Avoiding bad stories

Algorithms 2.3, 2.4

Slides 2.12, 2.17

2.3 - can guarantee n=2.

2.4 - cannot → race condition

We must say what the atomic states are.

The CS problem = avoid: Preprotocol, Preprotocol, CS, CS

The CS-problem can be solved by:

- Test and Set, compare and swap, ...
- Or just swap

HW-Solution

Preprotocol

loop

swap(local, common)

until local = true

CS

Swap(local, common)

Swap: atomic

two red, one green token

invariant since its true all the time

SW-Solution

Semaphores!

Definition: <Value, set of blocked procs>

initialized: <k, empty>

two operations, Process P calls sem S

Wait(S)

if k>0 then k:=k-1 else block P and add it to set

Signal(S)

if empty set then k:=k+1 else take a q from set and unblock it

Signal is undefined on a semaphore when k=1.

Invariants

k>=0

k = k.init + #signals - #waits

Proof by induction.

Initially true k = k.init + 0 - 0

The only changes are by signals and waits