

Inter-Process Communication

Shared Memory

Next: shared memory

Inter-Process Communication

Shared Memory

Next: shared memory

In early computers, all memory was shared between processes: one process could easily write to the memory allocated to another process

Inter-Process Communication

Shared Memory

Next: shared memory

In early computers, all memory was shared between processes: one process could easily write to the memory allocated to another process

This is generally a bad idea, so is now prevented by the kernel (recall MMUs and read/write flags)

Inter-Process Communication

Shared Memory

Next: shared memory

In early computers, all memory was shared between processes: one process could easily write to the memory allocated to another process

This is generally a bad idea, so is now prevented by the kernel (recall MMUs and read/write flags)

On the other hand, access to memory is very fast, so we might want to use it for IPC

Inter-Process Communication

Shared Memory

Next: shared memory

In early computers, all memory was shared between processes: one process could easily write to the memory allocated to another process

This is generally a bad idea, so is now prevented by the kernel (recall MMUs and read/write flags)

On the other hand, access to memory is very fast, so we might want to use it for IPC

Just like using files: A writes to memory, B reads from it

Inter-Process Communication

Shared Memory

Next: shared memory

In early computers, all memory was shared between processes: one process could easily write to the memory allocated to another process

This is generally a bad idea, so is now prevented by the kernel (recall MMUs and read/write flags)

On the other hand, access to memory is very fast, so we might want to use it for IPC

Just like using files: A writes to memory, B reads from it

Again, this goes against the original design of an OS, so must be carefully set up and controlled

Inter-Process Communication

Shared Memory

And, also just like files we have the issues of

- Which area of memory to use? A well-known area, or per-process areas?

Inter-Process Communication

Shared Memory

And, also just like files we have the issues of

- Which area of memory to use? A well-known area, or per-process areas?
- How does B know when data has arrived? Memory is “always there” unlike files which can be created and removed; so when polling memory it can be hard to know if you are reading the data you want or some previous junk that happened to be lying around

Inter-Process Communication

Shared Memory

And, also just like files we have the issues of

- Which area of memory to use? A well-known area, or per-process areas?
- How does B know when data has arrived? Memory is “always there” unlike files which can be created and removed; so when polling memory it can be hard to know if you are reading the data you want or some previous junk that happened to be lying around
- So A might write a special value to a specific memory location to flag that the data is complete; but again B must poll this location to see when this is done

Inter-Process Communication

Shared Memory

And, also just like files we have the issues of

- Which area of memory to use? A well-known area, or per-process areas?
- How does B know when data has arrived? Memory is “always there” unlike files which can be created and removed; so when polling memory it can be hard to know if you are reading the data you want or some previous junk that happened to be lying around
- So A might write a special value to a specific memory location to flag that the data is complete; but again B must poll this location to see when this is done
- The memory protections must be set properly to allow only the authorised processes to read or write it

Inter-Process Communication

Shared Memory

The speed of shared memory means that it is very good for IPC, as long as it is supported by further mechanisms like signals or semaphores to flag when data is ready



Inter-Process Communication

Shared Memory

The speed of shared memory means that it is very good for IPC, as long as it is supported by further mechanisms like signals or semaphores to flag when data is ready

More on shared memory when we get to memory management



Inter-Process Communication

Shared Memory

The speed of shared memory means that it is very good for IPC, as long as it is supported by further mechanisms like signals or semaphores to flag when data is ready

More on shared memory when we get to memory management

Exercise. Compare shared memory and pipes

