

Introduction to Computer Systems

15-213/18-243 Spring 2009
February 16, 2009

Agenda

- **News**
- **Buflab Questions**
- **Structs overview**
- **Basic Performance Optimizations**
- **Begin Review Exam in 1 week!**

News

- **Buflab due Thursday**
- **Exam in 1 week!**

BufLab Questions



Structs Overview

```
struct node {  
    char * ptr;  
    char data[128];  
}
```



Structs allow you to declare contiguous blocks of memory that can be referred to by name and can include various types such as:

```
set_node(Struct node *node1){  
    strcpy(Node1->data, "hello world!");  
}
```

Structs and alignment

- Elements of a struct can be referred to by `struct->item` but should not be referenced by `(*struct)+1`
- Here is why:
 - Various types have boundary requirements
 - ```
struct node{
 char a;
 int x;
}
```

**This will take 8 bytes not 5**



# Basic Performance Optimizations

## ■ Code Hoisting

```
■ void func(int a, int b, char [] data){
 for(i=0; i<10; i++)
 data[a*b+i] = "A";
}
```

**a\*b can be computed outside of the loop: anytime you can do something before the loop do so, including computing values in the for(...) itself such as for(i=0; i<a+b; i++)**

## ■ Common sub expression elimination

```
■ void func(int a, int b, char[] data){
 for(i=0; i<10; i++){
 if(data[i] < "z" && data[i]!="\n")
 data[i]++
 }
```

**Can declare char c = data[i] in for loop and avoid recalculating it 2-3 times in the if statement**

# Exam Review

- **Topics to review**

- Floating Point!
- Assembly code (writing and interpreting)
- Stack discipline
- structs and alignment

- **Old exams are on the website (with solutions 😊 )**