

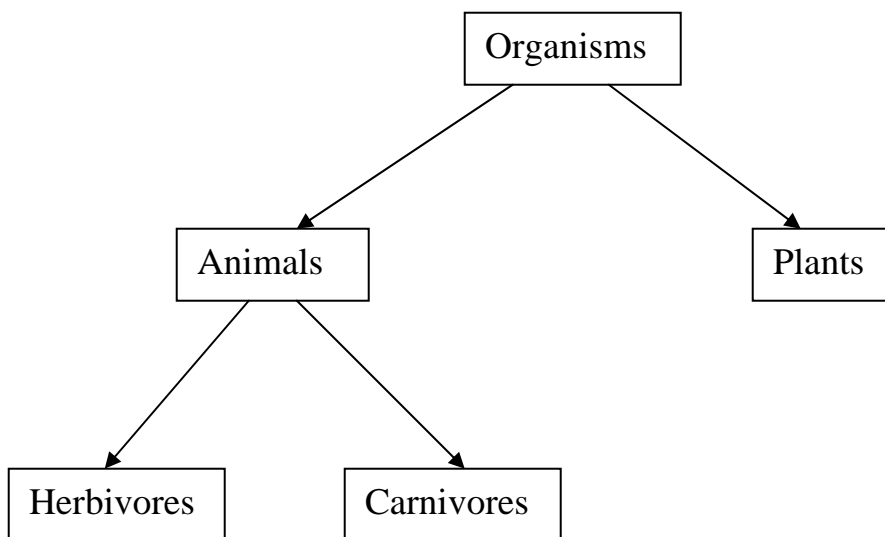
Οντοκεντρικός Προγραμματισμός II – C++ Project – Χειμερινό Εξάμηνο 2009-2010

Περιγραφή Project

Το project αφορά την προσομοίωση ενός οικοσυστήματος. Το οικοσύστημα μας αποτελείται από διάφορα είδη οργανισμών τα οποία ζουν ή κινούνται μέσα σε έναν προκαθορισμένο χώρο, ο οποίος και θα αποτελεί τον χάρτη του οικοσυστήματός μας. Κάθε οργανισμός καταναλώνει κάποιου είδους τροφή, η οποία το διατηρεί στην ζωή και η οποία μπορεί να διαφέρει ανάλογα με τον οργανισμό.

Ιεραρχία Οργανισμών

Οι οργανισμοί που ζουν στο οικοσύστημά μας κατατάσσονται σύμφωνα με την παρακάτω ιεραρχία:



Τα βέλη του σχήματος υποδηλώνουν σχέσεις isA μεταξύ των κλάσεων, οι οποίες αναπαράστώνται από τα ορθογώνια.

Πιο συγκεκριμένα για τις κλάσεις έχουμε:

- **Organisms:** Ένας οργανισμός έχει τρία κύρια χαρακτηριστικά, την ηλικία του (age), το μέγεθός του (size) και το βαθμό ανάπτυξής του (growth rate). Η ηλικία του οργανισμού μετρείται σε εβδομάδες και αρχικά είναι 0. Το μέγεθος του οργανισμού αρχικά είναι ένας αριθμός μεγαλύτερος του 0 και δίνεται κατά την δημιουργία του οργανισμού. Ο βαθμός ανάπτυξης είναι επίσης ένας αριθμός μεγαλύτερος του 0 και δείχνει το ποσό με το οποίο ο οργανισμός μεγαλώνει ανά εβδομάδα. Αν

κάποια στιγμή το μέγεθος του οργανισμού γίνει ίσος ή μικρότερος του 0 τότε ο οργανισμός πεθαίνει.

