CS 241 - Review

The final is coming up. Don't Panic! These exercises are meant to help you get ready for the hand coding section. Please try them out on a physical piece of paper before. Assume all function calls work unless otherwise specified.

Warm Ups - Fork-Exec-Wait

Give me a program that uses fork-exec-wait to stat a file (`execlp("stat", ...)`) given in `argv[1]`. If that program has exit code 0, use fork-exec-wait to `touch` the file? No need for error checking.

```c
int main() {

}
```

Memory allocation

Give me a slab buddy allocator. The minimum allocation size is 16 bytes. alloc_size is always a power of 2.

```c
#define HEAP_SIZE 128
typedef struct {
    char used[HEAP_SIZE]; // zeroed out initially
    char *heap[HEAP_SIZE];
} slab_state;
char *allocate(slab_state *slab, int alloc_size) {

}
int free(slab_state *slab, void *ptr) {

}
```

Synchronization

Give me a thread safe ring buffer! Add whatever you want to the struct.

```
#define BUF_SIZE 16
typedef struct {
    void *buffer[BUF_SIZE];
    int read_number;
    int write_number;

} ring_buffer;
// Assume create/destroy are implemented for you
void add_rb(ring_buffer *buffer, void *data) {

}
void *pop_rb(ring_buffer *buffer) {

}
```

Give me a TCP Client

The address is in `argv[1]`, and I want you to send “Hello!” to the server with TCP. You don’t need to worry about setting socket options, free memory, checking for errors.

```
int main(int argc, char *argv[]) {

}
```