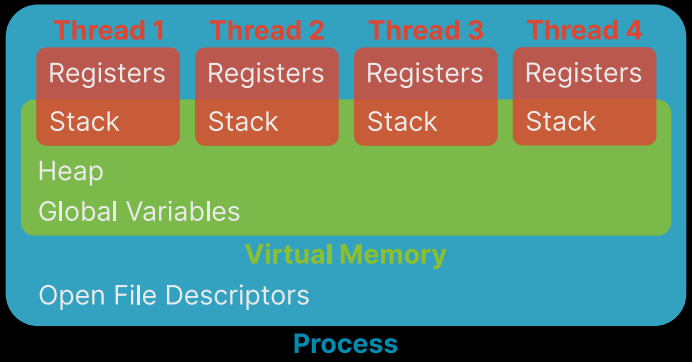
User Threads Lab Primer

2024 Fall ECE 344: Operating Systems
Jon Eyolfson

Lecture 20
2.0.0

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

A Process with 4 Threads



Remember, You Can Access These Examples!

Look at the `materials` repository, today: `lectures/20-user-threads-lab-primer`

First Example

`ucontext-example.c`

Done live, please see the recording

Second Example

`tailq-example.c`

Done live, please see the recording

ucontext Question

Global variables:

```
int i = 0;
ucontext_t uA; // Initialized to execute thread_a()
ucontext_t uB; // Initialized to execute thread b()
```

One kernel thread calls set_context(&uA), what happens?

```
void thread_a() {
    int d = 0;
    while (i < 3) {
        i++;
        printf("A: %d\n", i);
        d = 0;
        getcontext(&uA);
        if (d == 0) {
            d = 1;
            setcontext(&uB);
        }
    }
}
```

```
void thread_b() {
    int d = 1;
    while (i < 3) {
        i++;
        printf("B: %d\n", i);
        d = 1;
        getcontext(&uB);
        if (d == 1) {
            d = 0;
            setcontext(&uA);
        }
    }
}
```