PRINCIPLES OF OPERATING SYSTEMS

LECTURE – 24
Semaphores and Bounded Buffer

Semaphores

- ***Semaphore*** is a type of generalized lock
  - Defined by Dijkstra in the last 60s
  - Main synchronization primitives used in UNIX
  - Consist of a positive integer value
  - Two operations
    - ***P()***: an atomic operation that waits for semaphore to become positive, then decrement it by 1
    - ***V()***: an atomic operation that increments semaphore by 1 and wakes up a waiting thread at P(), if any.

Semaphores vs. Integers

- No negative values
- Only operations are P() and V()
  - Cannot read or write semaphore values
  - Except at the initialization times
- Operations are atomic
  - Two P() calls cannot decrement the value below zero
  - A sleeping thread at P() cannot miss a wakeup from V()

Binary Semaphores

- A ***binary semaphore*** is initialized to 1
- P() waits until the value is 1
  - Then set it to 0
- V() *sets* the value to 1
  - Wakes up a thread waiting at P(), if any

Two Uses of Semaphores

1. Mutual exclusion
   - Lock was designed to do this

```
lock->acquire();
// critical section
lock->release();
```

Two Uses of Semaphores

1. Mutual exclusion
    1. The lock function can be realized with a binary semaphore: semaphore subsumes lock.
        - Semaphore has an initial value of 1
        - P() is called before a critical section
        - V() is called after the critical section

```
semaphore litter_box = 1;
P(litter_box);
// critical section
V(litter_box);
```

Two Uses of Semaphores

1. Mutual exclusion
   - Semaphore has an initial value of 1
   - P() is called before a critical section
   - V() is called after the critical section

```
semaphore litter_box = 1;
P(litter_box);
// critical section
V(litter_box);
```

litter_box = 1

Two Uses of Semaphores

1. Mutual exclusion
   – Semaphore has an initial value of 1
   – P() is called before a critical section
   – V() is called after the critical section

```
semaphore litter_box = 1;
P(litter_box); // purrr…
// critical section
V(litter_box);
```

litter_box = 1 $\rightarrow$ 0

Two Uses of Semaphores

1. Mutual exclusion
   - Semaphore has an initial value of 1
   - P() is called before a critical section
   - V() is called after the critical section

```
semaphore litter_box = 1;
P(litter_box);
// critical section
V(litter_box);
```

$\boxed{\text{litter\_box = 0}}$

Two Uses of Semaphores

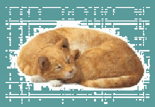
1. Mutual exclusion
   – Semaphore has an initial value of 1
   – P() is called before a critical section
   – V() is called after the critical section

```
semaphore litter_box = 1;
P(litter_box); // meow…
// critical section
V(litter_box);
```

litter_box = 0

Two Uses of Semaphores

1. Mutual exclusion
   - Semaphore has an initial value of 1
   - P() is called before a critical section
   - V() is called after the critical section

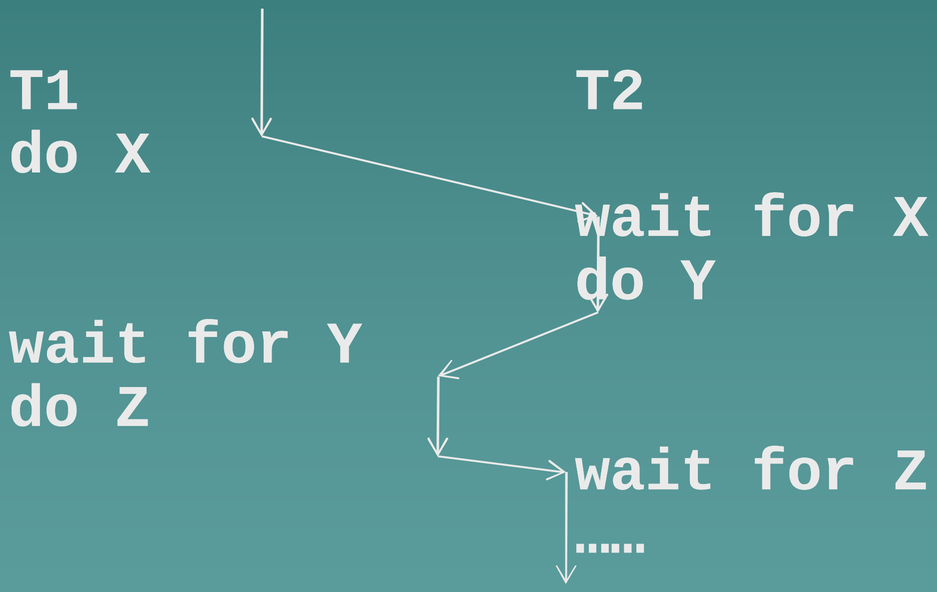


```
semaphore litter_box = 1;
P(litter_box);
// critical section
V(litter_box);
```

litter_box = 0 → 1

Two Uses of Semaphores

2. Synchronization: Enforcing some order between threads



```
T1                T2
do X
                  wait for X
                  do Y
wait for Y
do Z
                  wait for Z
                  ……
```