- **Animals:** Η κλάση αυτή περιέχει όλα τα ζώα του οικοσυστήματος. Ένα ζώο έχει ανάγκη να τραφεί ένα συγκεκριμένο ποσό κάθε εβδομάδα για να μπορέσει να μεγαλώσει σε μέγεθος (foodPerWeek). Κάθε εβδομάδα λοιπόν καταναλώνει κάποια ποσότητα τροφής καθώς περιπλανιέται στο οικοσύστημα (eatenThisWeek). Αν αυτή η ποσότητα που καταναλώνει είναι μεγαλύτερη από την ποσότητα που χρειάζεται για να επιβιώσει τότε το μέγεθος του ζώου μεγαλώνει κατά τον αντίστοιχο βαθμό ανάπτυξής του. Σε αντίθετη περίπτωση, που σημαίνει ότι το ζώο δεν κατάφερε να βρει όση τροφή χρειαζόταν για να επιβιώσει, το μέγεθός του μειώνεται κατά τον αντίστοιχο βαθμό ανάπτυξής του.
- **Plants:** Η κλάση αυτή περιέχει όλα τα φυτά του οικοσυστήματος. Το φυτό κάθε εβδομάδα αυξάνει το μέγεθός του πάντα κατά τον αντίστοιχο βαθμό ανάπτυξής του αφού το μόνο που χρειάζεται για να τραφεί είναι ήλιος και νερό τα οποία εμφανίζονται άφθονα στο οικοσύστημά μας. Το μέγεθος των φυτών όμως μπορεί να μειωθεί κατά τον αντίστοιχο βαθμό ανάπτυξής του όταν ένα ζώο τραφεί από αυτό.
- **Herbivores:** Η κλάση αυτή περιέχει τα ζώα που είναι φυτοφάγα και που μπορούν μόνο να τραφούν από φυτά που υπάρχουν διασκορπισμένα στο οικοσύστημα.
- **Carnivores:** Η κλάση αυτή περιέχει τα ζώα που είναι σαρκοφάγα και που μπορούν να τραφούν κάνοντας επίθεση σε άλλα ζώα (φυτοφάγα και σαρκοφάγα) των οποίων το μέγεθος είναι το μισό του μεγέθους τους, αφού δεν μπορούν να σκοτώσουν μεγαλύτερα ζώα. Το μέγεθος τροφής που λαμβάνουν από τα άλλα ζώα είναι ίσο με το βαθμό ανάπτυξης τους, δηλ. αν μία τίγρη επιτεθεί σε έναν λαγό, ο οποίος έχει βαθμό ανάπτυξης 10, τότε η τίγρη θα πάρει τροφή ίση με 10.

Γενικότερα ένα ζώο τρέφεται από ένα άλλο ζώο ή ένα άλλο φυτό, όταν αυτά πλησιάσουν σε μικρή απόσταση μεταξύ τους. Αυτή η απόσταση θα καθοριστεί από τις ομάδες και είναι συσχετισμένη με την υλοποίηση που θα κάνουν για τον χάρτη του οικοσυστήματος (δες παρακάτω).

Περιγραφή Εφαρμογής

Η εφαρμογή θα αποτελείται από μία σειρά λειτουργιών, έναν χάρτη και μία λίστα αντικειμένων. Πιο συγκεκριμένα:

- **Λειτουργίες:** Θα περιέχονται όλες οι γενικές λειτουργίες της προσομοίωσης, όπως αυτές περιγράφονται στο κεφάλαιο Λειτουργίες, και άλλες που πιθανόν να χρειαστεί να προστεθούν από τις ομάδες. Η υλοποίησή της θα μπορούσε να γίνει με την μορφή ενός menu επιλογών.
- **Λίστα Αντικειμένων:** Στην λίστα αυτή θα περιέχονται όλες οι ομάδες των οργανισμών του οικοσυστήματός μας. Από αυτή τη λίστα θα μπορεί να γίνεται η εισαγωγή ενός καινούργιου οργανισμού στο οικοσύστημά μας (δες κεφάλαιο Λειτουργίες).

- **Χάρτης:** Θα αποτελεί την περιοχή του οικοσυστήματός μας πάνω στον οποίο θα κινούνται οι οργανισμοί. Ο χάρτης του οικοσυστήματος θα είναι ένας δυσδιάστατος πίνακας. Κάθε οργανισμός θα αντιπροσωπεύεται στον χάρτη από τη θέση του σε κάποιο κενό κελί του πίνακα.

Λειτουργίες

Οι λειτουργίες που θα υποστηρίζονται από την εφαρμογή θα είναι:

- **Λειτουργίες Προσομοίωσης:** Οι λειτουργίες αυτές περιλαμβάνουν την έναρξη (Start), το πάγωμα (Pause), την συνέχιση (Resume) και την επανεκκίνηση (Restart) της προσομοίωσης. Θεωρούμε ότι όταν η προσομοίωση εκτελείται τα ζώα κινούνται με τυχαίο τρόπο. Κάθε ζώο εκτελεί N κινήσεις την ημέρα. Η κάθε κίνηση μπορεί να είναι σε ένα από τα 8 γειτονικά κελιά της τρέχουσας θέσης του. Μετά από κάθε κίνηση υπολογίζεται αν το ζώο τρέφεται με κάποιο φυτό ή άλλο ζώο ανάλογα με την κλάση του. Σε κάθε κελί μπορεί να υπάρχει μόνο ένας οργανισμός κάθε χρονική στιγμή.
- **Δημιουργία καινούργιου οργανισμού:** Η λειτουργία αυτή θα γίνεται επιλέγοντας έναν οργανισμό από τη λίστα αντικειμένων και επιλέγοντας την θέση στην οποία θα μπει ο οργανισμός στο χάρτη.
- **Πληροφορίες ανά πληθυσμό οργανισμών:** Πληροφορίες που αφορούν το μέσο όρο μεγέθους των οργανισμών της κατηγορίας που διαλέγουμε, το πλήθος των οργανισμών που ζουν αυτή την στιγμή, την μέση κατανάλωσή τους κατά την τελευταία εβδομάδα, ποσοστά θανάτων των οργανισμών σε σχέση με τους συνολικούς θανάτους κατά την διάρκεια της προσομοίωσης, στατιστικά των ηλικιών ανά οργανισμό. Οι πληροφορίες αυτές θα είναι διαθέσιμες από το μενού επιλογών.
- **Πληροφορίες ανά οργανισμού:** Επιλέγοντας έναν οργανισμό από τη λίστα αντικειμένων που υπάρχουν στον χάρτη θα εμφανίζονται τα στοιχεία του οργανισμού (π.χ. ηλικία, μέγεθος, βαθμός ανάπτυξης, foodPerWeek, eatenThisWeek) τα οποία και θα ανανεώνονται καθώς το ζώο κινείται στο χάρτη.

