Telerik Academy

6 – Processes and Threads

Marian Marinov
CEO of 1H Ltd.
mm@1h.com

Borislav Varadinov
System Administrator
bobi [at] itp.bg

Telerik Academy

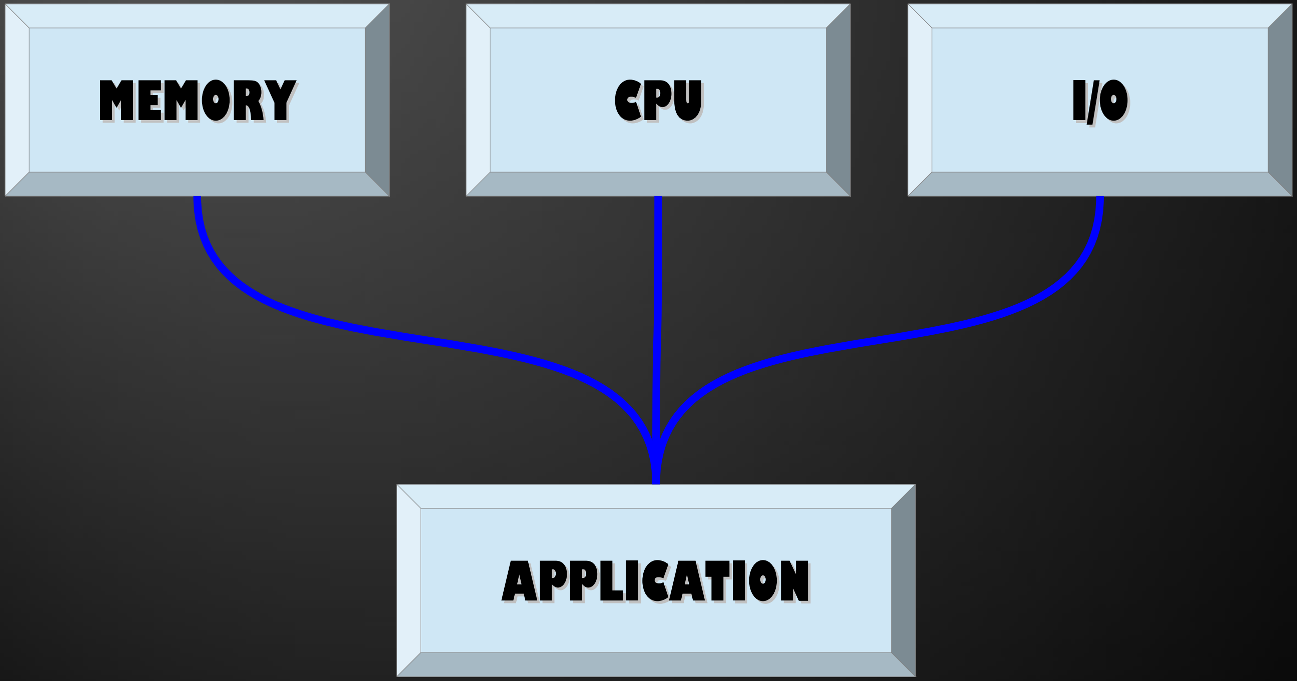
What is a Process?

Telerik Academy

What is a Process?

