



Δομές Δεδομένων (Εργ.)
Ακ. Έτος 2018-19
Διδάσκων: Ευάγγελος Σπύρου

Εργαστήριο 4 – Στοιίβα (υλοποίηση με συνδεδεμένη λίστα)

1. Στόχος του εργαστηρίου

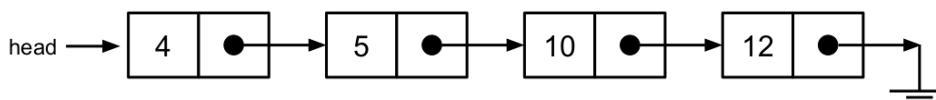
Στόχος του τέταρτου εργαστηρίου είναι η υλοποίηση της στοιίβας σε C με χρήση συνδεδεμένης λίστας. Θα υλοποιηθούν όσες συναρτήσεις χρειάζονται από αυτές που υλοποιήθηκαν στο εργαστήριο 1.

2. Συνδεδεμένες Λίστες

Κοινό χαρακτηριστικό των δομών που εξετάστηκαν έως τώρα είναι ότι διαδοχικοί κόμβοι καταλαμβάνουν διαδοχικές θέσεις μνήμης. Αυτό δημιουργεί όπως είδαμε ορισμένα προβλήματα:

- Δυσκολία στην εισαγωγή κόμβων
- Δυσκολία στη διαγραφή κόμβων (εξαιρούνται ο πρώτος και ο τελευταίος)
- Μη αποδοτική εκμετάλλευση διαθέσιμης μνήμης

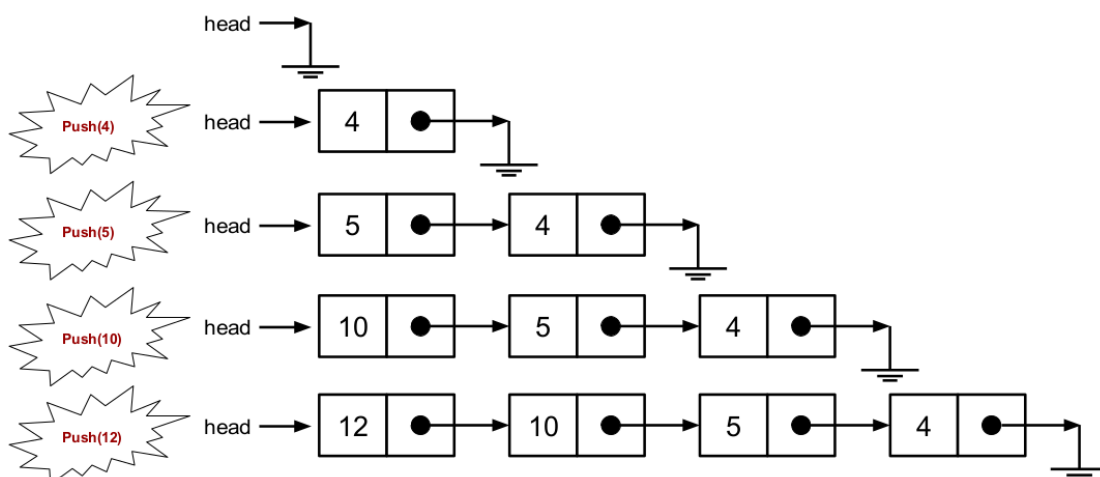
Οι συνδεδεμένες λίστες αποτελούν μια εναλλακτική, αποδοτική υλοποίηση των γραμμικών λιστών που είδαμε μέχρι τώρα. Κύριο χαρακτηριστικό των συνδεδεμένων λιστών είναι ότι οι κόμβοι τους βρίσκονται (συνήθως) σε απομακρυσμένες θέσεις μνήμης. Η σύνδεση μεταξύ των κόμβων γίνεται με δείκτες και διευκολύνονται έτσι η εισαγωγή και η διαγραφή δεδομένων. Η υλοποίηση των συνδεδεμένων λιστών γίνεται έτσι ώστε να μην απαιτείται εκ των προτέρων καθορισμός του μέγιστου αριθμού κόμβων, άρα μιλάμε για **δυναμικές** λίστες. Σε κάθε κόμβο της λίστας αποθηκεύονται κάποια **πληροφορία** (π.χ., στην περίπτωση μας ένας ακέραιος) και ένας **δείκτης** στο επόμενο αντικείμενο της συλλογής. Ο δείκτης head δείχνει το πρώτο στοιχείο (κεφαλή), ενώ το τελευταίο στοιχείο δεν δείχνει κάπου (θα λέμε ότι δείχνει σε NULL).



Οι βασικές λειτουργίες συνδεδεμένων γραμμικών λιστών είναι η Εισαγωγή και η Διαγραφή.

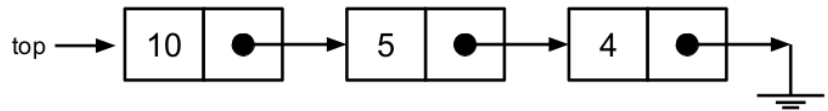
3. Στοιίβα ως συνδεδεμένη λίστα

Ας δούμε πώς λειτουργεί η push:

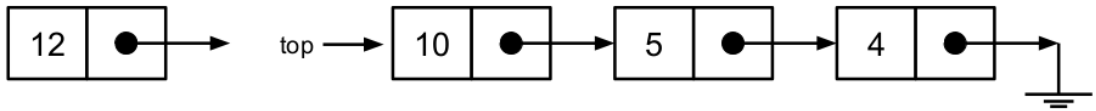


Θα μπορούσε να υλοποιηθεί ως εξής:

