

Προγραμματιστική Εργασία – Μέρος Β'

Δρακωνάκης Κώστας
Παπαϊωάννου Αντώνης



ΗΥ-240 Δομές Δεδομένων

Διαδικαστικά

- ❖ Παράδοση: Σάββατο, 19 Δεκεμβρίου 2016, ώρα 23:59
- ❖ **Compile & run σε μηχανήματα της σχολής**
 - ❖ Μέρος της βαθμολογίας
- ❖ Τρόπος παράδοσης: turnin
 - ❖ <http://csd.uoc.gr/~hy240b/current/submit.php>
- ❖ Online:
 - ❖ Main (java and C versions)
 - ❖ Test files
- ❖ <http://csd.uoc.gr/~hy240a>

Χρήσιμες Δομές

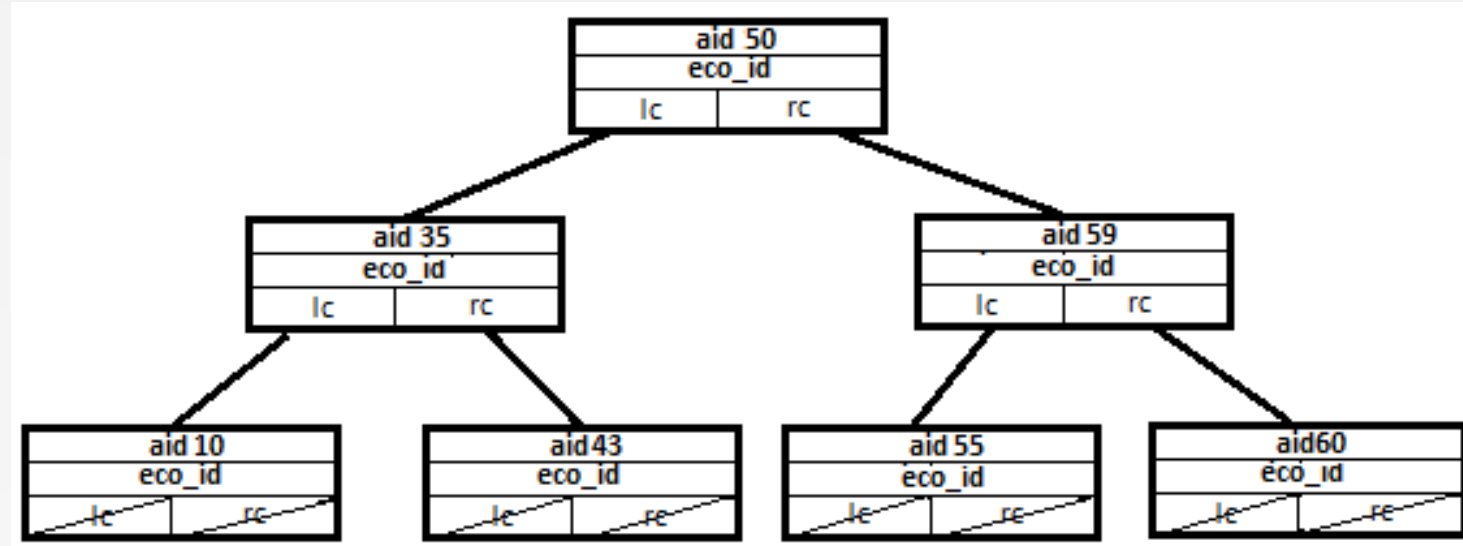
Animal

- **aid**: Αναγνωριστικό (int) που χαρακτηρίζει μοναδικά το ζώο
- **eco_id**: Αναγνωριστικό (int) που χαρακτηρίζει μοναδικά το οικοσύστημα όπου ανήκει το ζώο
- **lc**: Δείκτης (τύπου animal) στο αριστερό παιδί του κόμβου στο δένδρο ζώων
- **rc**: Δείκτης (τύπου animal) στο δεξί παιδί του κόμβου στο δένδρο ζώων

aid	
eco_id	
lc	rc

Animals Tree

- Δένδρο ζώων
- Δυαδικό δένδρο αναζήτησης
- Απλά συνδεδεμένο
- Ταξινομημένο σε αύξουσα διάταξη ως προς το **aid**



Αναγνωριστικά

- Κάθε οικοσύστημα μπορεί να φιλοξενήσει έως 20 ζώα
- Τα αναγνωριστικά (aid) των ζώων είναι συσχετισμένα με το οικοσύστημα στο οποίο ανήκουν
 - Ecosystem 0: $0 \leq \text{aid} \leq 19$
 - Ecosystem 1: $20 \leq \text{aid} \leq 39$
 - Ecosystem 2: $40 \leq \text{aid} \leq 59$ κ.ο.κ