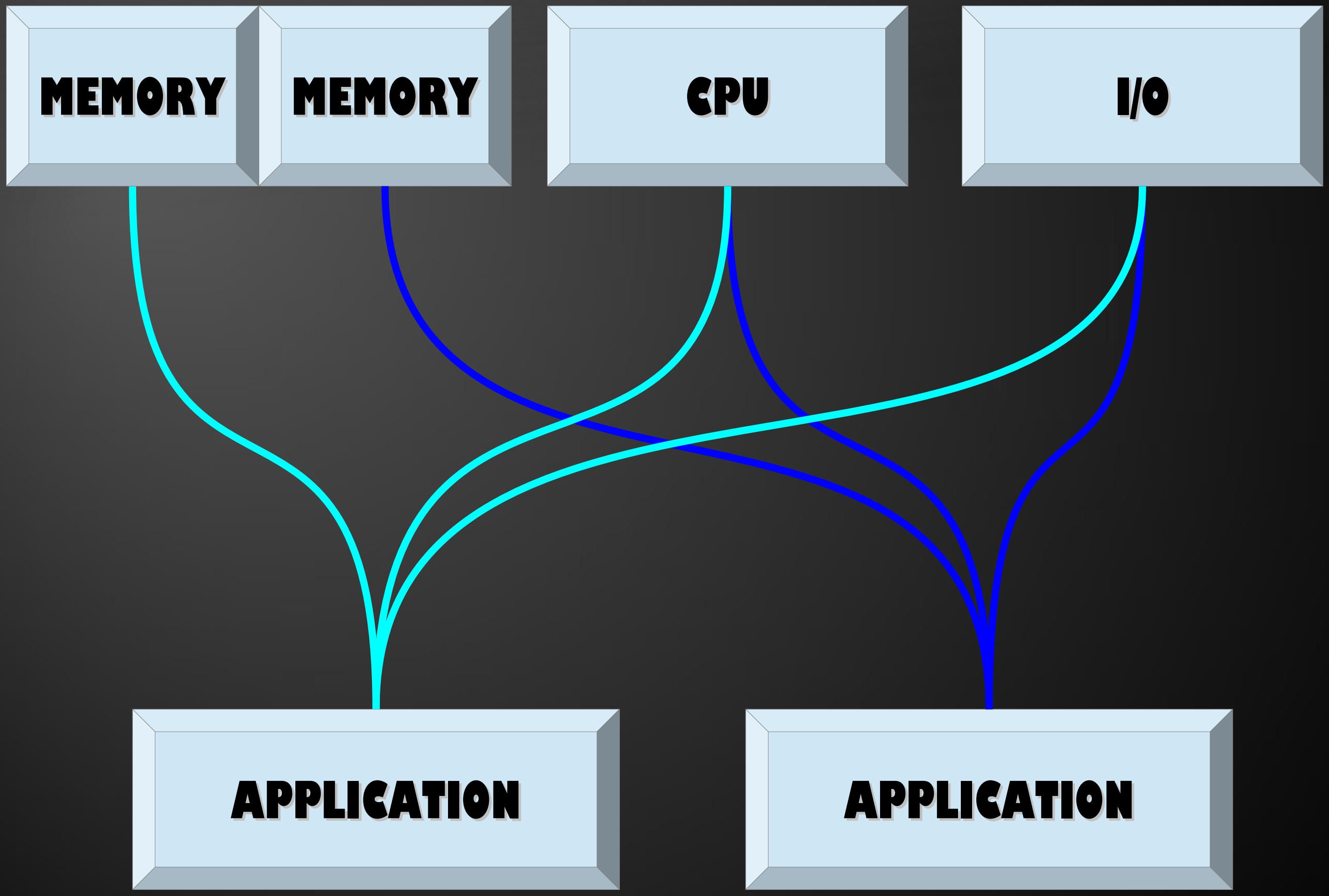
- Single process OS – Arduino
- Multy process OS – Any modern kernel

What is a Process?



Telerik Academy

Multiple Processes?



Telerik Academy

Multiple Processes?

