

This is CS50

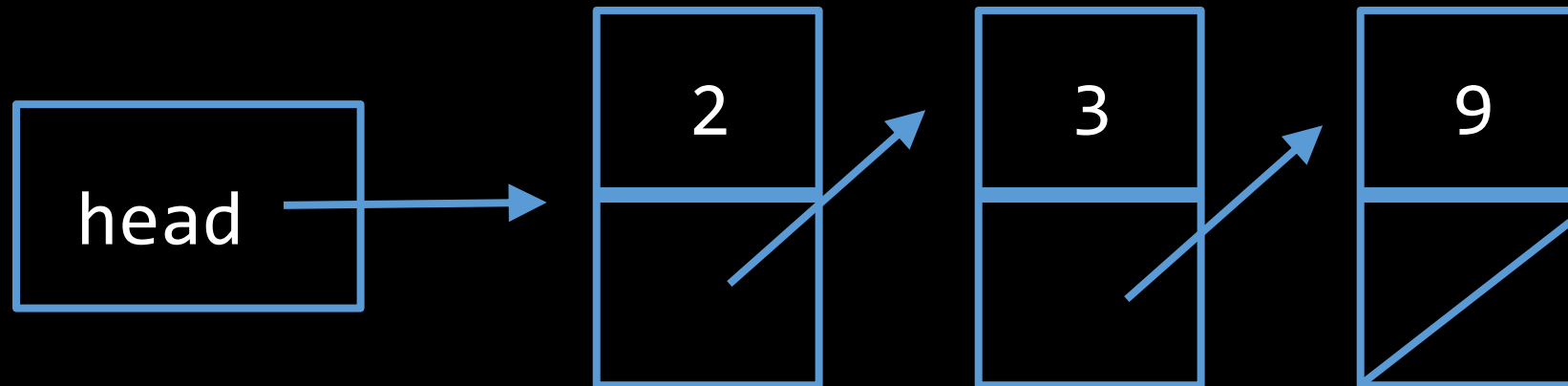
Section, Week 5

TA: Andi Peng

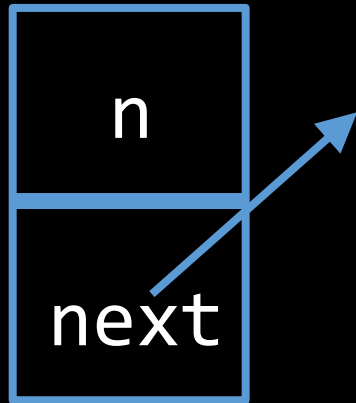
Agenda

- Linked Lists
- Quiz 0 Practice Problems
- Questions?
