# COMP3500: Project 3-2 Cats and Mice Design

**🟊: >85%, 🟊🟊: 70-85%, 🟊🟊🟊: 55-70%, 🟊🟊🟊🟊: 40-55%, 🟊🟊🟊🟊🟊: < 40%**

**🟊 Exercise 1:** Can you use a diagram to illustrate the following specifications? (1.5 Min)

* Two cat food dishes
* 6 cats and 2 mice
* Only 1 mouse or cat may eat at a given dish at any one time
* If a cat is eating at either dish, a mouse attempting to eat from the other dish will be seen and therefore eaten
* When cats aren't eating, they will not see mice eating.

**🟊🟊 Exercise 2:** Please list all possible cases. (1.5 Min)

**🟊🟊🟊🟊🟊 Exercise 3:** What semaphores should you declare? (2 Min)

**🟊🟊🟊🟊 Exercise 4:** Complete the following algorithm - First Cat and No Mouse (1.5 Min)

;

If (all\_dishes\_available) {

all\_dishes\_availalbe = ;

; /\* let first cat in \*/

}

cats\_wait\_count++;

;

wait(cats\_queue); /\*first cat in, other wait\*/

If (no\_cat\_eat) {

no\_cat\_eat = ;

first\_cat\_eat = ;

}

else first\_cat\_eat = ;

**🟊🟊🟊 Exercise 5:** How does the first cat control the kitchen? (1.5 Min)

if (first\_cat\_eat == true) {

;

if (cat\_wait\_count > 1) {

another\_cat\_eat = ;

; /\*let another cat in\*/

}

;

}

kprintf(“Cat in the kitchen.\n”); /\*cat name \*/

**🟊🟊 Exercise 6:** Complete the following algorithm - All cats (first cat and non-first cat) in the kitchen (1.5 Min)

; /\*protect shared variables\*/

if (dish1\_busy == false) {

dish1\_busy = ;

mydish = 1;

}

else {

assert(dish2\_busy == false);

dish2\_busy = ;

mydish = 2;

}

;

kprint(“Cat eating.\n”); /\* cat name \*/

clocksleep(1); /\* enjoys food \*/

kprint(“Finish eating.\n”); /\* done. \*/