- Segmentation Fault
- Bus Error
- Access violation

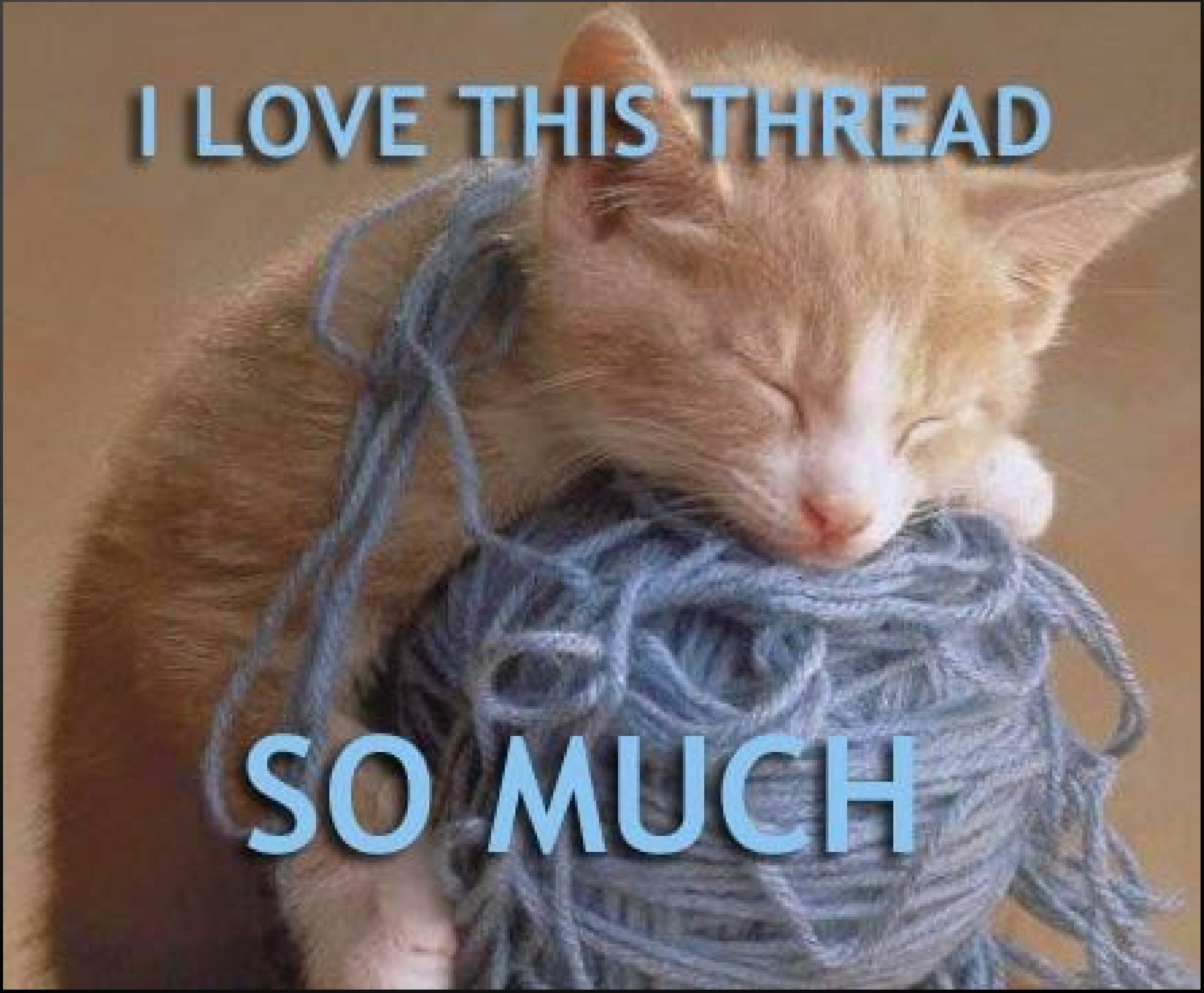
SEGFAULT/SIGSEGV

- Since Linux 3.2

  CROSS MEMORY ATTACH

