CS 202, Spring 2024
Handout 3 (Class 4)

1. Example to illustrate interleavings: say that thread tid1 executes f() and thread tid2 executes g(). (Here, we are using the term "thread" abstractly. This example applies to any of the approaches that fall under the word "thread".)

a. [this is pseudocode]

```
    int x;

    int main(int argc, char** argv) {

        tid tid1 = thread_create(f, NULL);
        tid tid2 = thread_create(g, NULL);

        thread_join(tid1);
        thread_join(tid2);

        printf("%d\n", x);
    }

    void f()
    {
        x = 1;
        thread_exit();
    }

    void g()
    {
        x = 2;
        thread_exit();
    }
```

What are possible values of x after tid1 has executed f() and tid2 has executed g()? In other words, what are possible outputs of the program above?

b. Same question as above, but f() and g() are now defined as follows:

```
    int y = 12;

    f() { x = y + 1; }
    g() { y = y * 2; }
```

What are the possible values of x?

c. Same question as above, but f() and g() are now defined as follows:

```
    int x = 0;
    f() { x = x + 1; }
    g() { x = x + 2; }
```

What are the possible values of x?

2. Linked list example

```
    struct List_elem {
        int data;
        struct List_elem* next;
    };

    List_elem* head = 0;

    insert(int data) {
        List_elem* l = new List_elem;
        l->data = data;
        l->next = head;
        head = l;
    }
```

What happens if two threads execute insert() at once and we get the following interleaving?

```
    thread 1: l->next = head
    thread 2: l->next = head
    thread 2: head = l;
    thread 1: head = l;
```