**Basics of Interprocess Communication**

**What is a race condition?**

Imagine: Two processes are running at the same time. Both increase a number, saved at the same variable, by one. Then they are printing the number. If the first process increases the number and doesn’t print it and then the second process increases the number, both processes are printing the same number. In some cases, this would be a problem.

**Disabling Interrupts**

**Why is it impossible to achieve Mutual Exclusion via disabling interrupts on a multi-core machine?**

Because in a multicore cpu the disabling only applies to that core and not to the entire system. So two processes could be in the critical region at the same time.

**Why is it dangerous to give user processes the power to disable interrupts?**

**Play through the two scenarios of the handout of Peterson's solution. Document how it works.**

Me is setted to 0, 0 enters the region

**Play through the scenario which makes the strict alternation approach fail. Document how it fails.**

**What is the meaning of the variable “loser” in Peterson's solution? When does it have any effect?**

If only two processes are running

Extend the given functions such they can handle three processes.