ECE326 – Fall 2019: Week 12 Exercise Questions

1. True or False [1 mark each]

Circle T is true, otherwise circle F for false.

1. Lifetime parameter must be added to all structures with non-static references. **(T)** **F**
2. Lifetime elision optimizes the binary by eliminating the need to copy parameters. **T** **(F)**
3. Concurrent programming helps reduce bugs by organizing code into independent threads of execution. **T** **(F)**
4. On a uniprocessor, it is safe to use Rc<T> instead of Arc<T>. **T** **(F)**
5. In Java, the synchronized operator enables synchronization between threads. **T** **(F)**

2. Channel **[10 marks]**

Use mpsc::channel and multiple threads to improve the performance of large square matrix multiplication.

3. Dining Philosopher **[10 marks]**

In the dining philosopher problem, there are N philosophers and N chopsticks in between each pair of philosophers. As you may know, you need a pair of chopsticks to be able to eat. A philosopher must successfully acquire both chopsticks to his/her left and right before proceeding to eat. Simulate this problem by creating one thread per philosopher, and use a monitor to synchronize the use of chopsticks.

4. Findall **[10 marks]**

Write a function that takes two string slices, text and word, and return a vector of all occurrences of the word in text in string slices. Note, you may need to "fix" the function signature. You may assume the text to consist of only ascii characters.

```
fn findall(text: &str, word: &str) -> Vec<&str>;
```