

Inter Process Communication





Processes in the system

Processes running concurrently may be -

Independent (cannot affect or be affected by other process)

Or

Cooperating (can affect or be affected by other process)

Process cooperation is needed for -

- Information sharing
- Computational speedup
- Modularity
- Convenience

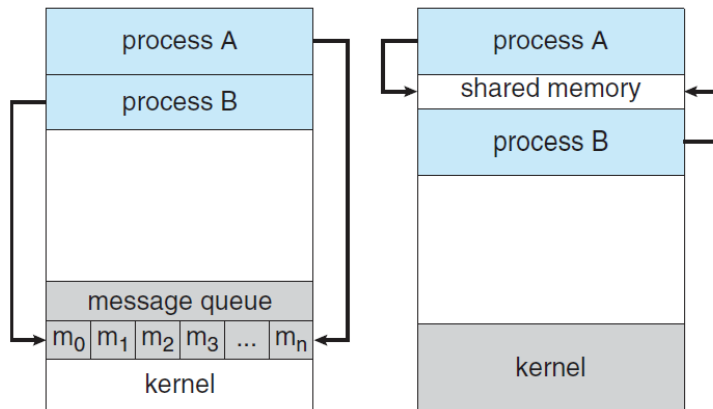


Inter Process Communication

IPC is a ***mechanism*** to exchange data and information among processes.

Two fundamental model of IPC -

1. Shared Memory
2. Message Passing

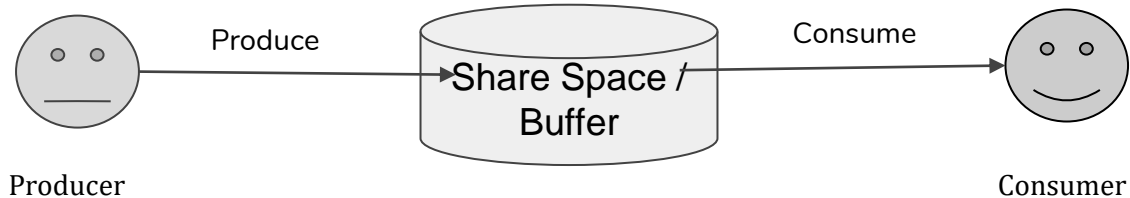




Shared Memory System (Producer-Consumer Problem)

Producer: produces products for consumer

Consumer: consumes products provided by producer



Producer-Consumer Problem (Producer)

```
item next_produced;

while (true) {
    /* produce an item in next_produced */

    while (((in + 1) % BUFFER_SIZE) == out)
        ; /* do nothing */

    buffer[in] = next_produced;
    in = (in + 1) % BUFFER_SIZE;
}
```

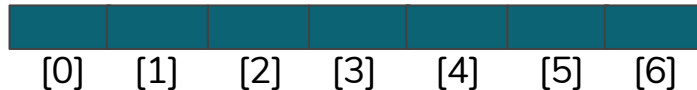
in: next free position in buffer
out: first full position in buffer

Both initialized with 0.

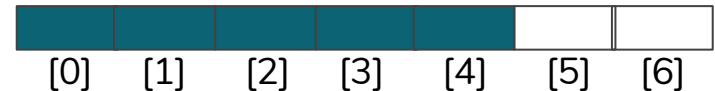
in = 0
out = 0

Here, BUFFER_SIZE = 7

When buffer is full,
in = 6, out = 0



When buffer is not full,
in = 5, out = 0





Producer-Consumer Problem (Consumer)

```
item next_consumed;

while (true) {
    while (in == out)
        ; /* do nothing */

    next_consumed = buffer[out];
    out = (out + 1) % BUFFER_SIZE;

    /* consume the item in next_consumed */
}
```

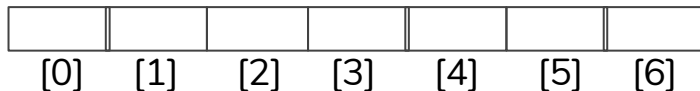
in: next free position in buffer
out: first full position in buffer

Both initialized with 0.

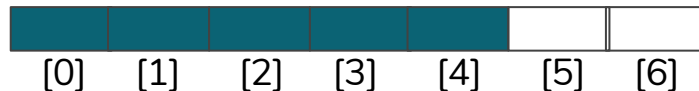
in = 0
out = 0

Here, BUFFER_SIZE = 7

When buffer is empty,
in = 0 , out = 0



When buffer is not empty,
In = 5, out = 0

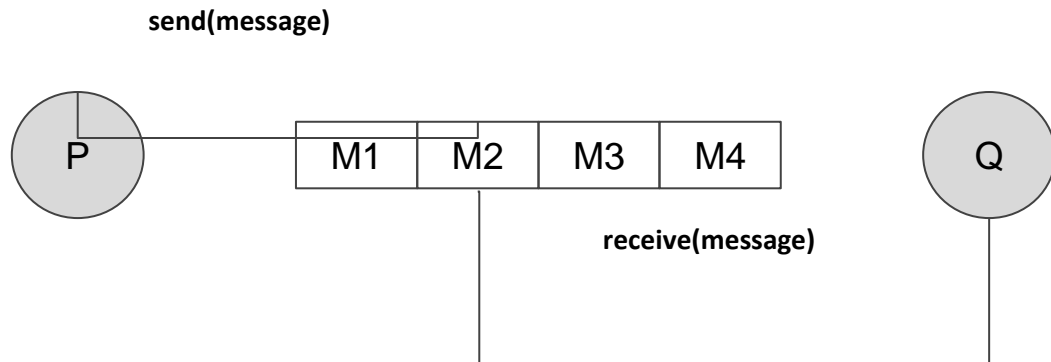




Message Passing System

If processes P and Q want to communicate, they must send messages to and receive messages from each other.

A communication link must exist between P and Q.



- Useful for exchanging small amount of data
- More suited for distributed systems than shared memory