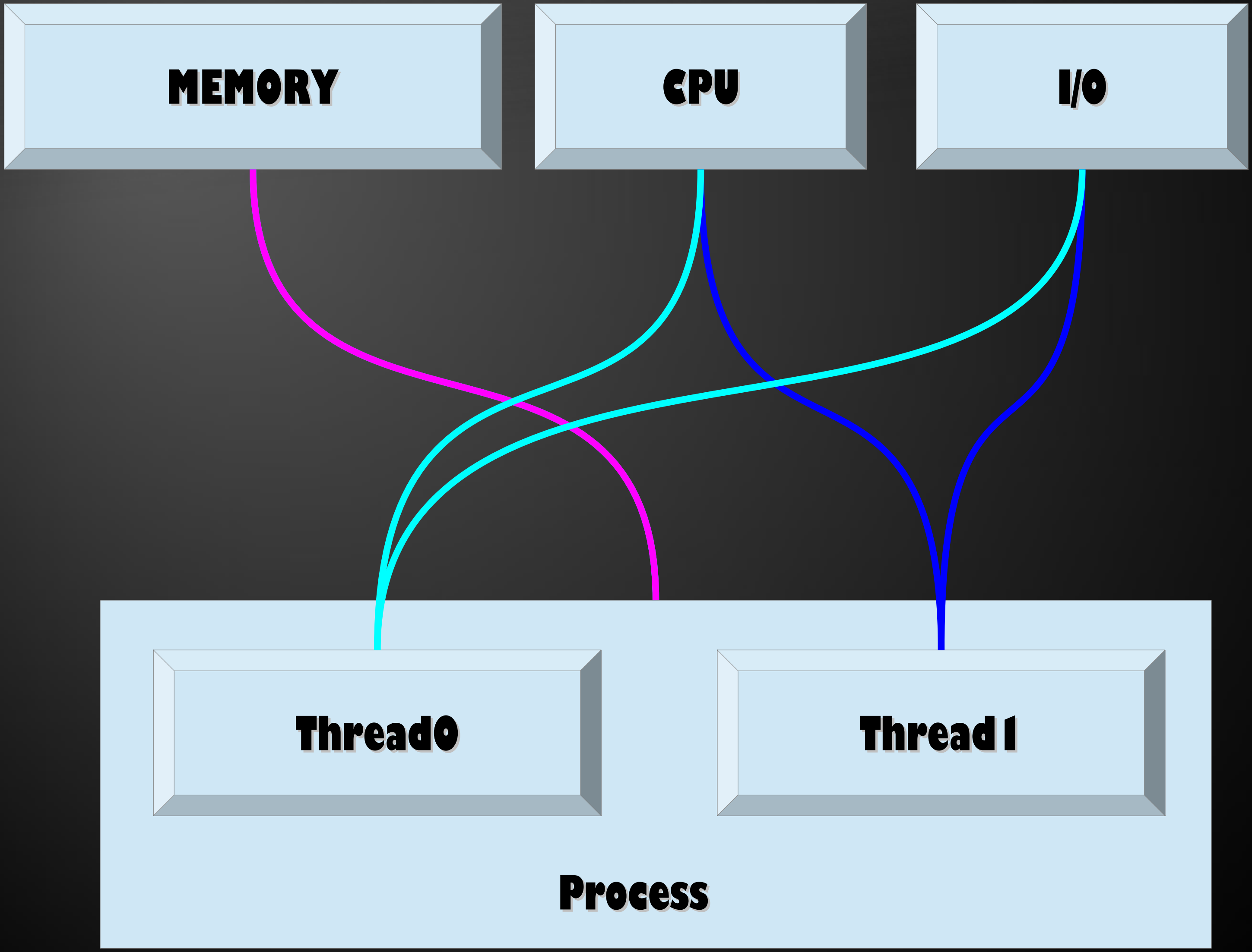
Telerik Academy

What is a Thread?



Telerik Academy

What is a Thread?



Telerik Academy

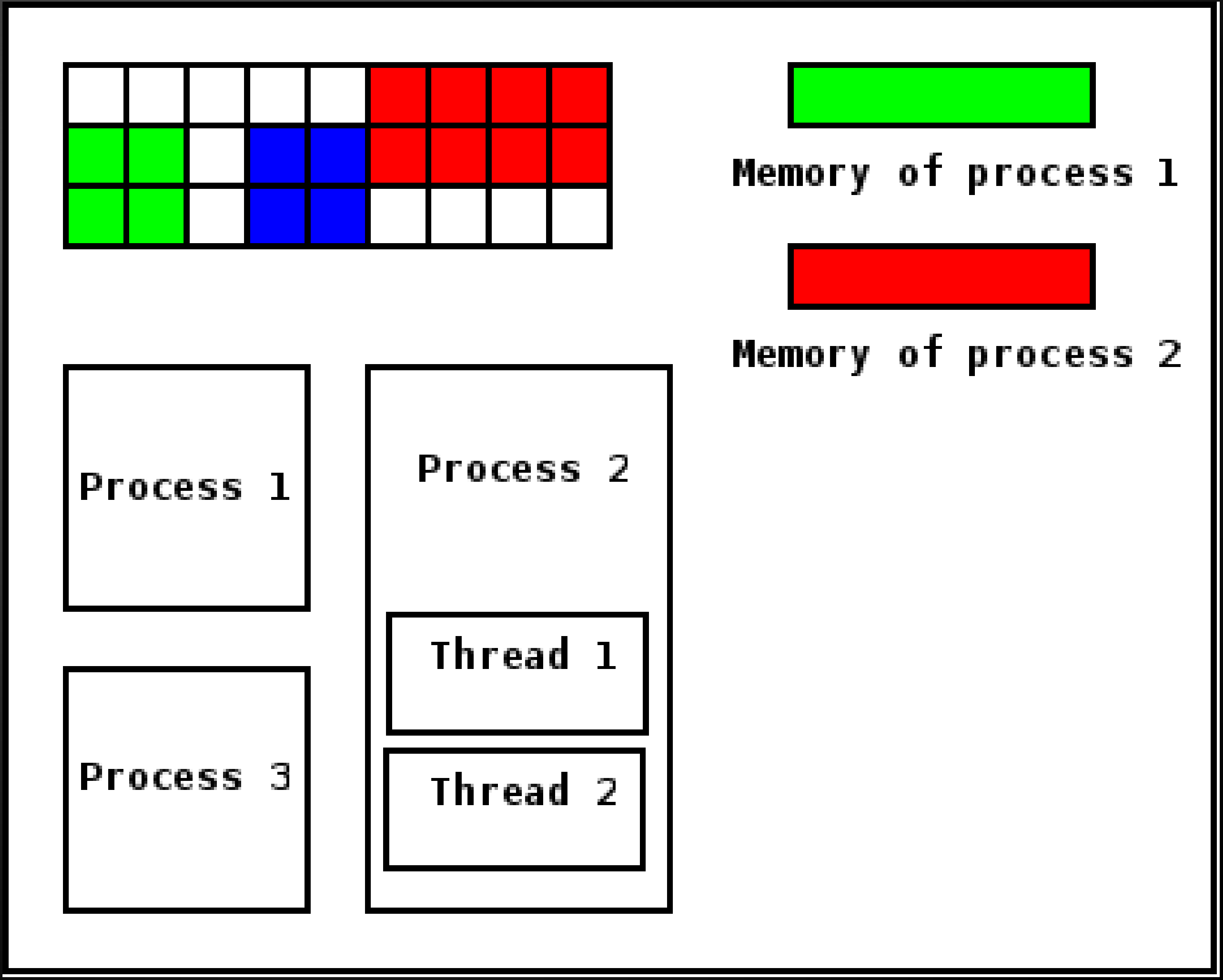
What is a Thread?