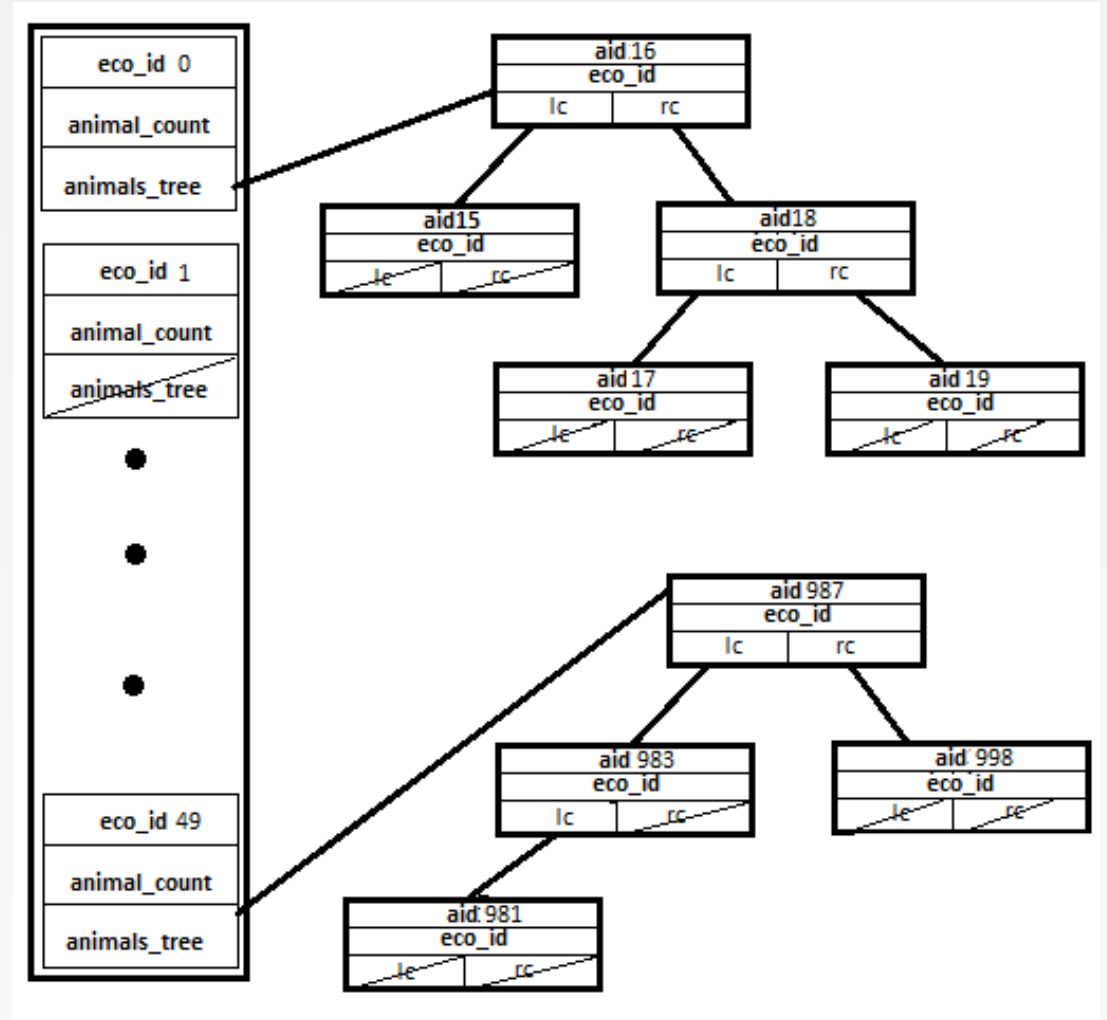
Ecosystem

- **eco_id**: Αναγνωριστικό (int) του οικοσυστήματος
- **animal_count**: Μετρητής (int) του πλήθους των ζώων που κατοικούν στο οικοσύστημα
- **animals_tree**: Δείκτης (τύπου animal) στην ρίζα ενός δυαδικού δένδρου αναζήτησης:
 - Απλά συνδεδεμένο
 - Ταξινομημένο ως προς το αναγνωριστικό των ζώων σε αύξουσα διάταξη

eco_id
animal_count
animals_tree

Ecosystems Array

- Πίνακας σταθερού μεγέθους 50 θέσεων
- Κάθε θέση του πίνακα είναι μία εγγραφή τύπου ecosystem



