνδεσης του προγράμματος (makefile ή DevC++ project file ή Visual Studio Solution File ή code::blocks project file)