Για όλες τις λειτουργίες που υποστηρίζει η εφαρμογή, θα πρέπει να καταγράψετε λεπτομερώς στην αναφορά σας τα βήματα που θα πρέπει να ακολουθεί ο χρήστης για να πραγματοποιήσει τη λειτουργία αυτή (users manual).

Φάση 1η - Σχεδιασμός

Σε αυτή τη φάση πρέπει να γίνει ο σχεδιασμός της εφαρμογής βάσει των ιδεών και των αρχών που έχετε διδαχθεί. Αποτέλεσμα του σχεδιασμού θα είναι να καθοριστούν οι οντότητες, τα χαρακτηριστικά τους και η συμπεριφορά τους, οι οποίες θα αποτελέσουν δομικά στοιχεία για την υλοποίηση της ζητούμενης εφαρμογής.

Το αποτέλεσμα αυτής της φάσης θα είναι να καθοριστεί ακριβώς τι θα γίνει και να σκιαγραφηθεί σε επίπεδο classes και interfaces το πώς θα γίνει. Επιγραμματικά, οι σημαντικότερες εργασίες που πρέπει να γίνουν σε αυτή τη φάση είναι:

- Αναγνώριση των κλάσεων. Ποιά είναι τα χαρακτηριστικά του συστήματος και πως μοντελοποιούνται.
- Αναγνώριση του interface κάθε κλάσης.
- Εύρεση των χαρακτηριστικών και των ιδιοτήτων κάθε κλάσης.
- Τι δεδομένα πρέπει να κρατάει κάθε κλάση.
- Μεθόδους που υποστηρίζει κάθε αντικείμενο.
- Εύρεση των σχέσεων των αντικειμένων. Πώς επικοινωνούν (εάν επικοινωνούν) μεταξύ τους.
- Συμπεριφορά (behavior) των αντικειμένων.
- Αναγνώριση των ευθυνών κάθε κλάσης. Τι ενέργειες πρέπει να κάνει κάθε αντικείμενο.
- Τι ελέγχους πρέπει να κάνουν στα δεδομένα (pre/post conditions) τα αντικείμενα,
- μέσω των μεθόδων τους.
- Interface των κλάσεων, υπογραφές (signatures) των μεθόδων που υποστηρίζει, συναρτήσεις εγκατάστασης (constructors), συναρτήσεις επόπτες (observers), συναρτήσεις μετατροπής (modifiers).

Φάση 2η - Υλοποίηση

Σε αυτή τη φάση πρέπει να γίνει η υλοποίηση της εφαρμογής, βάσει της σχεδίασης που έχει προηγηθεί (φάση 1). Μολονότι δεν επιβάλλεται να χρησιμοποιηθεί αυτούσια η σχεδίαση της 1ης φάσης, καθότι κάποιες σχεδιαστικές επιλογές αποδεικνύονται στην πορεία άκυρες και χρειάζονται αναθεώρηση, εντούτοις η τελική βαθμολογία θα εξαρτηθεί και από το πόσο απείχε η τελική υλοποίηση από την αρχική σχεδίαση.

