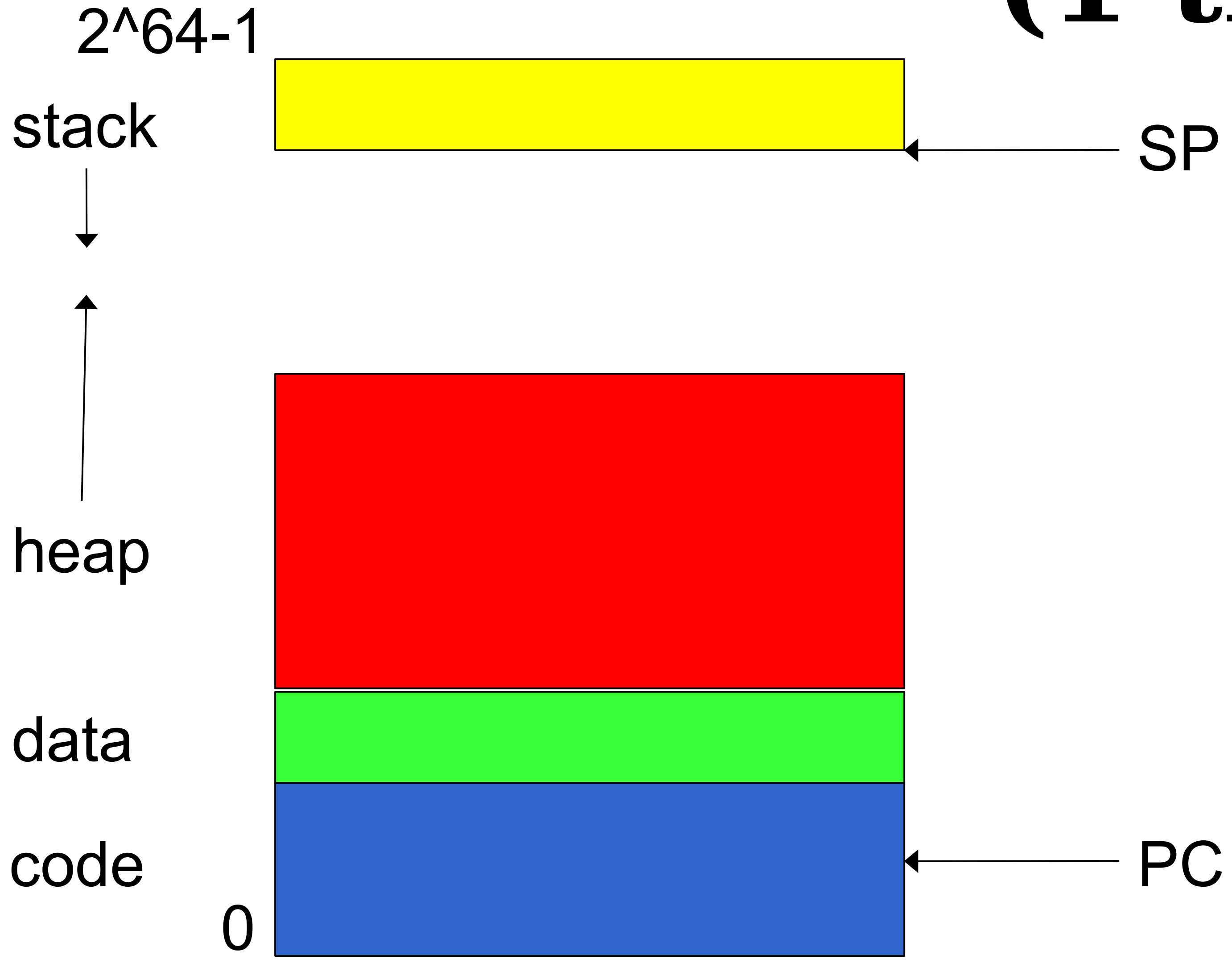
Operating Systems
CSE 511

Lecture 25

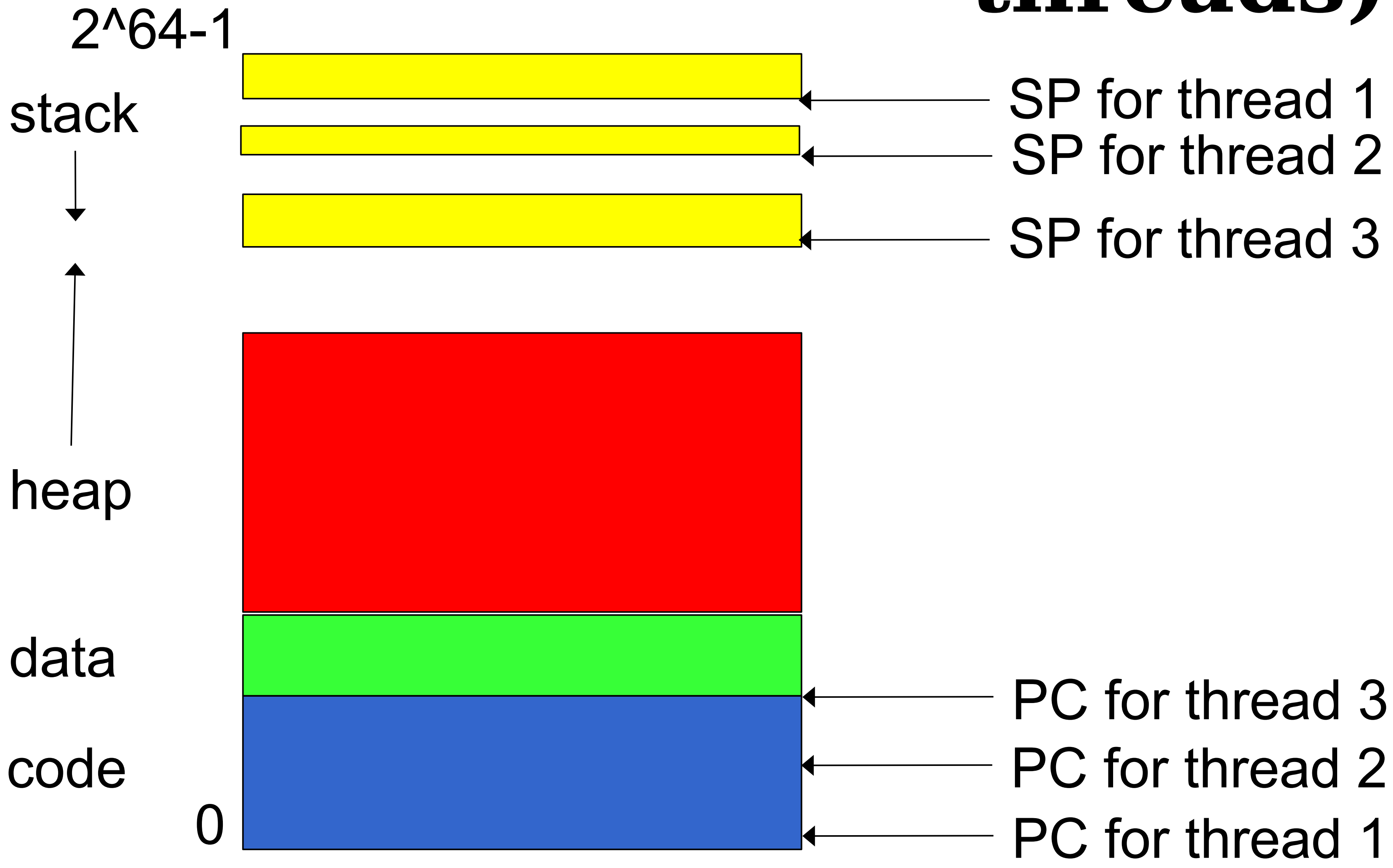
Instructor: Ruslan Nikolaev

Posix Threads

Process Address Space (1 thread)



Process Address Space (> 1 threads)