- Pump Up Speech

Linked Lists

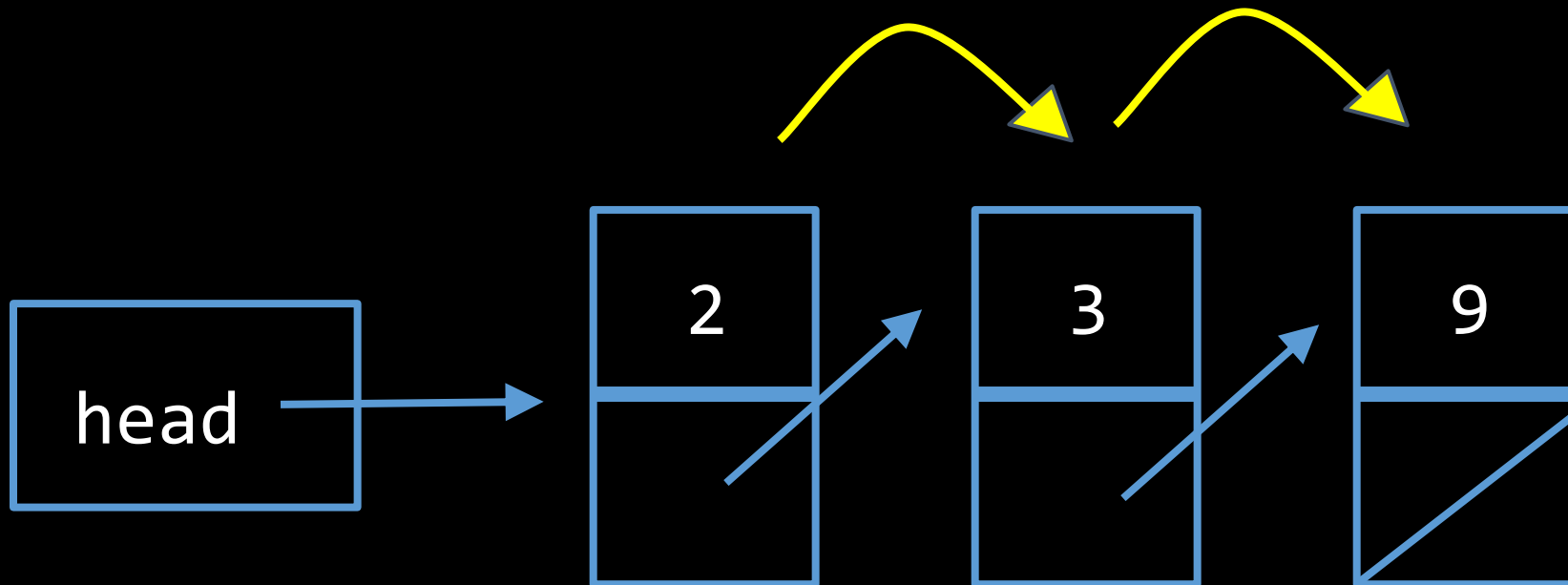


Nodes

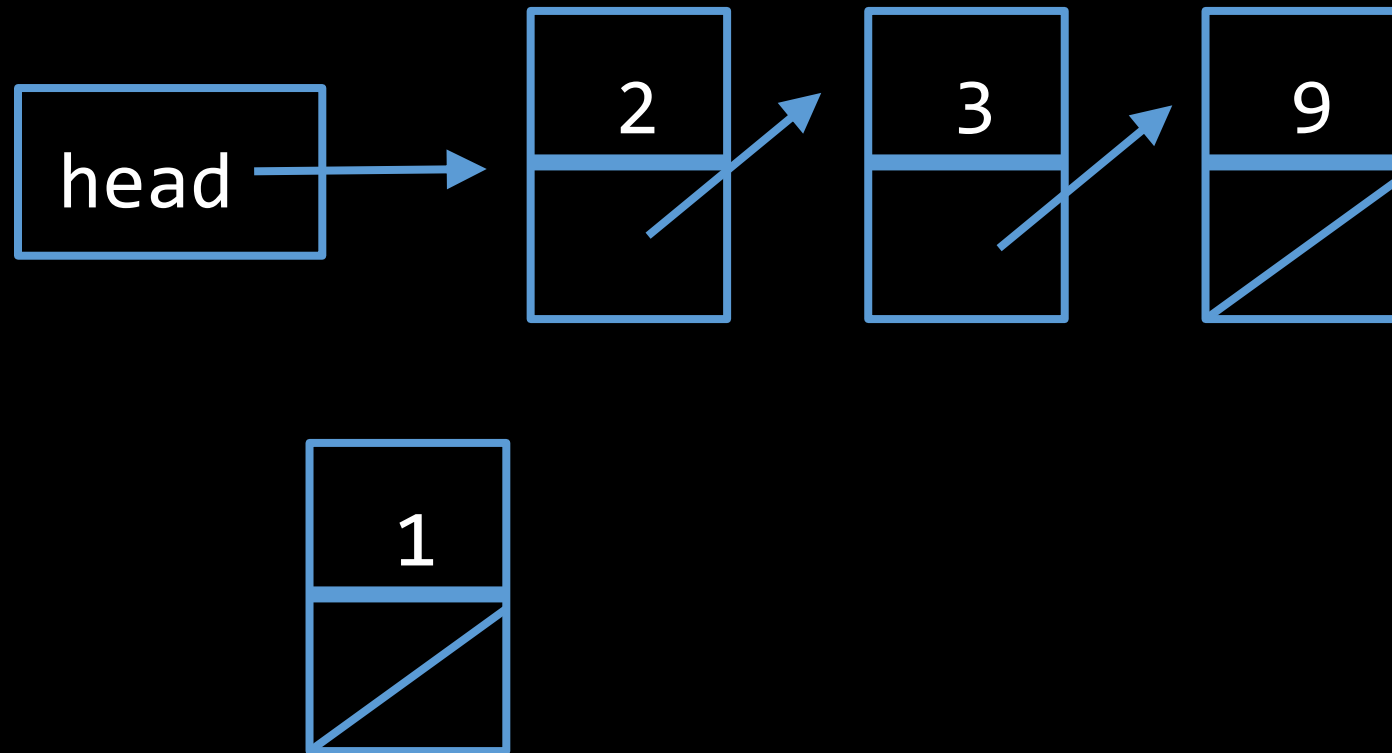


```
typedef struct node
{
    int n;
    struct node* next;
}
node;
```