Σε αυτή τη φάση, παραδίδεται είναι :

- ο πηγαίος κώδικας που υλοποιεί την ζητούμενη εφαρμογή
- αναφορά, στην οποία θα αναλύεται :
 - ο η -τελική- σχεδίαση της εφαρμογής,
 - ο ποιές αλλαγές έγιναν σε σχέση με τη σχεδίαση της 1ης φάσης (και γιατί),
 - ο οι αλγόριθμοι που χρησιμοποιήθηκαν,
 - ο τα προβλήματα που αντιμετωπίστηκαν,
 - ο οι σχεδιαστικές ή προγραμματιστικές αποφάσεις που λήφθηκαν και πώς αυτό αντανακλάται στον τελικό χρήστη (πχ ευκολία/δυσκολία χειρισμού),
 - ο πιθανές επεκτάσεις (πχ καινούριες λειτουργίες) και πώς θα μπορούσαν να γίνουν με την παρούσα σχεδίαση,
 - ο ποιες αλλαγές θα έπρεπε να γίνουν στην υλοποίηση ή/και την σχεδίαση για να υποστηρίζονται επιπλέον λειτουργίες ή σχήματα.
 - ο ... γενικά ό,τι άλλο κρίνετε απαραίτητο να αναφέρετε.

Βαθμολογία εργασίας

Για την βαθμολογία της εργασίας θα συνεκτιμηθούν

- εάν (και πόσο) η σχεδίαση της εφαρμογής εφαρμόζει τις έννοιες και τεχνικές του οντοκεντρικού προγραμματισμού που διδάχθηκαν στο μάθημα
- εάν (και πόσο) υλοποιήθηκαν οι ζητούμενες λειτουργίες της εφαρμογής
- η πληρότητα της τελικής αναφοράς, η οποία θα καταγράφει και θα τεκμηριώνει την σχεδίαση και υλοποίηση της εφαρμογής

ΟΔΗΓΙΕΣ παράδοσης εργασίας

1) Η αναφορά της εργασίας θα είναι σε μία από τις μορφές:

- Acrobat pdf
- Postscript (ps)
- Microsoft Word 2000 compatible (doc)

Το **παραδοτέο αρχείο αναφοράς** θα πρέπει να έχει όνομα

cpp_YYYY_YYY1.xxx

Όπου: **xxx** το κατάλληλο file extension, **YYYY**, **YYY1** οι αριθμοί μητρώου των φοιτητών που συμμετέχουν στην κάθε ομάδα εργασίας

2) Στο κείμενο της αναφοράς **ΔΕΝ** θα πρέπει να περιλαμβάνεται εκτύπωση του κώδικα

3) Η αναφορά θα περιλαμβάνει στην πρώτη σελίδα το **Ονοματεπώνυμο, Αριθμό Μητρώου, Έτος** κάθε μέλους της ομάδας.

4) Ο κώδικας θα πρέπει να παρουσιάζεται όλος μαζί και να συνοδεύεται από τα απαραίτητα σχόλια.

5) Ένα **c++ project** μέσα σε **φάκελο** με όνομα **cppSource** που θα περιέχει μόνο **μία έκδοση του κώδικα** και το **αρχείο καταγραφής δεδομένων**. Θα περιλαμβάνεται επίσης **εκτελέσιμος κώδικας (compiled)** ο οποίος εκτελώντας τον δε θα παράγει λάθη.

6) Όλα τα παραπάνω αρχεία θα πρέπει να συμπεσθούν σε ένα αρχείο ZIP (.zip) με όνομα:

YYYY_YYY1_YYY2_YYY3.zip

Όπου: **YYYY**, **YYY1**, **YYY2**, **YYY3** οι αριθμοί μητρώου των φοιτητών που συμμετέχουν στην κάθε ομάδα εργασίας

7) Το **αρχείο ZIP ή RAR** θα πρέπει να **μετονομαστεί** σε .kip ή .kar και θα παραδοθεί **μόνο** ηλεκτρονικά με **email** και στις 2 διευθύνσεις skarlas@ceid.upatras.gr, foka@ceid.upatras.gr έως την **ημερομηνία της γραπτής εξέτασης του μαθήματος στην εξεταστική Ιανουαρίου, Φεβρουαρίου** με **θέμα (subject)** **"Cplusplus"** και στο **κείμενο (body)** του email θα αναφέρετε **Ονοματεπώνυμο,**

Αιριθμό Μητρώου, Έτος κάθε μέλους της ομάδας. (παρακαλούμε δώστε προσοχή στο θέμα, το κείμενο και την προσάρτηση συνημμένου ώστε να παραδοθεί επιτυχώς η εργασία σας)

- 8) Παράταση στην παράδοση της εργασίας δε θα δοθεί.
- 9) Οι φοιτητές παλαιότερων ετών θα πρέπει να παραδώσουν εκ νέου την παρούσα εργασία για να περάσουν το μάθημα.
- 10) Οι ομάδες εργασίας αποτελούνται από ένα (1) έως τέσσερα (2) άτομα
- 11) Η εργασία λαμβάνει το 30% του τελικού βαθμού μαθήματος και η γραπτή εξέταση το 70%.