Telerik Academy

Creating a process

- FORK
  - Copy the memory of the parent
  - Inherit FD table
  - Inherit credentials
  - Inherit security
- EXEC
  - Create new memory space
  - Inherit FD table
  - Inherit credentials

What is a Thread?



Creating a process

```
#include <stdio.h>
#include <unistd.h>
int main() {
    int i;
    pid_t p;
    p = fork();
    if (p == 0) {
        for (i=0;i<=3;i++)
            printf("I'm the child(%d)!\n", p);
    } else {
        for (i=0;i<=3;i++)
            printf("My child is %d\n", p);
    }
    return 0;
}
```

Telerik Academy

Creating a process

```
hackman@terion:~$ ./f
My child is 3721
My child is 3721
My child is 3721
My child is 3721
I'm the child(0)!
I'm the child(0)!
I'm the child(0)!
I'm the child(0)!
hackman@terion:~$
```

Telerik Academy

Creating a process

```
#include <stdio.h>
#include <unistd.h>
int main() {
    int i;
    pid_t p;
    p = fork();
    if (p == 0) {
        for (i=0;i<=3;i++)
            printf("I'm the child(%d)!\n", getpid());
    } else {
        for (i=0;i<=3;i++)
            printf("My child is %d\n", p);
    }
    printf("This is not good(%d)!\n", getpid());
    return 0;
}
```

Creating a process

```
hackman@terion:~$ ./f
My child is 3762
My child is 3762
My child is 3762
My child is 3762
This is not good(3761)!
I'm the child(3762)!
I'm the child(3762)!
I'm the child(3762)!
I'm the child(3762)!
This is not good(3762)!
hackman@terion:~$
```

Telerik Academy

Creating a process

EXEC functions

```
int execl(const char *path, const char *arg, ...);
int execlp(const char *file, const char *arg, ...);
int execle(const char *path, const char *arg,
        ..., char * const envp[]);
int execv(const char *path, char *const argv[]);
int execvp(const char *file, char *const argv[]);
int execvpe(const char *file, char *const argv[],
        char *const envp[]);
```

Exec functions does not return

Telerik Academy

Creating a process

```
system() = exec('/bin/sh -c CMD')

Which means:

your process
    - exec /bin/sh
        - exec CMD
```

Types of processes

- Foreground
- Background
- Daemons
  - co-relation with terminal

Telerik Academy

Foreground processes

- It has access to the terminal's
  - STDIN – 0
  - STDOUT – 1
  - STDERR – 2
- It is directly controlled by the user
- It is connected to the terminal (text or graphic)

Telerik Academy

Background processes

- It has access only to the terminal's
  - STDOUT – 1
  - STDERR – 2
- It is NOT directly controlled by the user
- It is connected to the terminal (text or graphic)

Daemon processes

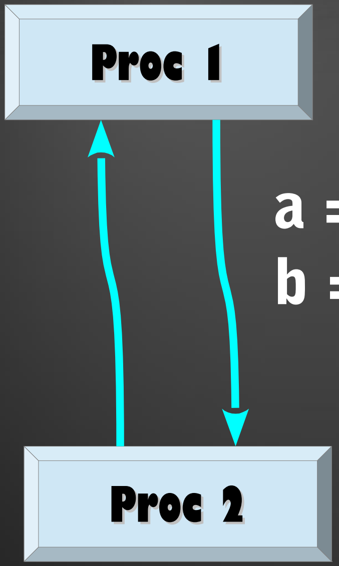
- Its STDIN/OUT/ERR are redirected to files
- It is NOT directly controlled by the user
- It is NOT connected to the terminal (text or graphic)

IPC

- FS PIPES
- Process PIPES
- Unix Domain Socket
- Shared Memory SHM (SysV/POSIX)
- Message Queues (SysV/POSIX)
- Semaphores (SysV/POSIX)
- Signals