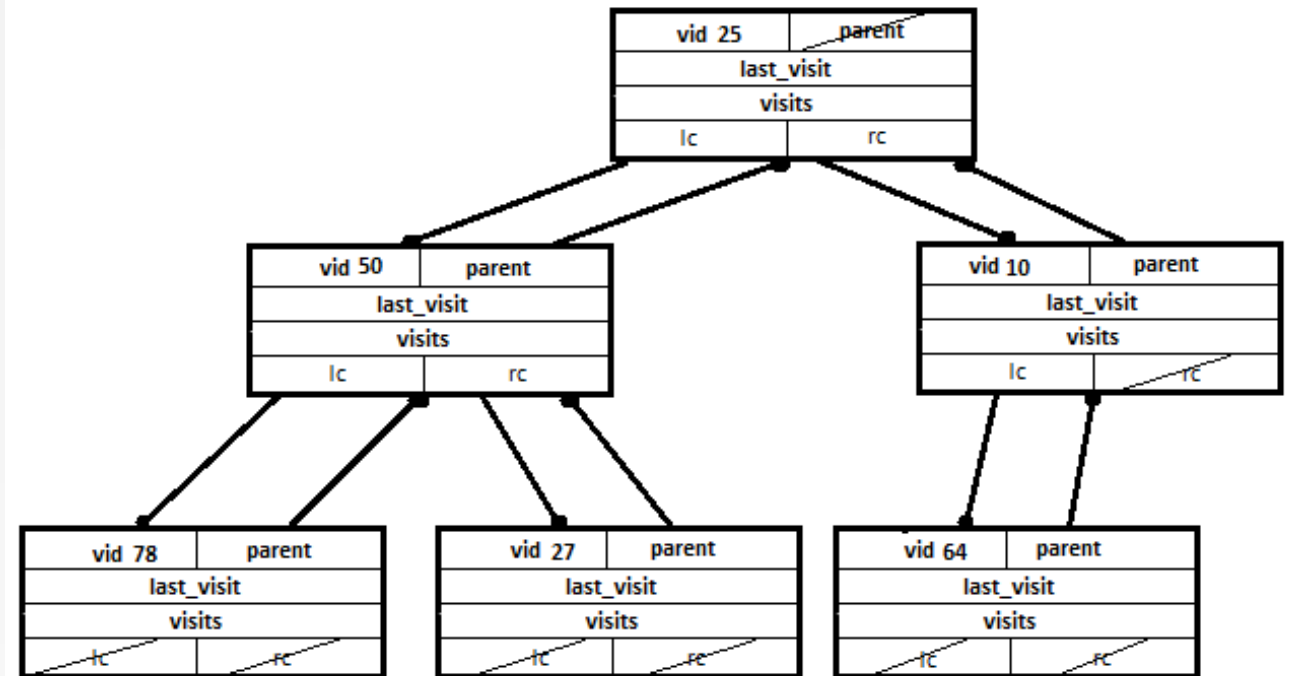
Visitor

- **vid**: Αναγνωριστικό (int) του επισκέπτη
- **last_visit**: Πεδίο (τύπου int) που αντιστοιχεί στο τελευταίο έτος επίσκεψης
- **visits**: Μετρητής (int) του αριθμού των επισκέψεων στο πάρκο
- **p**: Δείκτης (τύπου visitor) στον πατρικό κόμβο στο δένδρο επισκεπτών
- **lc**: Δείκτης (τύπου visitor) στο αριστερό παιδί του κόμβου στο δένδρο επισκεπτών
- **rc**: Δείκτης (τύπου visitor) στο δεξί παιδί του κόμβου στο δένδρο επισκεπτών

vid	p
last_visit	
visits	
lc	rc

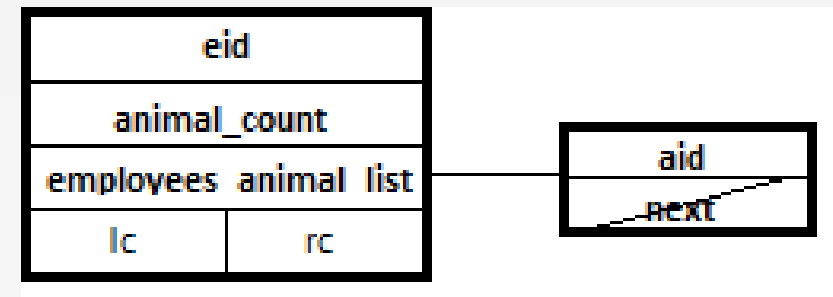
Visitors Tree

- Δένδρο επισκεπτών
- Πλήρες δυαδικό δένδρο
- Διπλά συνδεδεμένο
- Μη ταξινομημένο
- Η υλοποίηση θα πρέπει να γίνει με pointers κι όχι με πίνακα
- Δείκτης στην ρίζα αλλά και στο δεξιότερο φύλλο του δένδρου