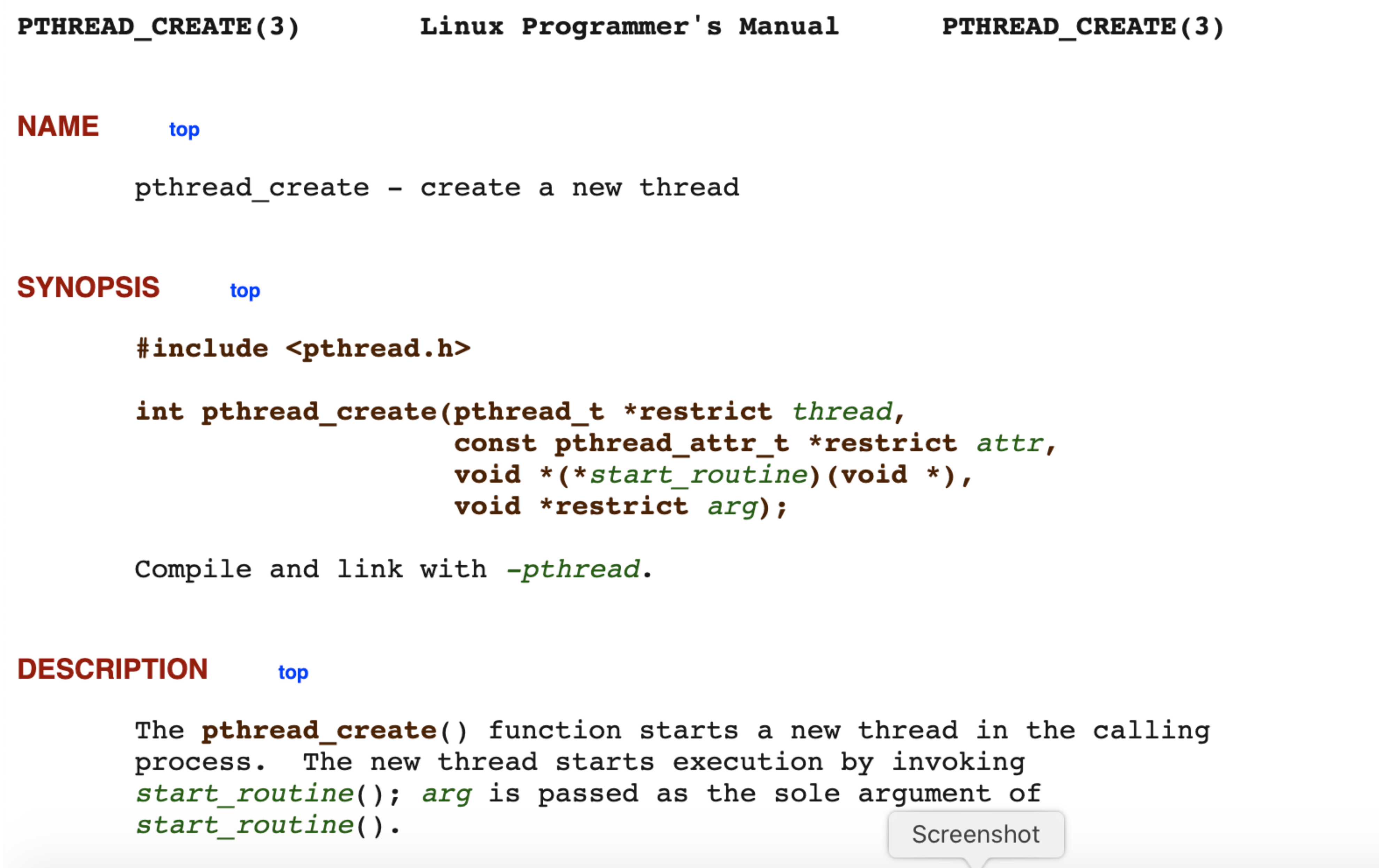


All threads in a process share the same address space

Pthreads data types of interest

- #include <pthread.h>
- pthread_t
- pthread_attr_t

Thread creation



PTHREAD_CREATE(3)   Linux Programmer's Manual   PTHREAD_CREATE(3)

NAME top

pthread_create - create a new thread

SYNOPSIS top

```
#include <pthread.h>

int pthread_create(pthread_t *restrict thread,
                   const pthread_attr_t *restrict attr,
                   void *(*start_routine)(void *),
                   void *restrict arg);
```

Compile and link with *-pthread*.

DESCRIPTION top

The **pthread_create**() function starts a new thread in the calling process. The new thread starts execution by invoking *start_routine*(); *arg* is passed as the sole argument of *start_routine*().

