# CS107 Final Exam
# Answer Booklet

## CS107 Spring 2019 – Instructor: Nick Troccoli

You may not use any internet devices. You will be graded on functionality – but good style saves time and helps graders understand what you were attempting. You have 180 minutes. We hope this exam is an exciting journey.

**Note:** Only work in this answer booklet will be graded. The backs of pages are available as space for each problem as well.

First Name: _____

Last Name: _____

SUNET ID (part before @stanford.edu): _____

By signing below, I commit to the letter and spirit of the honor code. I agree not to access any unauthorized resources or swap to any other applications for the duration of the exam.

_____

*Sign here*

# Problem 1A

## Problem 1B

# Problem 1C

# Problem 1D

## Problem 2A

```
void remove_at(void *base, size_t nelems, size_t elem_size_bytes,
size_t remove_index, void *dest) {

	// your code here
```

## Problem 2B

```
void set_at(void **base, size_t *p_nelems, size_t elem_size_bytes,
void *elem, size_t index) {

       // your code here
```

## Problem 2C

```
void order_pairs(void *base, size_t nelems, size_t elem_size_bytes,
int (*cmp_fn)(void *, void *)) {

       // your code here
```

## Problem 2D

```
int compare_ints(void *a, void *b) {
	// your code here
```

## Problem 2E

```
// write parameter 1 below


// write parameter 2 below


// write parameter 3 below


// write parameter 4 below


// write parameter 5 below
```

## Problem 2F

```
// write parameter 1 below


// write parameter 2 below


// write parameter 3 below


// write parameter 4 below
```

## Problem 3A

Fill in the blanks below

```
int func2(int x, int y) {
    int z = _____________;
    return _____________ << _____________;
}
void func1(int *arr, int count) {
    int z = _____________;
    for (int i = _____________; i < _____________; i += _____________) {
        _____________ += func2(_____________, _____________);
    }
    printf("%d\n", _____________);
}
```

## Problem 3B

## Problem 3C

## Problem 3D

# Problem 4A

## Problem 4B

## Problem 5A

```
bool is_used(size_t *header) {
    // your implementation here
```

## Problem 5B

```
size_t get_size(size_t *header) {
    // your implementation here
```

## Problem 5C

```
void set_size(size_t *header, size_t new_size) {
    // your implementation here
```

## Problem 5D

```
size_t *get_next_header(size_t *header) {
    // your implementation here
```

## Problem 5E

```
void coalesce_starting_at(size_t *header) {
    // your implementation here
```

## Problem 5F

```
void coalesce_all() {
    // your implementation here
```

## Problem 5G

```
// write your expression below
```

## Problem 5H

```
// parameter 1




// parameter 2




// parameter 3
```