Two Uses of Semaphores

2. Synchronization
   - Semaphore usually has an initial value of 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;

Left_Paw() {                Right_Paw() {
  slide_left();               P(wait_left);
  V(wait_left);               slide_left();
  P(wait_right);              slide_right();
  slide_right();              V(wait_right);
}                           }
```

Two Uses of Semaphores

2. Scheduling
   – Semaphore usually has an initial value of 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;
```

wait_left = 0
wait_right = 0

```
Left_Paw() {                  Right_Paw() {
  slide_left();                 P(wait_left);
  V(wait_left);                 slide_left();
  P(wait_right);                slide_right();
  slide_right();                V(wait_right);
}                             }
```

Two Uses of Semaphores

2. Scheduling
   – Semaphore usually has an initial value of 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;
```

```
wait_left = 0
wait_right = 0
```

```
Left_Paw() {
  slide_left();
  V(wait_left);
  P(wait_right);
  slide_right();
}
```

```
Right_Paw() {
  P(wait_left);
  slide_left();
  slide_right();
  V(wait_right);
}
```

Two Uses of Semaphores

2. Scheduling
   - Semaphore usually has an initial value of 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;
```

wait_left = 0
wait_right = 0

```
Left_Paw() {                Right_Paw() {
  slide_left();               P(wait_left);
  V(wait_left);               slide_left();
  P(wait_right);              slide_right();
  slide_right();              V(wait_right);
}                           }
```

wait

Two Uses of Semaphores

2. Scheduling
   – Semaphore usually has an initial value of 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;
```

wait_left = 0
wait_right = 0

```
Left_Paw() {                  Right_Paw() {
  slide_left();                 P(wait_left);
  V(wait_left);                 slide_left();
  P(wait_right);                slide_right();
  slide_right();                V(wait_right);
}                             }
```

Two Uses of Semaphores

2. Scheduling
   - Semaphore usually has an initial value of 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;

Left_Paw() {                Right_Paw() {
  slide_left();               P(wait_left);
  V(wait_left);               slide_left();
  P(wait_right);              slide_right();
  slide_right();              V(wait_right);
}                           }
```

wait_left = 0 →1
wait_right = 0

Two Uses of Semaphores

2. Scheduling
   - Semaphore usually has an initial value of 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;

Left_Paw() {                 Right_Paw() {
  slide_left();                P(wait_left);
  V(wait_left);                slide_left();
  P(wait_right);               slide_right();
  slide_right();               V(wait_right);
}                            }
```

wait_left = 1 $\rightarrow$ 0
wait_right = 0

Two Uses of Semaphores

2. Scheduling
   - Semaphore usually has an initial value of 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;

Left_Paw() {                Right_Paw() {
  slide_left();               P(wait_left);
  V(wait_left);               slide_left();
  P(wait_right);              slide_right();
  slide_right();              V(wait_right);
}                           }
```

```
wait_left = 0
wait_right = 0
```

Two Uses of Semaphores

2. Scheduling
   – Semaphore usually has an initial value of 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;

Left_Paw() {                Right_Paw() {
  slide_left();               P(wait_left);
  V(wait_left);               slide_left();
  P(wait_right);              slide_right();
  slide_right();              V(wait_right);
}                           }
```

wait_left = 0
wait_right = 0

wait

Two Uses of Semaphores

2. Scheduling
   – Semaphore usually has an initial value of 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;
```

```
wait_left = 0
wait_right = 0
```

```
Left_Paw() {                  Right_Paw() {
  slide_left();                 P(wait_left);
  V(wait_left);                 slide_left();
  P(wait_right);                slide_right();
  slide_right();                V(wait_right);
}                             }
```

Two Uses of Semaphores

2. Scheduling
   – Semaphore usually has an initial value of 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;
```

wait_left = 0
wait_right = 0 $\rightarrow$1

```
Left_Paw() {                Right_Paw() {
  slide_left();               P(wait_left);
  V(wait_left);               slide_left();
  P(wait_right);              slide_right();
  slide_right();              V(wait_right);
}                           }
```

Two Uses of Semaphores

2. Scheduling
   - Semaphore usually has an initial value of 0

wait_left = 0
wait_right = 1 → 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;

Left_Paw() {                Right_Paw() {
  slide_left();               P(wait_left);
  V(wait_left);               slide_left();
  P(wait_right);              slide_right();
  slide_right();              V(wait_right);
}                           }
```

Two Uses of Semaphores

2. Scheduling
   - Semaphore usually has an initial value of 0

```
semaphore wait_left = 0;
semaphore wait_right = 0;
```

wait_left = 0
wait_right = 0

```
Left_Paw() {                  Right_Paw() {
  slide_left();                 P(wait_left);
  V(wait_left);                 slide_left();
  P(wait_right);                slide_right();
  slide_right();                V(wait_right);
}                             }
```

Two Uses of Semaphores

2. Synchronization
   – Semaphore usually has an initial value of 0

```
semaphore s1 = 0;
semaphore s2 = 0;

A() {                       B() {
  write(x);                   P(s1);
  V(s1);                      read(x);
  P(s2);                      write(y);
  read(y);                    V(s2);
}                           }
```