Cats – catsem function

This function represents the simulation of a cat. Upon execution each "cat" takes a resource of catsemaphore, which will be initialized to either NCATS or 0. If the catsemaphore is initialized to 0, all cats will wait and be halted until all mice are done eating. Otherwise a cat will take a bowl. Depending on the number of bowls available cats may wait for bowls to be unoccupied before continuing. The cat begins eating and increments catseating variable. While cats are eating a for loop asserting no mice are eating (on a volatile variable) is ran. After these iterations the catseating variable is decremented and the cat replaces the bowl resource. After a cat finishes they increment a volatile variable catsdoneeating. When all cats are done eating, and if no mice have eaten, we allocate NMICE resources to mousesemaphore so mice can eat.

Mice – mousesem function

This is implemented the exact same way only visa versa on mice and cat variables. Take note that if a mouse or cat starts eating, it will increase the count of catseating or miceeating. If a cat is eating it asserts (miceeating == 0), or a mosue is eating it asserts (catseating == 0). Thus assuring niether are eating at the same time.

Printing

A semaphore "printsem" with a resource of 1 is used before every print statement then given back after the end of the print statement assuring no prints are interlaced.

Cycles

Implemented using a volatile variable, being assigned to a variable within catsem or mouessem after it begins eating. It is also incremented there too.

Bowl Accounting

Used a interger array of 1's and 0's like a binary array. The size of this array is NFOODBOWLS. A 1 represents that bowl is present while a 0 represents it is taken, the index is the bowl number. Each catsem or mousesem thread takes ownership of a bowl by setting a varible in their scope to the first available bowl. Since bowlsem accounts for bowl resources this code must be called after P(bowlsem) and before V(bowlsem) so there will always be a bowl available in the array.

Initialization – catmousesem function

Cats or mice may start first, one must be set to 0, while the other to NCATS or NMICE respectively. Also the bowl array must be initialized to all 1's. As well as all variables to 0, with the exception of cycle set as 1 as an offset for numbering.