```
      char* mesg = "Welcome CS341 students!";
```

0. Some stuff *you* will learn (aka Learning Objectives)

Interact with OS in C via **system** calls
Understand how OS allocates, deallocates and accesses memory
Understand **virtual memory**
Create, use, manipulate **processes** and **threads**
Understand how OS **schedules** processes and threads
Communicate and **synchronize** between threads and processes
Determine when **deadlock and race conditions** may occur and how to avoid them
Manipulate **filesystem** structures (inodes etc.)
Communicate across **networks**

1. Why is CS341 hard? AKA "*Look Mom no training wheels!*"

2. What's the difference between a program image and a process?

Overleaf, sketch the contents of the address space of a process:
Where are the Environment vars,Program Arguments,Stack,Heap, Uninitialized vars, Initialized vars, Code?

3. Things to get up to speed on before we can talk about threads or system calls in detail,

C != C++
Lifetime of variables
Arrays
Buffered I/O
Use of * and &
C string gotchas
heap memory allocation

C library I/O (`fprintf,fopen,puts,getchar`...)
uses low-level POSIX *system calls* (`read,write,open`)

4. Explain what is going on in each line and how many bytes are allocated and where.

```
void test() {
  char * t1 = "hi";
  char t2[] = "ab";
}
```

5. Can one process create another process?

6. What is `sizeof(int)`?

7. What is `sizeof(char)`?

8. What is `sizeof(char*)` ?

9. `int A[8];` What is `sizeof(A)`?

10. How many system programmers does it take to change a lightbulb?

11. What are `malloc`, `calloc`, `realloc` and `free`?

12. A program calls `printf("Hello")`
when does the C library call `write`?

13. MPs, Lab assignments, Ed, Honors, Peer Tutoring, Office Hours

14. Resources: cs341.cs.illinois.edu

15. Invited Guests & Extra Credit

Scott Fisher (Alumnus)

Rob Kooper (NCSA Software Engineer)