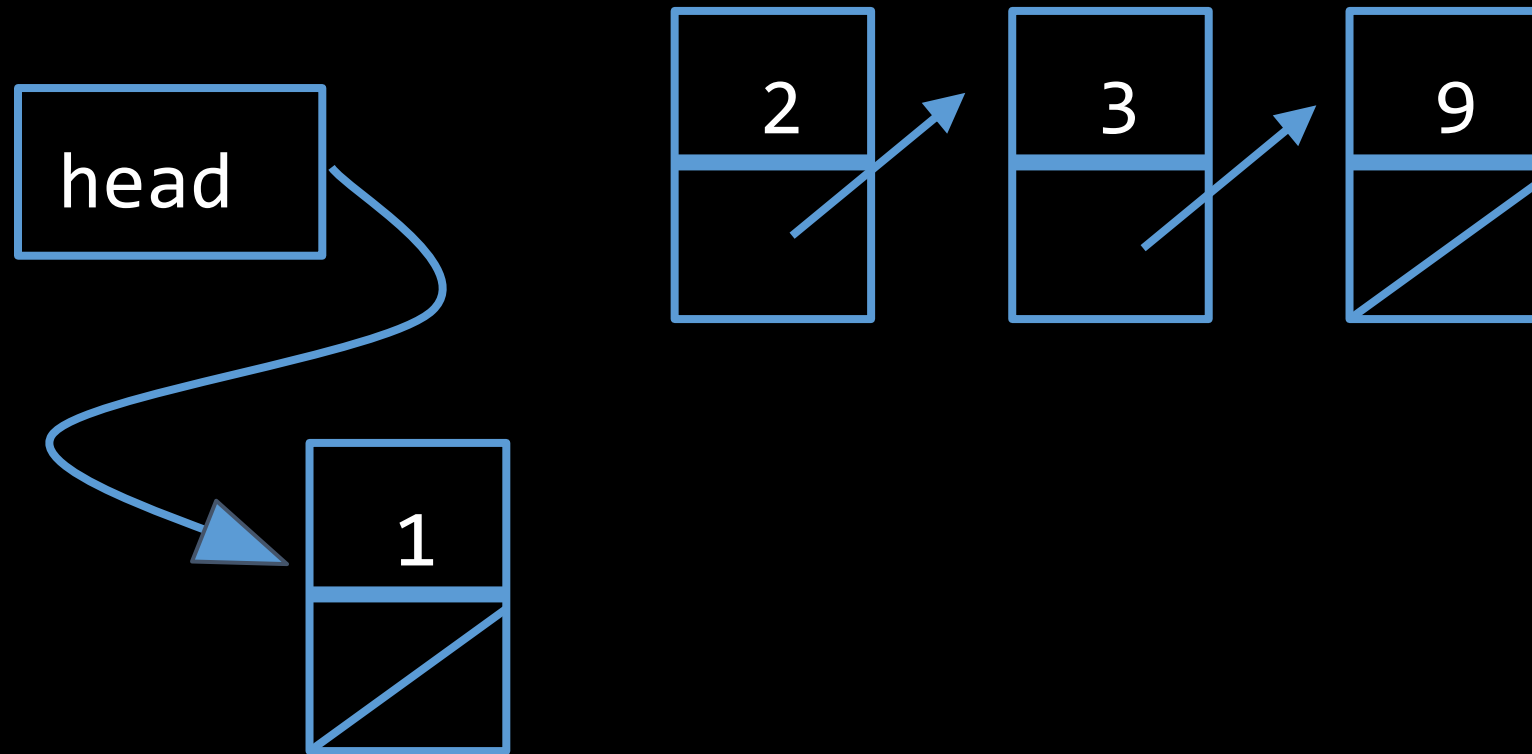
Search



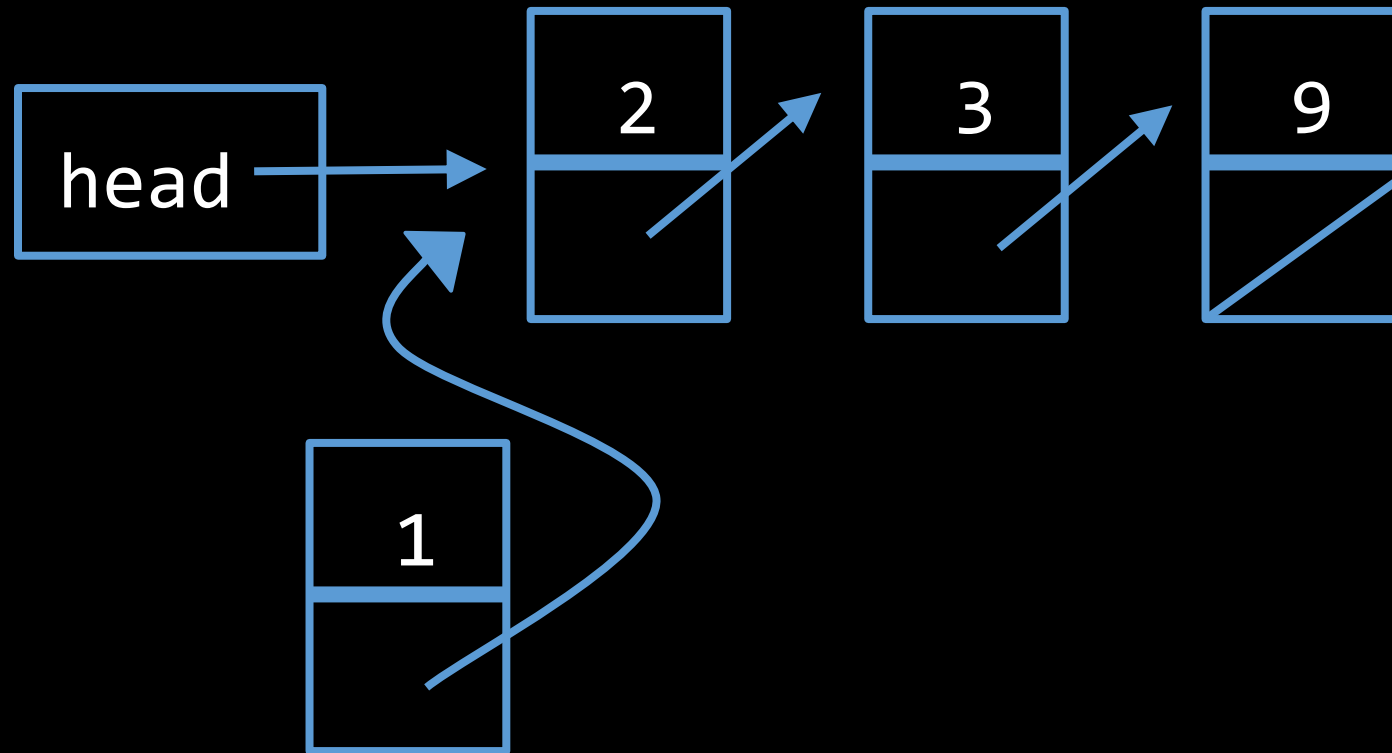
Insertion



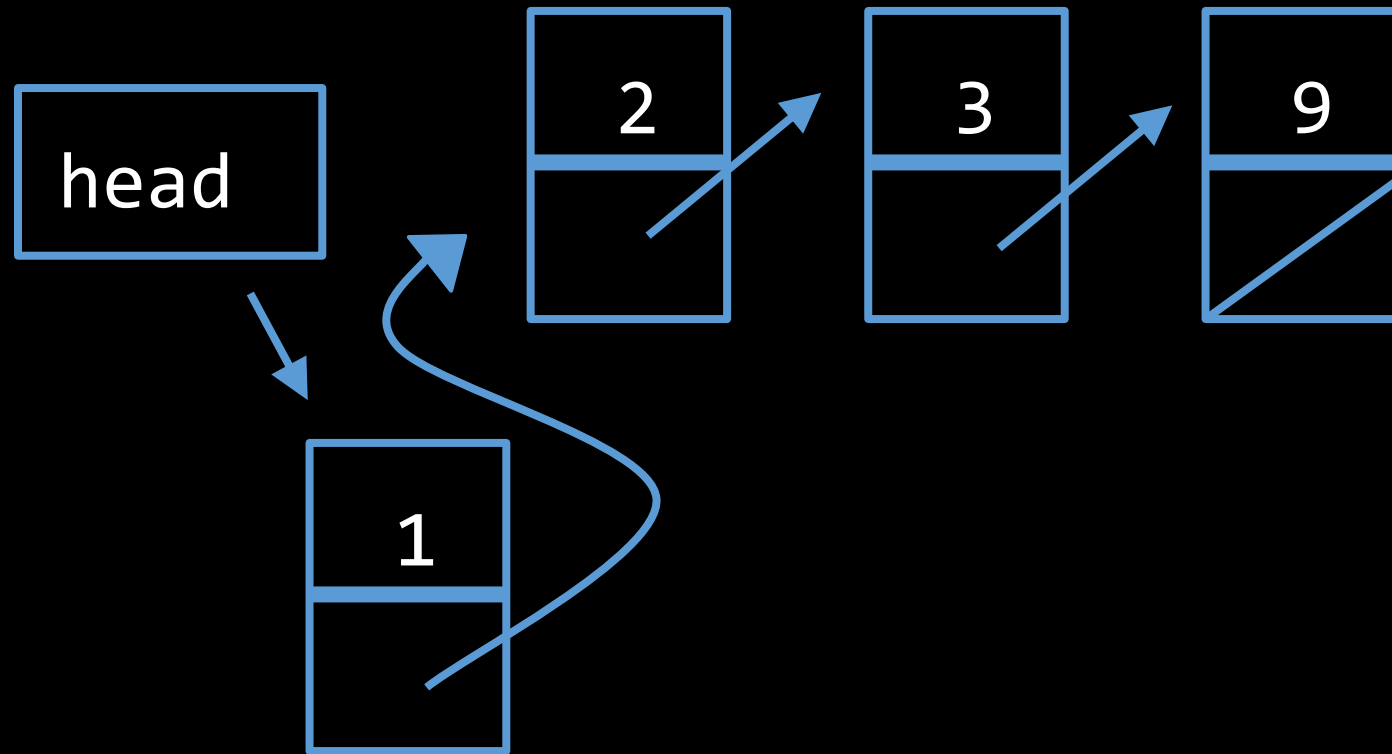
Insertion



Insertion



Insertion



Quiz 0

- Binary. ASCII. Algorithms. Pseudocode. Source code. Compiler. Object code. Scratch. Statements. Boolean expressions. Conditions. Loops. Variables. Functions. Arrays. Threads. Events.
- Linux. C. Compiling. Libraries. Types. Standard output.
- Casting. Imprecision. Switches. Scope. Strings. Arrays. Cryptography.
- Command-line arguments. Searching. Sorting. Bubble sort. Selection sort. Insertion sort. O . Ω . Θ . Recursion. Merge Sort.
- Stack. Debugging. File I/O. Hexadecimal. Strings. Pointers. Dynamic memory allocation.
- Heap. Buffer overflow. Linked lists.
- Hash tables. Tries. Trees. Stacks. Queues.