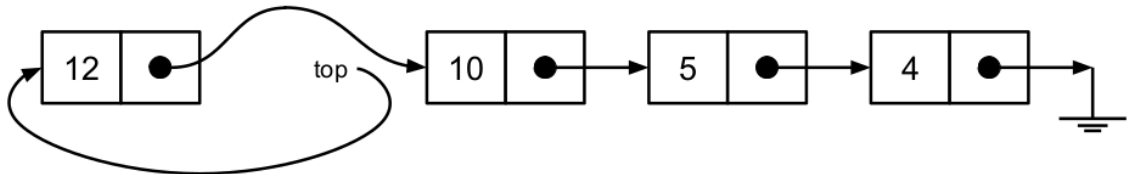
Έστω ότι η στοίβα αρχικά έχει τρεις κόμβους, θέλω να κάνω **Push(12)**



Φτιάχνω έναν **νέο** κόμβο και θέτω την τιμή του ίση με 12

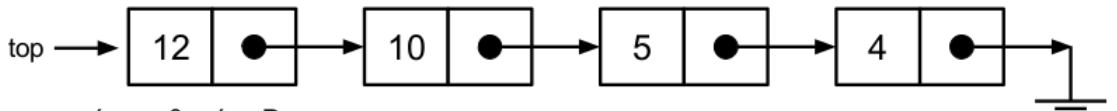


Ο **top** της στοίβας πρέπει να "δείξει" στο **νέο** κόμβο, ο **next** του νέου κόμβου εκεί που **έδειχνε** ο head

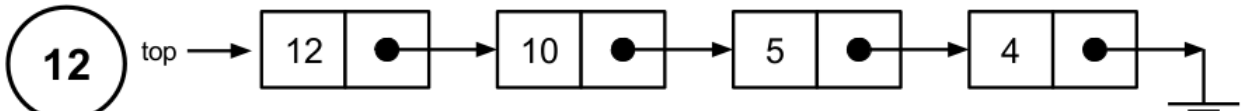


Αντίστοιχα, η pop θα μπορούσε να υλοποιηθεί ως εξής:

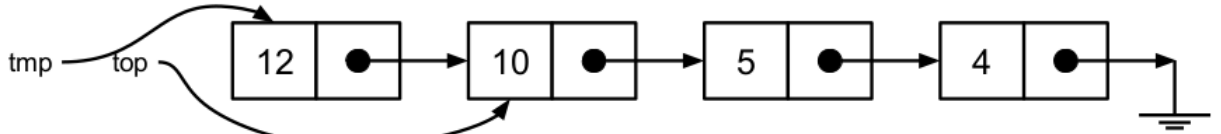
Έστω ότι η στοίβα αρχικά έχει τέσσερις κόμβους, θέλω να κάνω **Pop()**



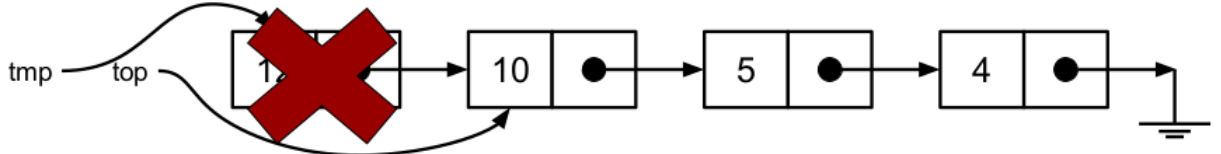
Κρατάω την τιμή που θα γίνει **Pop**



Αλλάζω τον **top** αλλά προσέχω να μη **χάσω** τον κόμβο που πρέπει να σβηστεί

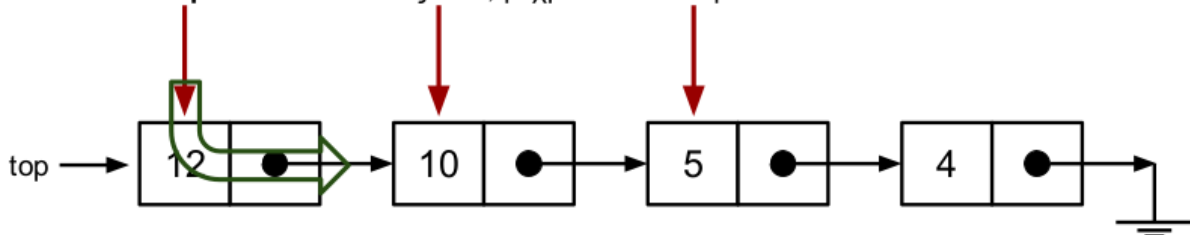


Καθαρίζω τον κόμβο που δε χρειάζεται από τη **μνήμη**



Ας δούμε πώς θα μπορούσαμε να περάσουμε από όλους τους κόμβους μιας στοίβας:

Αν ξεκινήσω με έναν δείκτη που δείχνει όπου και ο top, ο **επόμενος** κόμβος είναι αυτός που δείχνει ο **next** του **top**..**ακολουθώ** τους next, μέχρι να συναντήσω **NULL**



4. Υλοποίηση Στοίβας ως Συνδεδεμένη Λίστα

Να υλοποιηθεί μια στοίβα με χρήση συνδεδεμένης λίστας, η οποία να υποστηρίζει τις λειτουργίες της στοίβας που είδαμε στο εργαστήριο 1, κατάλληλα τροποποιημένες. Να θεωρήσετε ότι στη στοίβα θα αποθηκεύονται θετικοί ακέραιοι αριθμοί. Στη `main` να συμπεριληφθούν παραδείγματα όλων των λειτουργιών.