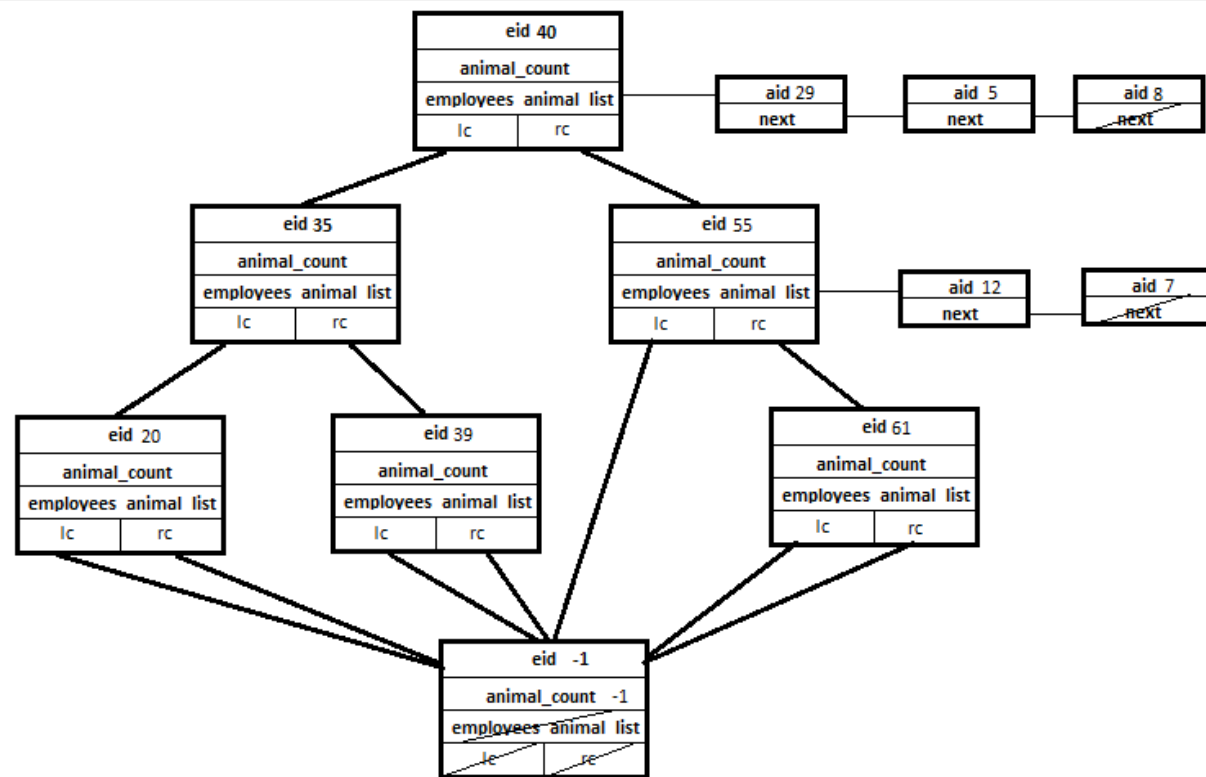
Employee

- **eid**: Αναγνωριστικό (int) του εργαζομένου
- **animal_count**: Μετρητής (int) για το πλήθος των ζώων για τα οποία είναι υπεύθυνος ο εργαζόμενος
- **lc**: Δείκτης (τύπου employee) στο αριστερό παιδί στο δένδρο εργαζομένων
- **rc**: Δείκτης (τύπου employee) στο δεξί παιδί στο δένδρο εργαζομένων
- **employees_animal_list**: Δείκτης (τύπου employee_animal) στο πρώτο στοιχείο της λίστας ζώων του εργαζομένου
 - Απλά συνδεδεμένη, μη ταξινομημένη
 - Κάθε κόμβος (employee_animal) περιέχει:
 - **aid**: Το αναγνωριστικό του ζώου
 - **next**: Δείκτης (τύπου employee_animal) στον επόμενο κόμβο της λίστας ζώων του εργαζομένου



Employees Tree

- Δένδρο εργαζομένων
- Δυαδικό δένδρο αναζήτησης
- Απλά συνδεδεμένο
- Ταξινομημένο ως προς το αναγνωριστικό των εργαζομένων σε αύξουσα διάταξη
- Διαχειριστικός κόμβος φρουρός τύπου employee με κενή λίστα ζώων κι αναγνωριστικό eid = -1



