CS241 #01 Welcome

0. About the course:

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 CS241 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:
You should include at least Environment,Program Arguments,Stack,Heap,Unitialized 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 lower level posix calls (`read`, `write`, `open`)

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

```
01  void test() {
02    char* t1 = "hi";
03    char t2[] = "ab";
04
05    *t2 = 'A';
06    *(t2 + 1) = 'B';
07    t2[1] = 'B';
08    *t1 ='H';
09  }
```

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`? Why are they hard?

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

13 MPs, lab assignments, Piazza policy. Honors course