Quiz 0

- <http://cdn.cs50.net/2015/fall/quizzes/0/yale.html>
- Wednesday
 - if your LAST name starts with A – N, go to the Law School Auditorium
 - If your LAST name starts with O – Z, go to the Davies Auditorium
- Thursday
 - Go to SSS 114

Quiz 0

- Tips
 - Practice writing code by hand
 - Take practice quizzes (under time constraints!)
 - Be familiar with your psets
 - Review lecture and section slides
 - Everything that involves memorization – stick it on your reference sheet

Quiz 0

Convert the following binary numbers to decimal

101010

110010

Convert the following binary numbers to hexadecimal

11111111

01011010

Quiz 0

Convert the following binary numbers to decimal

$$101010 = 42$$

$$110010 = 50$$

Convert the following binary numbers to hexadecimal

$$11111111 = 0xFF$$

$$01011010 = 0x5A$$

Quiz 0

Consider the program below

```
1. #include <stdio.h>
2.
3. int main(void)
4. {
5.     printf("Good luck on the quiz!\n");
6.     return 0;
7. }
```

Quiz 0

What's inside of `stdio.h` that's of interest to the program?

What does `void` signify in line 3?

What does returning 0 from `main`, as in line 6, generally signify?

Quiz 0

What's inside of `stdio.h` that's of interest to the program?

The declaration of `printf` (including its return type and parameters).

What does `void` signify in line 3?

That `main` does not expect any command line arguments.

What does returning 0 from `main`, as in line 6, generally signify?

That the program executed successfully without any errors. By contrast, non-0 values generally indicate failures of some sort.