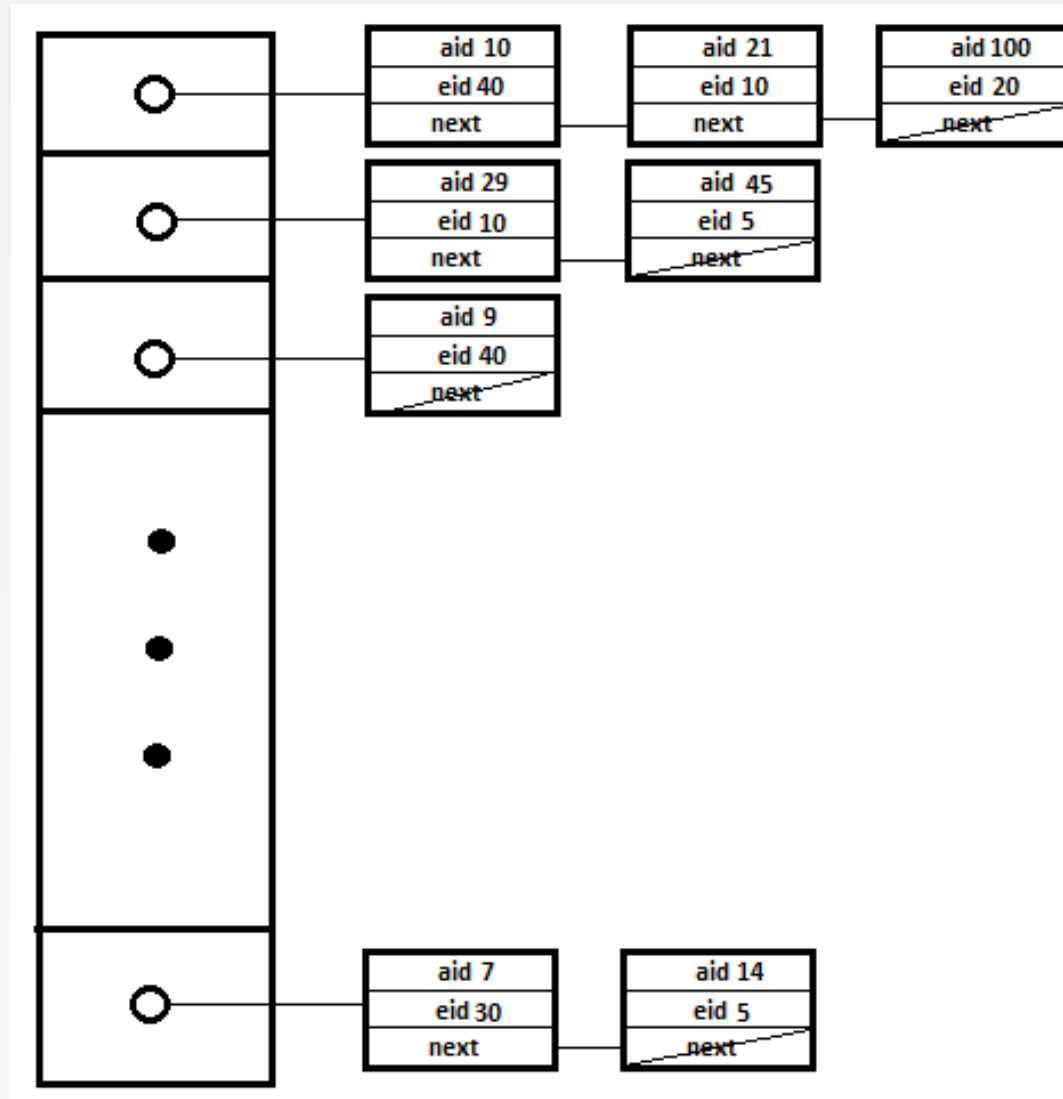
Πίνακας Κατακερματισμού

- Σκοπός: Αντίστροφη αντιστοίχιση για το ποιός εργαζόμενος είναι υπεύθυνος για κάποιο ζώο
- Μέθοδος των ταξινομημένων αλυσίδων
- Κάθε στοιχείο του πίνακα κατακερματισμού περιέχει έναν δείκτη (τύπου `animal_to_employee`) στο πρώτο στοιχείο μιας απλά συνδεδεμένης λίστας, η οποία θα είναι ταξινομημένη ως προς το αναγνωριστικό του ζώου
- Τα στοιχεία τύπου `animal_to_employee` περιέχουν:
 - **aid**: Το αναγνωριστικό του ζώου
 - **eid**: Το αναγνωριστικό του εργαζομένου
 - **next**: Δείκτης (τύπου `animal_to_employee`) στο επόμενο στοιχείο της αλυσίδας

Πίνακας Κατακερματισμού

- Ο κατακερματισμός θα γίνεται με βάση το αναγνωριστικό του ζώου **aid**
- Για την συνάρτηση κατακερματισμού θα πρέπει να χρησιμοποιηθεί η τεχνική του καθολικού κατακερματισμού