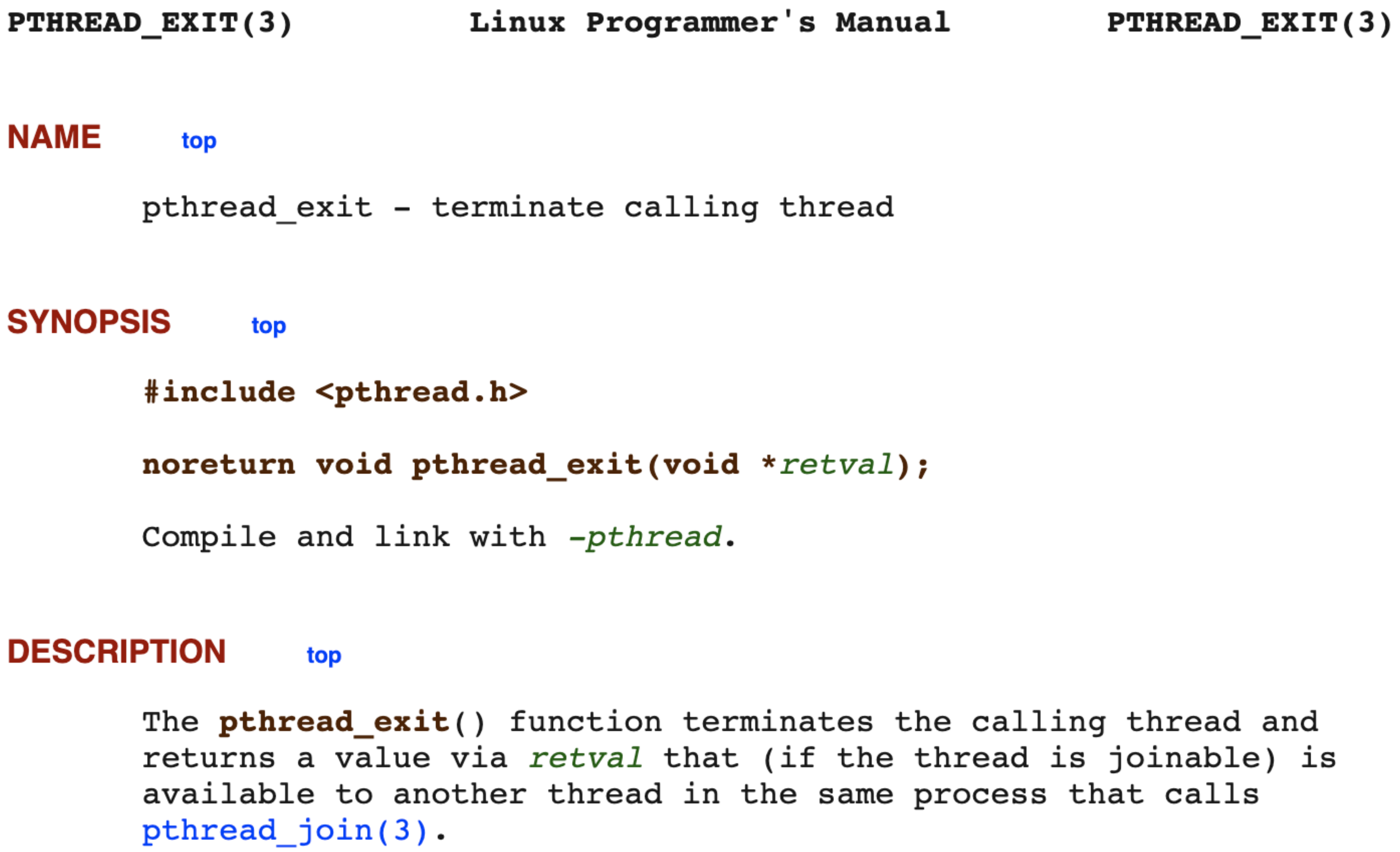
C revision

- Keyword **const**:
  - tells the compiler that this is a read-only variable
  - outcome of writing to it implementation-specific (a page may have a ‘read-only’ permission bit set)
- Keyword **restrict**:
  - keyword that can be used in pointer declarations
  - hints to the compiler that for the lifetime of the pointer, only the pointer itself or a value directly derived from it (such as pointer + 1) will be used to access the object to which it points
  - limits the effects of pointer aliasing, aiding compiler optimizations

Thread creation

- Understanding the arguments to pthread_create:
  - #1: pthread_t *restrict thread
  - #2: const pthread_attr_t *restrict attr
  - #3: void *(*start_routine) (void *)
  - #4: void *restrict arg

Thread termination



```
PTHREAD_EXIT(3)           Linux Programmer's Manual          PTHREAD_EXIT(3)
```

NAME top

```
       pthread_exit - terminate calling thread
```

SYNOPSIS top

```
       #include <pthread.h>

       noreturn void pthread_exit(void *retval);

       Compile and link with -pthread.
```

