Operating Systems

Homework 5

Question 1

```
// shared memory
semaphore match_sem(0);
semaphore paper_sem(0);
semaphore tobacco_sem(0);
Semaphore agent_sem(1);
```

```
// process with matches (needs paper and tobacco)
while (true) {
        if (paper_sem > 0 && tobacco_sem > 0) {
                paper_sem.wait();
                tobacco_sem.wait();
                smoke();
                agent_sem.signal();
        }
}
```

```
// agent
while (true) {
        agent_sem.wait();
        int random = rand() % 3;
        if (random != 2) {
                match_sem.signal();
        }
        if (random != 1) {
                paper_sem.signal();
        }
        if (random != 0) {
                tobacco_sem.signal();
        }
}
```

Question 2

```
// shared memory
semaphore mutex(1);
semaphore barber(0);
semaphore full_chairs(0);
queue<customer> chairs;{
int empty_chairs = 3;
```

```
// barber code
while (true) {
	full_chairs.wait();
	customer current_customer = chairs.dequeue();
	mutex.wait();
	++empty_chairs;
	mutex.signal();
	cut_hair(current_customer);
```

```
// customer code
mutex.wait();
if (empty_chairs) {
	--empty_chairs;
	mutex.signal();
	chairs.enqueue(a_cusomer);
	full_chairs.signal();
} else {
	mutex.signal();
}
```