Telerik Academy

IPC – process PIPEs

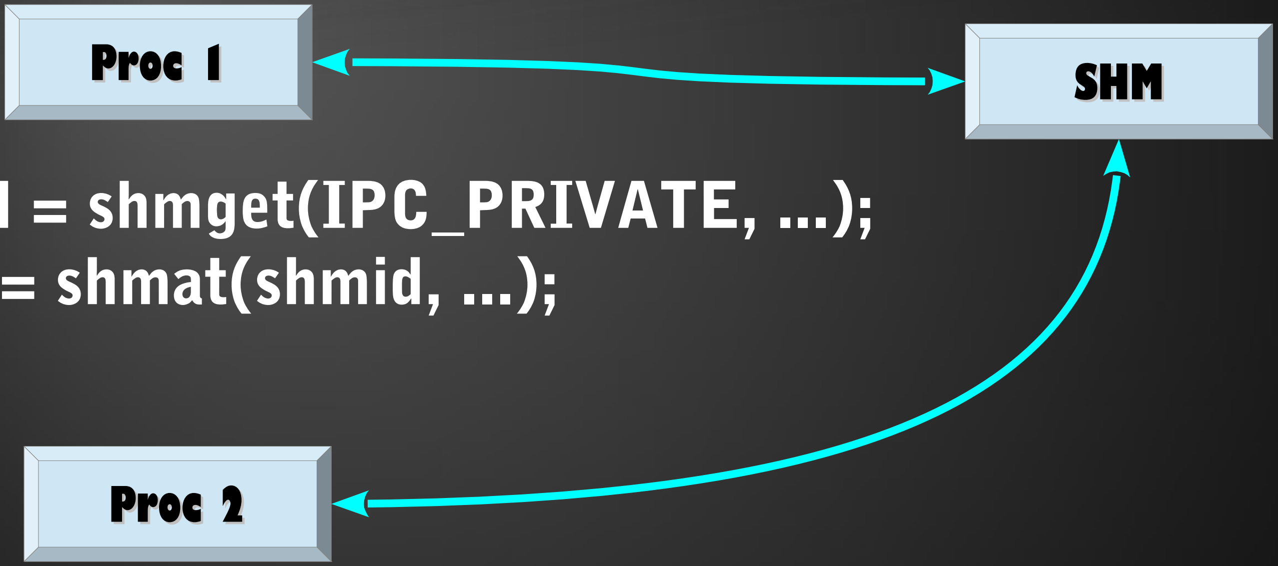


```
a = open(PIPE, "r");
b = open(PIPE, "w");
```

```
close(a); close(b);
a = open(PIPE, "w");
b = open(PIPE, "r");
```

Telerik Academy

IPC – Shared Memory

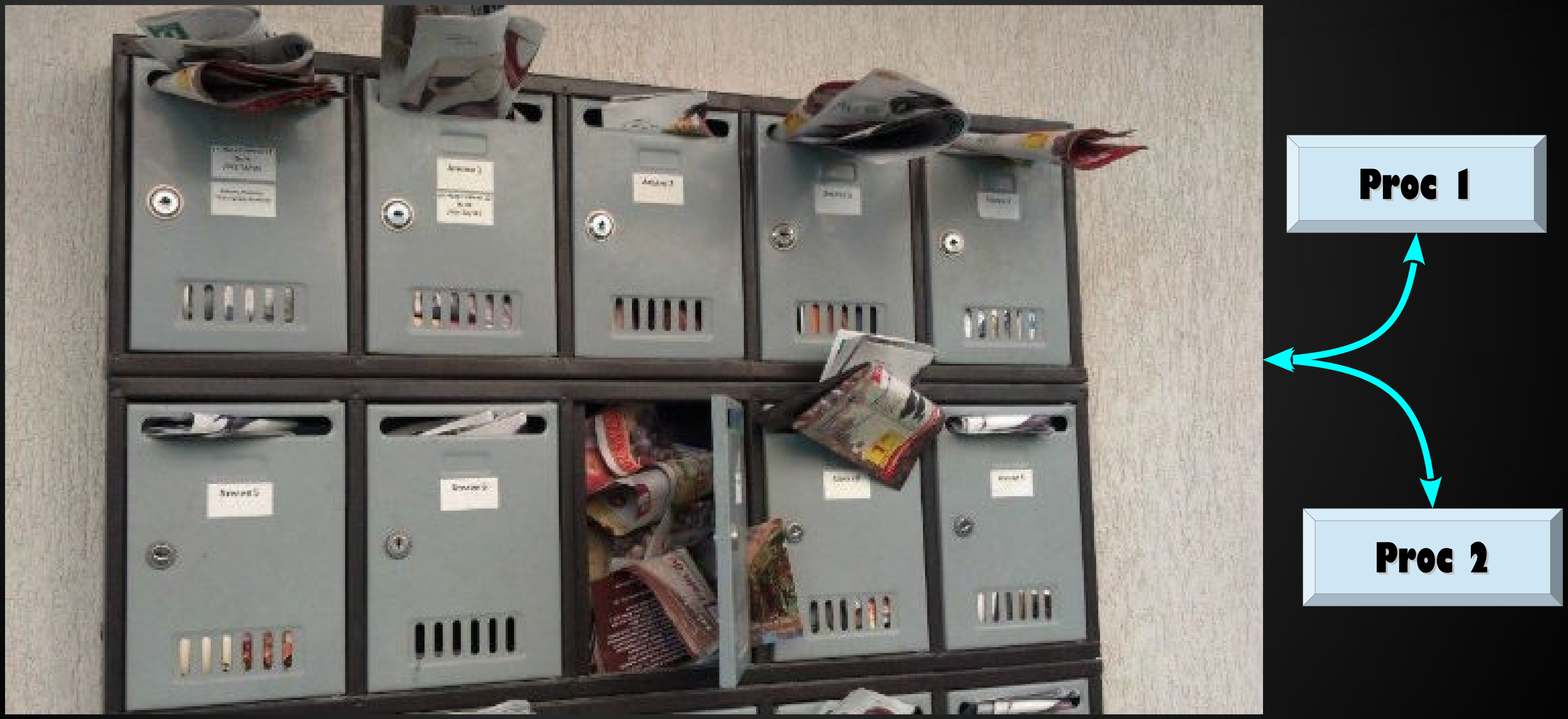


```
shmid = shmget(IPC_PRIVATE, ...);
mem = shmat(shmid, ...);
```

