Linux Fork/Exec Example

```c
int child_pid_or_zero = fork();
if (child_pid_or_zero == 0) {
    execvp("ls", argv);
} else {
    waitpid(child_pid_or_zero, &status, options);
};
```
Windows Process Creation

BOOL CreateProcess(
    LPCTSTR lpApplicationName,
    LPTSTR lpCommandLine,
    LPSECURITY_ATTRIBUTES lpProcessAttributes,
    LPSECURITY_ATTRIBUTES lpThreadAttributes,
    BOOL bInheritHandles,
    DWORD dwCreationFlags,
    LPVOID lpEnvironment,
    LPCTSTR lpCurrentDirectory,
    LPSTARTUPINFO lpStartupInfo,
    LPPROCESS_INFORMATION lpProcessInformation
);

WaitForSingleObject(lpProcessInformation->hProcess, INFINITE);
Thread Creation

```cpp
#include <thread>
...
std::thread t(func);
...
t.join();

void func()
{
    /* This code will run concurrently with the code to the left. */
    ...
}
```
Processes vs. Threads

A thread is part of a process:

- A process can contain many threads
- A process also contains state
Create New Thread

Create a new thread:

```cpp
new std::thread(...)```

New thread shares state with existing threads.
Create New Thread

- Child starts with one thread
- Child’s state is a copy of parent’s
  - No sharing
Context Switch

Process A Control Block

Threads
- A1
- A2
- A3

thread state

A3 Stack

Core

Hardware Registers

R0
R1

RN

SP

B1 Stack

Process B Control Block

Threads
- B1

thread state
Context Switch

Process A Control Block

Threads
A1
A2
A3

Core

Hardware Registers

Process B Control Block

Threads
B1

A3 Stack

Saved Registers (all but SP)

B1 Stack

Control Block

A2 Stack

A1 Stack

Core

R0
R1
RN
SP

Control Block

Thread state

Thread state

A3 Stack

B1 Stack
Context Switch

Process A Control Block

Threads
- A1
- A2
- A3

A3 Stack

Core

Hardware Registers

R0
R1
RN
SP

Process B Control Block

Threads
- B1

B1 Stack

A1 Stack

...
Context Switch

Process A Control Block

Threads
A1
A2
A3

Control Block

A3 Stack

Hardware Registers

Core
R0
R1
RN
SP

thread state

Process B Control Block

Threads

B1
thread state

B1 Stack

A1 Stack

Threads

A2 Stack

A3 Stack