Πίνακας Κατακερματισμού



Γεγονότα

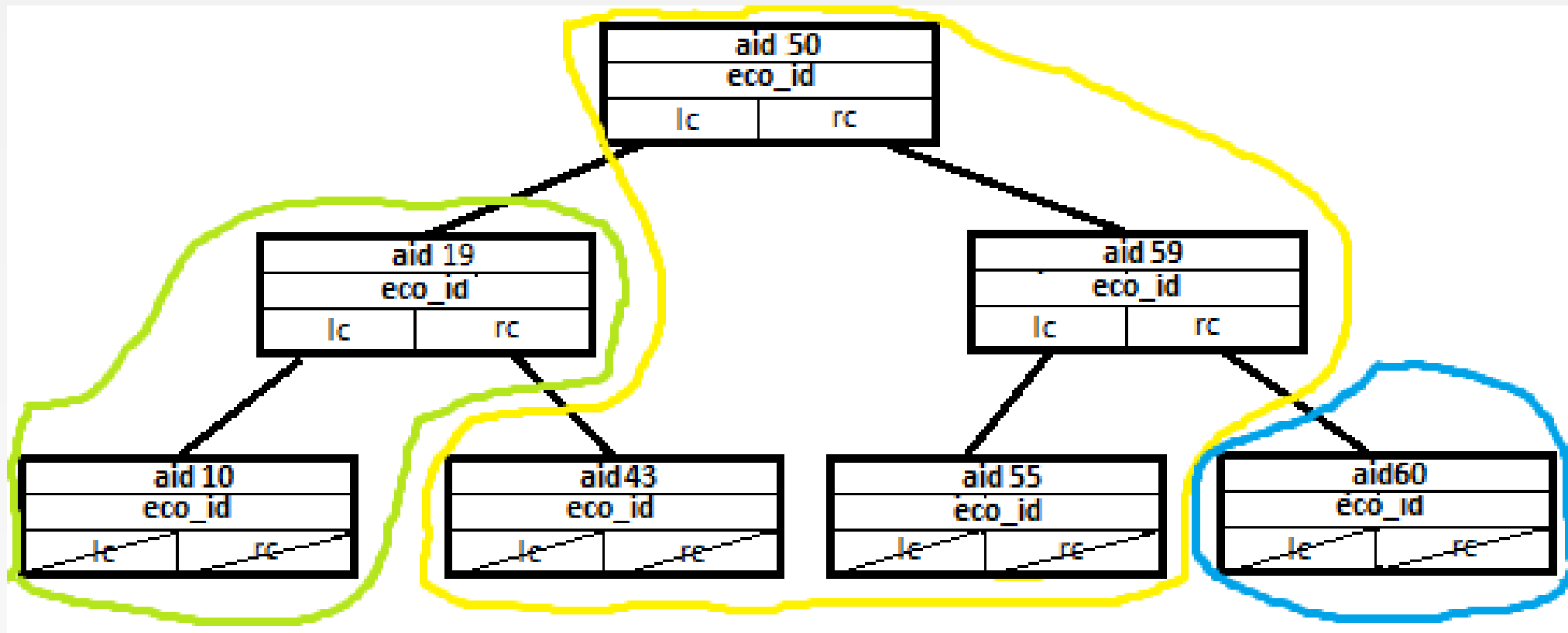
Γεγονότα

- L <aid> <eco_id>
- Εισαγωγή του ζώου με αναγνωριστικό aid στο δένδρο ζώων σε χρόνο $O(h)$
- <eco_id> το αναγνωριστικό του οικοσυστήματος στο οποίο κατοικεί το ζώο
- Μετά από κάθε εισαγωγή το δένδρο ζώων θα πρέπει να παραμένει ταξινομημένο

Γεγονότα

- D
- Διαχωρισμός των ζώων στα οικοσυστήματα που κατοικούν μέσα στο πάρκο
- Ο διαχωρισμός θα γίνεται ομαδικά **κι όχι** ατομικά για κάθε κόμβο
- Για κάθε οικοσύστημα ο διαχωρισμός θα πρέπει να γίνεται με βάση το μέγιστο αναγνωριστικό ζώου που του αντιστοιχεί
 - Για το οικοσύστημα 0 με βάση το aid 19, για το οικοσύστημα 1 με βάση το 39 κ.ο.κ
- Μετα από κάθε διαχωρισμό θα πρέπει να έχουν δημιουργηθεί δύο δένδρα
 - Το **T** που περιέχει τα στοιχεία που είναι μικρότερα ή ίσα από το ζητούμενο αναγνωριστικό
 - Το **T'** που περιέχει τα στοιχεία που είναι μεγαλύτερα από το ζητούμενο αναγνωριστικό
- Το **T** θα **μεταφέρεται** στην αντίστοιχη θέση του πίνακα οικοσυστημάτων
- Το **T'** θα χρησιμοποιείται κάθε φορά για τον διαχωρισμό των ζώων του επόμενου οικοσυστήματος έως ότου μείνει κενό
- Πολυπλοκότητα $O(h * \text{αριθμός οικοσυστημάτων})$
- Hint: 3^η σειρά ασκήσεων, άσκηση 2β

Γεγονότα



Γεγονότα

- V <vid> <year>
- Ο επισκέπτης με αναγνωριστικό <vid> επισκέπτεται το πάρκο το έτος <year>
- Αναζήτηση του <vid> στο δένδρο επισκεπτών
 - Αν βρεθεί τότε ανανεώνεται το πεδίο <last_visit> με την τιμή <year> και το πεδίο <visits> αυξάνεται κατά ένα
 - Αν δεν βρεθεί, δημιουργείται νέα εγγραφή τύπου visitor και εισάγεται κατάλληλα στο δένδρο επισκεπτών

Γεγονότα

- Ο <years_interval>
- Διαγραφή των επισκεπτών από το δένδρο επισκεπτών, οι οποίοι έχουν να επισκεφθούν το πάρκο περισσότερα από <years_interval> έτη

Γεγονότα

- Η $\langle \text{eid} \rangle$
- Ο εργαζόμενος με αναγνωριστικό $\langle \text{eid} \rangle$ προσλαμβάνεται στο πάρκο
- Δημιουργία του νέου κόμβου και εισαγωγή στο δένδρο εργαζομένων
- Η λίστα ζώων του εργαζόμενου είναι αρχικά κενή
- Εκτέλεση σε χρόνο $O(h)$

