CSCI 1300

Intro to Computing

Gabe Johnson

Lecture 36 Apr 15, 2013

Linked Lists Coding

Upcoming Homework Assignment

HW #9 **Due: Friday, Apr 19**

Linked Lists

This assignment is all about using C++ pointers and structs to implement a *linked list* data structure and related operations like append and remove. I won't push the due date, but on the plus size it is worth 30 points—so you can get up to 20 points extra credit on this one.

Spring 2013 - CSCI 1300

	Sa	Su	**Mo**	**Tu**	**We**	**Th**	**Fr**	
Jan	12	13	14	15	16	17	18	
	19	20	~~21~~	22	23	24	**25**	**hw 1**
Feb	26	27	28	29	30	31	**1**	**hw 2**
	2	3	4	5	6	7	**8**	**test 1**
	9	10	11	12	13	14	**15**	**hw 3**
	16	17	18	19	20	21	**22**	**hw 4**
Mar	23	24	25	26	27	28	**1**	**test 2**
	2	3	4	5	6	7	**8**	**hw 5**
	9	10	11	12	13	14	**15**	**hw 6**
	16	17	18	19	**20**	21	22	**test 3**
	23	24	~~25~~	~~26~~	~~27~~	~~28~~	~~29~~	
Apr	30	31	1	2	3	4	**5**	**hw 7**
	6	7	8	9	10	11	**12**	**hw 8**
	13	14	15	16	17	18	**19**	**hw 9**
	20	21	22	23	24	25	26	
May	27	28	29	30	1	2	**3**	**project**
	4	5	6	7	8	9	10	

Extra Cred. HW on Apr 26

We will also cover C++ *classes*. I will release a special extra credit homework assignment that will let you (re-) implement linked lists in an object-oriented way.

It will be worth 10 points and will be due April 26.

The algorithms will be the same, but you'll have to approach the problem slightly differently in an OO-context.

Pointer Syntax

```
values
pointers
addresses
```

Pointer Syntax

```
int x=42;
```

`x` is an integer that has the *value* 42. It is declared with the data type `int`, and stores the value 42.

Pointer Syntax

```
int x=42;
int* x_ptr;
```

`x` is an integer that has the *value* 42. It is declared with the data type `int`, and stores the value 42.

`x_ptr` is a *pointer to an int*. It is declared with the data type `int*`, but its value is unspecified.

Pointer Syntax

```
x_ptr = &x;
```

We know the value of an int is some number. The value of `x` is 42.

The *value* of a *pointer variable* is an *address.*

In the above code, we take the *address of* ***x***, and store it in `x_ptr`.

Pointer Syntax

`x` ← This is our int x. Value is 42.

`&x` ← Memory address of x.

`x_ptr` ← int pointer. Value is the address of int x.

`*x_ptr` ← Dereference an int pointer. Value is 42.

Coding Linked Lists

Now we code linked lists!

code / cpp / linked-list