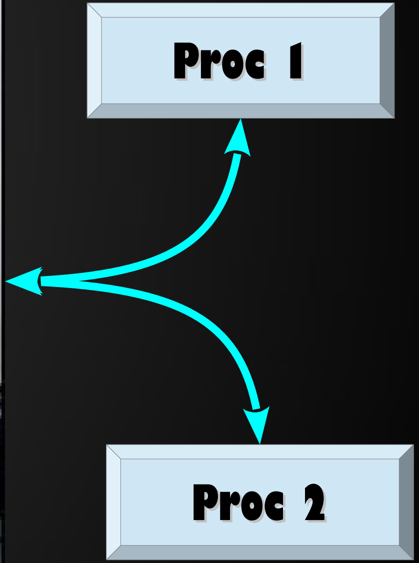
```
memset(mem, '\0', 200);
```

Telerik Academy

IPC – Message Queues



IPC – Semaphores



IPC – Signals

Proc 1

Proc 2

Telerik Academy

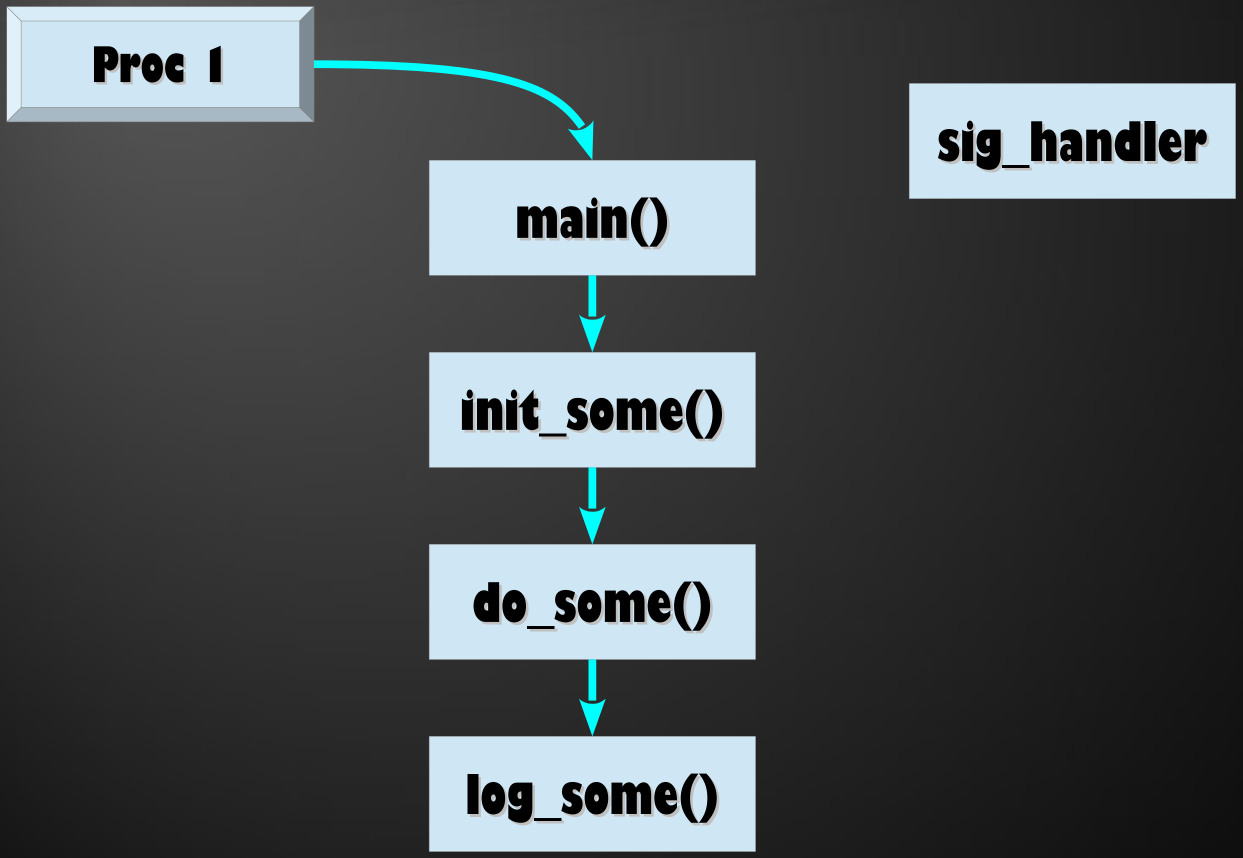
Signals - Kill

Proc 1

Proc 2

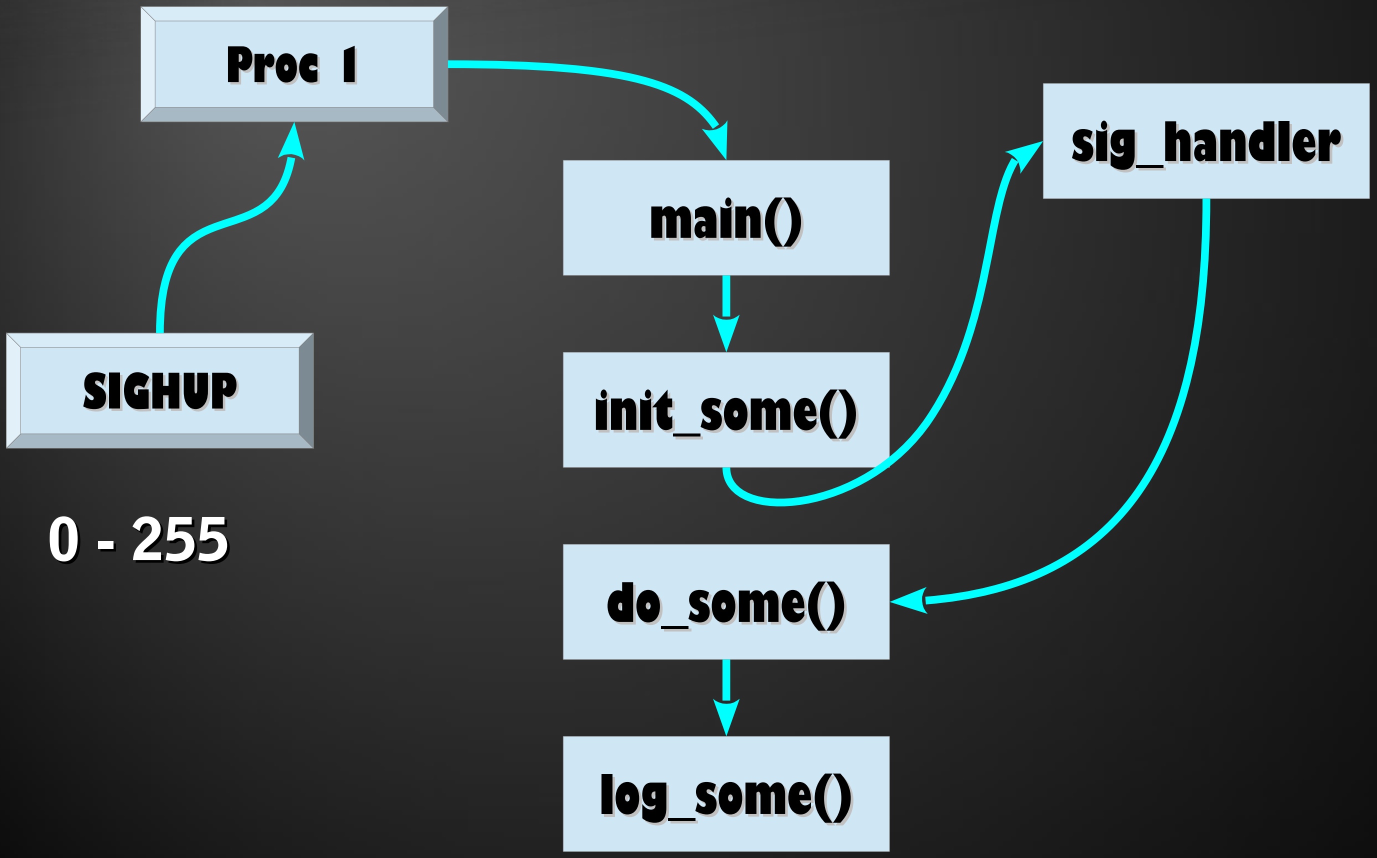
Telerik Academy

Signals – Signal handlers



Telerik Academy

Signals – Signal handlers



0 - 255

Telerik Academy

Processes and Threads

Questions?

Telerik Academy

Free Trainings @ Telerik Academy

- C# Programming @ Telerik Academy

- Telerik Software Academy

Telerik Academy

- Telerik Academy @ Facebook

- Telerik Software Academy Forums

.Net Ninja