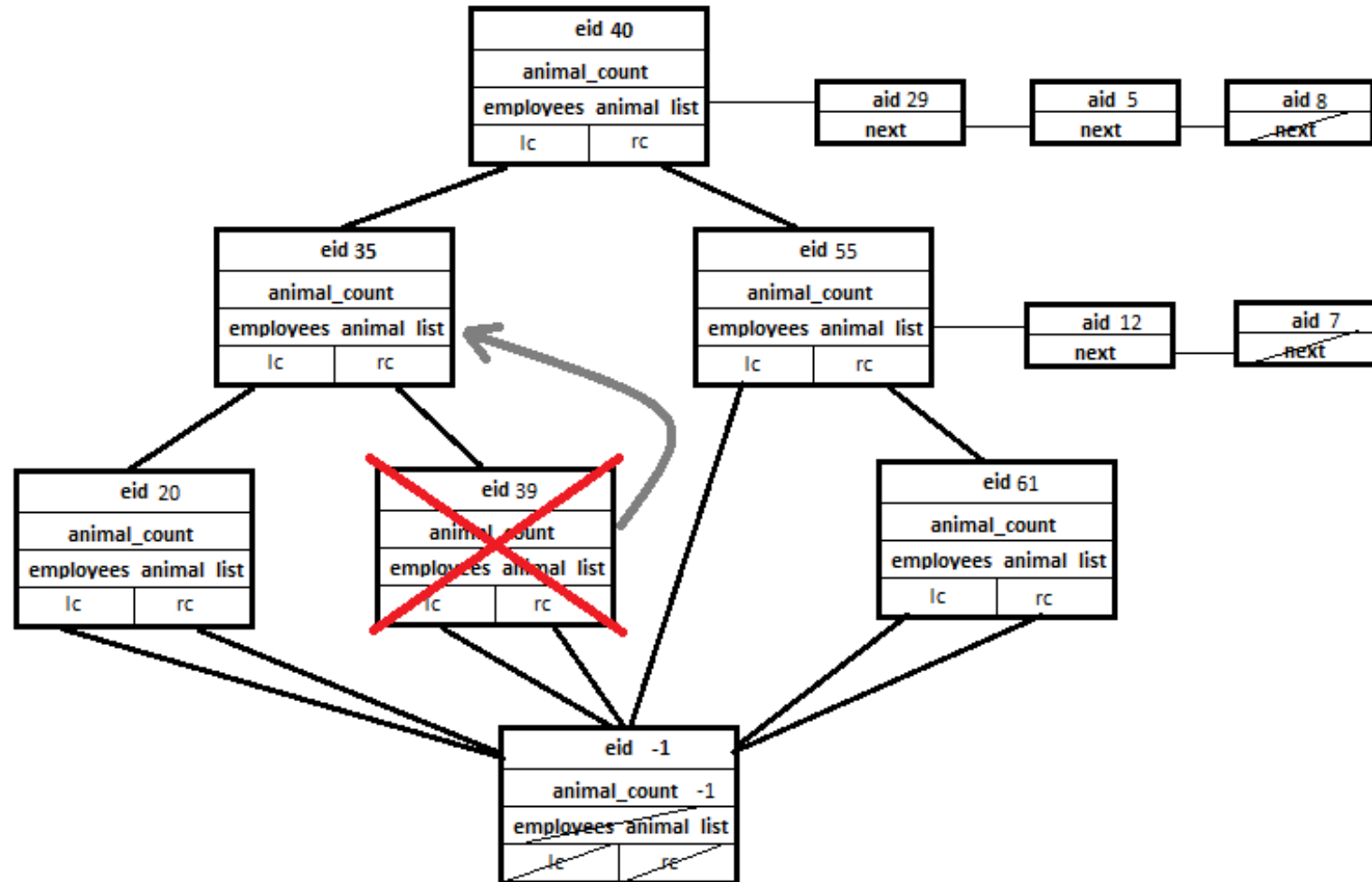
Γεγονότα

- Α <eid> <aid>
- Ανάθεση του ζώου με αναγνωριστικό <aid> στον εργαζόμενο με αναγνωριστικό <eid>
 - Αναζήτηση του εργαζόμενου <eid> στο δένδρο εργαζομένων
 - Εισαγωγή του ζώου <aid> στην λίστα ζώων του εργαζομένου που εντοπίστηκε
 - Εύρεση της κατάλληλης θέσης στον πίνακα κατακερματισμού με βάση το αναγνωριστικό του ζώου <aid> και εισαγωγή της νέας εγγραφής

