COMP1511 PROGRAMMING FUNDAMENTALS

LECTURE 16

Starting Revision

LAST WEEK...

- Multi-files
- More linked lists

FOR OUR FINAL WEEK...

- REVISION!

WHERE IS THE CODE?

Live lecture code can be found here:

HTTPS://CGI.CSE.UNSW.EDU.AU/~CS1511/23T1/LIVE/WEEK10/

COURSE FEEDBACK

my Experience surveys

http://myexperience.unsw.edu.au/

REVISION CLASSES

PLEASE BOOK NOW!

Come along and work on revision problems with the support of our lovely tutors:

- FACE TO FACE in Sitar/Kora labs J17:
  - Monday 2-4pm (Sitar) - Anivridh and Gab
- ONLINE:
  - Wednesday 10-12pm - Salina and Liz

Register:
https://www.eventbrite.com.au/e/560086883947

REVISION CLASSES

EXAM ENVIRONMENT

Let me show you the exam environment quickly and the different commands - good for those of you online that are not able to come in before sitting the actual exam :)

LINKED LISTS

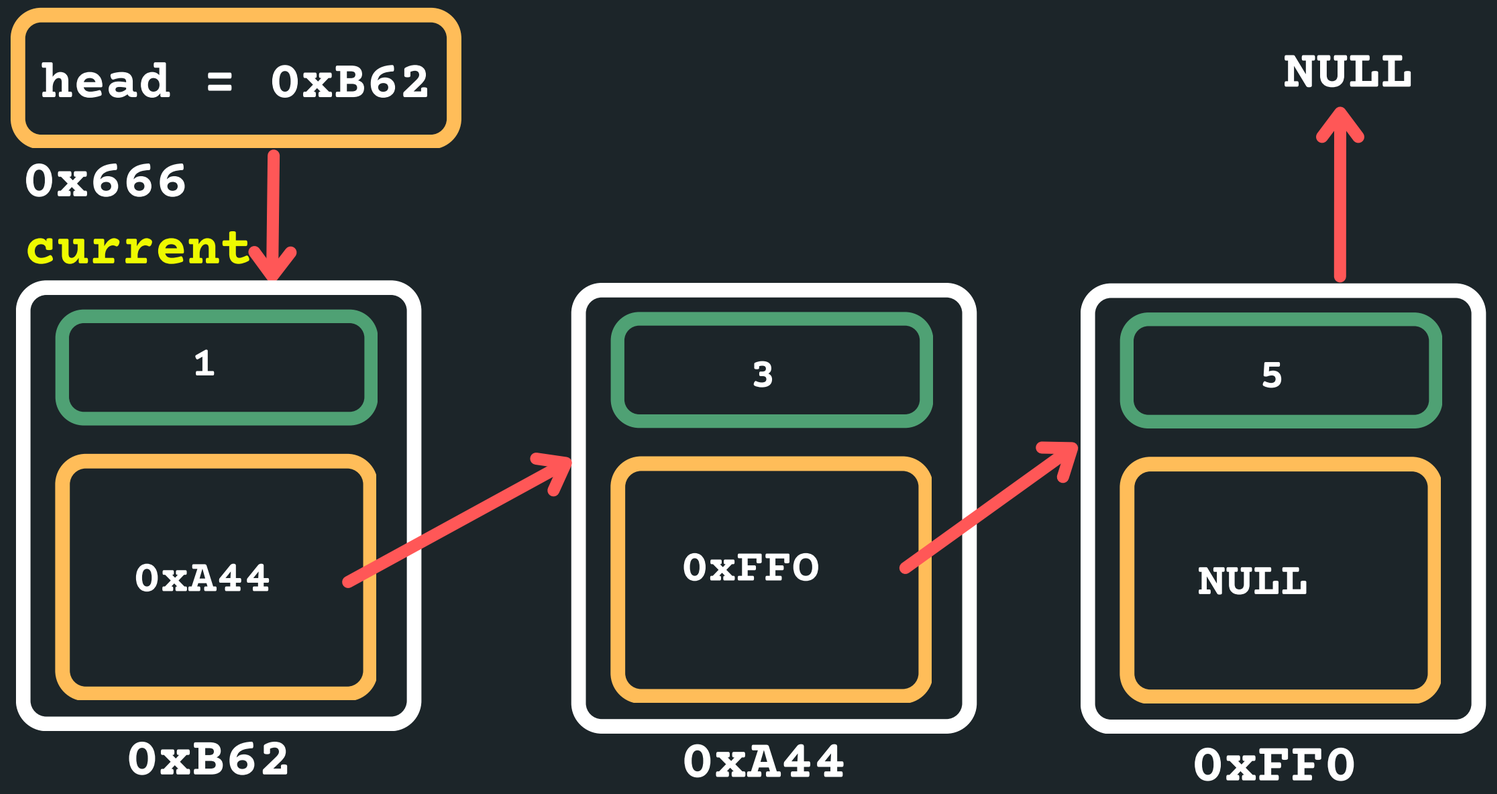
REVISION

- Can only access things sequentially by traversing the whole list
- Can add nodes in as needed (dynamic memory allocation) - by using malloc(sizeof(struct node))
- Can delete nodes as needed (by using free()
- Can check for memory leaks (has everything been freed?) by using: dcc --leakcheck

RECALL A LINKED LIST STRUCTURE

head is just a pointer (not a node!) that holds the address of the first node

End of the list reached when you hit NULL



LINKED LISTS

REVISION

- Some special boundary conditions that you need to consider when you manipulate lists:
  - Empty list
  - List with 1 element
  - Something happening at the beginning of the list
  - Something happening at the end of the list
  - Something will not occur, the item is not in the list (inserting after a number that doesn't exist etc)

THE EXAM

EXAMPLE QUESTION 2

●○○

Perform some computation on a linked list

Given a linked list, print the largest value in that list

Edit the function

```
int largest (struct node *head)
```

EXAMPLE PROBLEM

Problem 1: Find the range (the difference between the biggest term and the smallest term) of a linked list

(see the working files for the details spec)

EXAMPLE PROBLEM

Problem 2: Concatenate two linked lists (join one linked list to another)

EXAMPLE PROBLEM

Problem 3: Given two linked lists, return the difference in the number of items in the two lists.

EXAMPLE PROBLEM

Problem 4: Count all the elements in the linked list that are divisible by 6 and output the count.

EXAMPLE PROBLEM

Problem 5: Given two linked lists, count the number of even numbers in both linked lists and return the difference.

FAKE KAHOOT

Go to **www.menti.com/alqa9z6bmnor**

Mentimeter

Week 10: The final HUZZAH

BREAK TIME...

Did you enjoy your first taste of programming?

EXAMPLE PROBLEM

Problem 6: Insert a specified number into the middle of a linked list. Assume that there is always going to be an even number of numbers in the list before insertion.

EXAMPLE PROBLEM

Problem 7: Delete the first node in the list that is divisible by 6

EXAMPLE PROBLEM

Problem 8: Duplicate every node in the list by inserting the same node after the original node.

EXAMPLE PROBLEM

Problem 9: One that we make up ourselves :)

WHAT DID WE LEARN TODAY?

REVISION

Linked Lists

problem1.c
problem2.c
problem3.c
problem4.c
problem5.c
problem6.c

REVISION

problem7.c
problem8.c
problem9.c

REACH OUT

CONTENT RELATED QUESTIONS

Check out the forum

ADMIN QUESTIONS

cs1511@unsw.edu.au