Quiz 0

Consider the program below

```
#include <stdio.h>

int main(void)
{
    float answer = 1 / 10;
    printf("%.1f\n", answer);
    return 0;
}
```

Quiz 0

When executed, this program prints

0.0

Which is not, of course, one tenth!

Explain why this program thinks that 1 divided by 10 printed to .1 decimals places is something other than 0.1.

Quiz 0

Explain why this program thinks that 1 divided by 10 printed to .1 decimals places is something other than 0.1.

Quiz 0

Consider the program below

```
#include <stdio.h>
#include <cs50.h>

int main(void)
{
    string s = GetString();
    string t = GetString();
    if (s == t)
        printf("You typed the same thing!\n");
    else
        printf("You typed different things!\n");
    return 0;
}
```

Quiz 0

Even when a user types exactly the same English word twice, this program prints

You typed different things!

Explain why this program always responds that the inputs are different, even when the words are the same.

Quiz 0

Explain why this program always responds that the inputs are different, even when the words are the same.

`GetString` returns the address of the first char in a string. Even though a user might type the same English word twice, each string will be stored in a different location in memory, and so the addresses in `s` and `t` will be different.

Quiz 0

Suppose that you've forgotten (as seems to happen annually) in which header file `strlen` is declared, and so you must re-implement it yourself.

Complete the implementation `strlen` below in such a way that it returns the length of `s`.

- Assume that `s` will not be `NULL`
- Assume that `s` will be terminated with `\0`, which does not count as apart of its length.
- For instance, the length of `hello` would be 5. Do not worry about integer overflow.

Quiz 0

```
int strlen(char* s)
{
```

```
}
```

Quiz 0

```
int strlen(char* s)
{
    int n = 0;
    while (s[n] != '\0')
    {
        n++;
    }
    return n;
}
```

Quiz 0

What does the program below print out?

```
void func()
{
    int x = 100;
    printf("%i\n", x);
}
int main(void)
{
    int x = 10;
    func();
    printf("%i\n", x);
}
```

Quiz 0

What does the program below print out?

```
void func()
{
    int x = 100;
    printf("%i\n", x);
}
int main(void)
{
    int x = 10;
    func();
    printf("%i\n", x);
}
```

100
10

Quiz 0

Respond to the following email.

Dear Andi,

I think something is going wrong with my compiler! I'm certain that my code is correct, but I keep seeing Segmentation Fault every time I run my program. What's going on?!?? Please help <3333333

Sincerely, certain that this is not an error.

Quiz 0

Respond to the following email.

Segmentation faults occur when a program attempts to access memory that it doesn't have access to. Common examples might including indexing into an illegal index in an array or trying to dereference a pointer to an arbitrary address in memory.

Quiz 0

Respond to the following email.

Dear Andi,

If Merge sort is so much faster than the other sorts we learned about, why do we even bother ever even coding any of the other types of sorts?

Sincerely, confused on why you make us write harder code.

Quiz 0

Respond to the following email.

While significantly more efficient, Merge sort requires twice as much space in memory (to store values while merging).

Quiz 0

Respond to the following email.

Dear Andi,

In the course of doing pset4, I keep getting this message from Valgrind when running my program. What do I do???

`definitely lost: 40 bytes in 1 blocks`

Sincerely, memory is hard.

Quiz 0

Respond to the following email.

Be sure to free (with `free()`) any memory allocated by `malloc`.

Quiz 0

Search and Sort Run Times

	linear search	binary search	bubble sort	selection sort	insertion sort	merge sort
O	n	$\log(n)$	n^2	n^2	n^2	$n \log(n)$
Ω	1	1	n	n^2	n	$n \log(n)$
Θ				n^2		$n \log(n)$

- O upper bound (in the worst case)
- Ω lower bound (in the best case)
- Θ identical upper and lower bound

Quiz 0

Why is Selection sort's runtime the same in both best and worst case scenarios (even if the list is already sorted?)

Quiz 0

Why is Selection sort's runtime the same in both best and worst case scenarios (even if the list is already sorted?)

Selection sort doesn't know which of its not-yet-sorted elements is smallest until it's traversed them all, since the smallest might be the last. To sort n elements, then, Selection sort must look at first n elements, then $n - 1$ elements, then $n - 2$ elements, and so forth, which adds up to $n(n + 1) / 2$, which is on the order of n^2 .

Questions?

Andi's Pump Up Speech