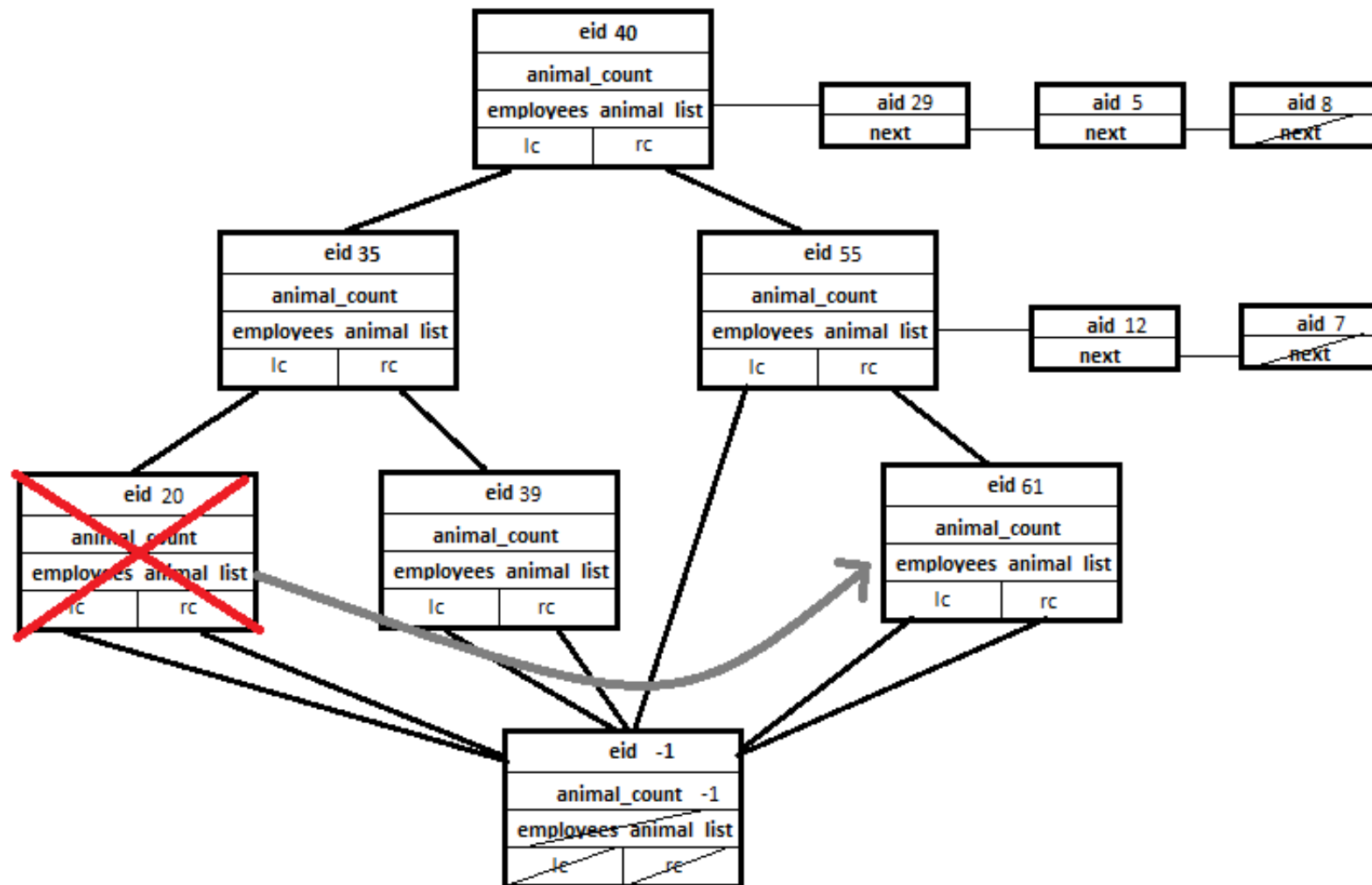
Γεγονότα

- R <eid>
- Διαγραφή του εργαζόμενου με αναγνωριστικό <eid> από το δένδρο εργαζομένων
- Τα ζώα για τα οποία είναι υπεύθυνος ο εργαζόμενος θα μεταφέρονται στον αμέσως προηγούμενο κόμβο του δένδρου όπως προκύπτει από την ενδοδιατεταγμένη (in-order) διάσχισή του
 - Σε περίπτωση που δεν υπάρχει προηγούμενος in-order κόμβος τα ζώα θα μεταφέρονται στον δεξιότερο κόμβο του δένδρου
- Εντοπισμός κι ανανέωση του πεδίου **eid** των κατάλληλων εγγραφών στον πίνακα κατακερματισμού

Γεγονότα



Γεγονότα



Γεγονότα

- X
 - Εκτύπωση όλων των οικοσυστημάτων του πίνακα οικοσυστημάτων
- Y
 - Εκτύπωση όλων των επισκεπτών του δένδρου επισκεπτών
- Z
 - Εκτύπωση όλων των εργαζομένων του δένδρου εργαζομένων
- E
 - Εκτύπωση όλων των αλυσίδων του πίνακα κατακερματισμού

Bonus

- Τα δένδρα των οικοσυστημάτων θα είναι τύπου AVL κι όχι απλά δυαδικά δένδρα αναζήτησης
- Επιπρόσθετο bonus στην βαθμολογία, αλλά μη υποχρεωτική υλοποίηση των AVL δένδρων
 - Η υλοποίηση που περιγράφηκε παραπάνω **ΕΙΝΑΙ ΥΠΟΧΡΕΩΤΙΚΗ!**
- Σε περίπτωση παράδοσης του Bonus θα πρέπει να παραδωθούν **δύο (2)** φάκελοι, ένας για κάθε εκδοχή των δένδρων των οικοσυστημάτων
 - Κάθε φάκελος θα πρέπει να περιέχει ότι χρειάζεται για να κάνει compile και να εκτελεστεί η εργασία

Διαδικαστικά

- ❖ Παράδοση: Σάββατο, 19 Δεκεμβρίου 2016, ώρα 23:59
- ❖ **Compile & run σε μηχανήματα της σχολής**
 - ❖ Μέρος της βαθμολογίας
- ❖ Τρόπος παράδοσης: turnin
 - ❖ <http://csd.uoc.gr/~hy240b/current/submit.php>
- ❖ Online:
 - ❖ Main (java and C versions)
 - ❖ Test files
- ❖ <http://csd.uoc.gr/~hy240a>

Questions?