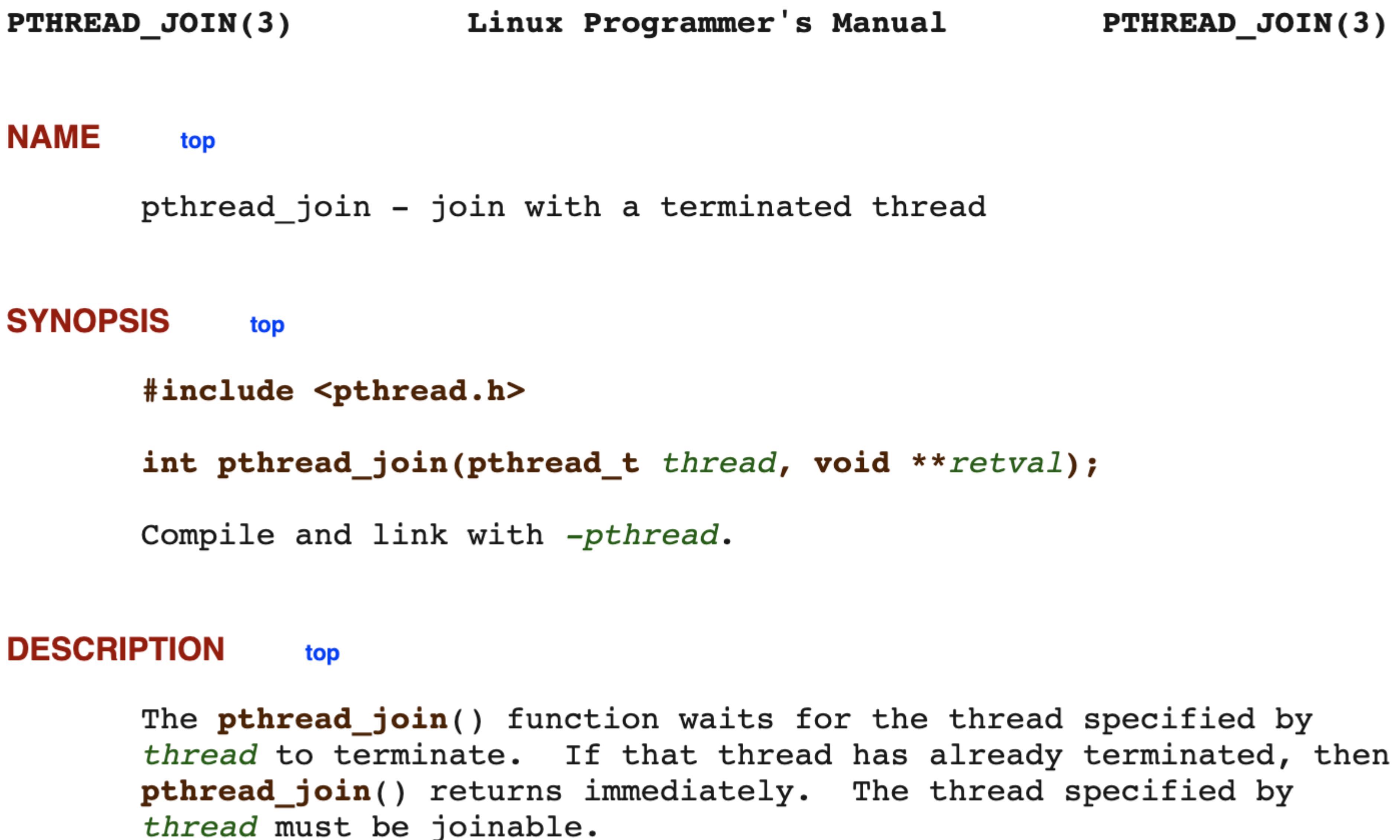
DESCRIPTION top

```
       The pthread_exit() function terminates the calling thread and
       returns a value via retval that (if the thread is joinable) is
       available to another thread in the same process that calls
       pthread_join(3).
```

Thread termination

- Keyword **noreturn**: Specifies that the function does not return to its point of invocation
- **Argument**: returns a value via “retval” that is available to another thread in the same process that calls pthread_join

Thread join



PTHREAD_JOIN(3)   Linux Programmer's Manual   PTHREAD_JOIN(3)

NAME top

pthread_join - join with a terminated thread

SYNOPSIS top

```
#include <pthread.h>

int pthread_join(pthread_t thread, void **retval);
```

Compile and link with *-pthread*.

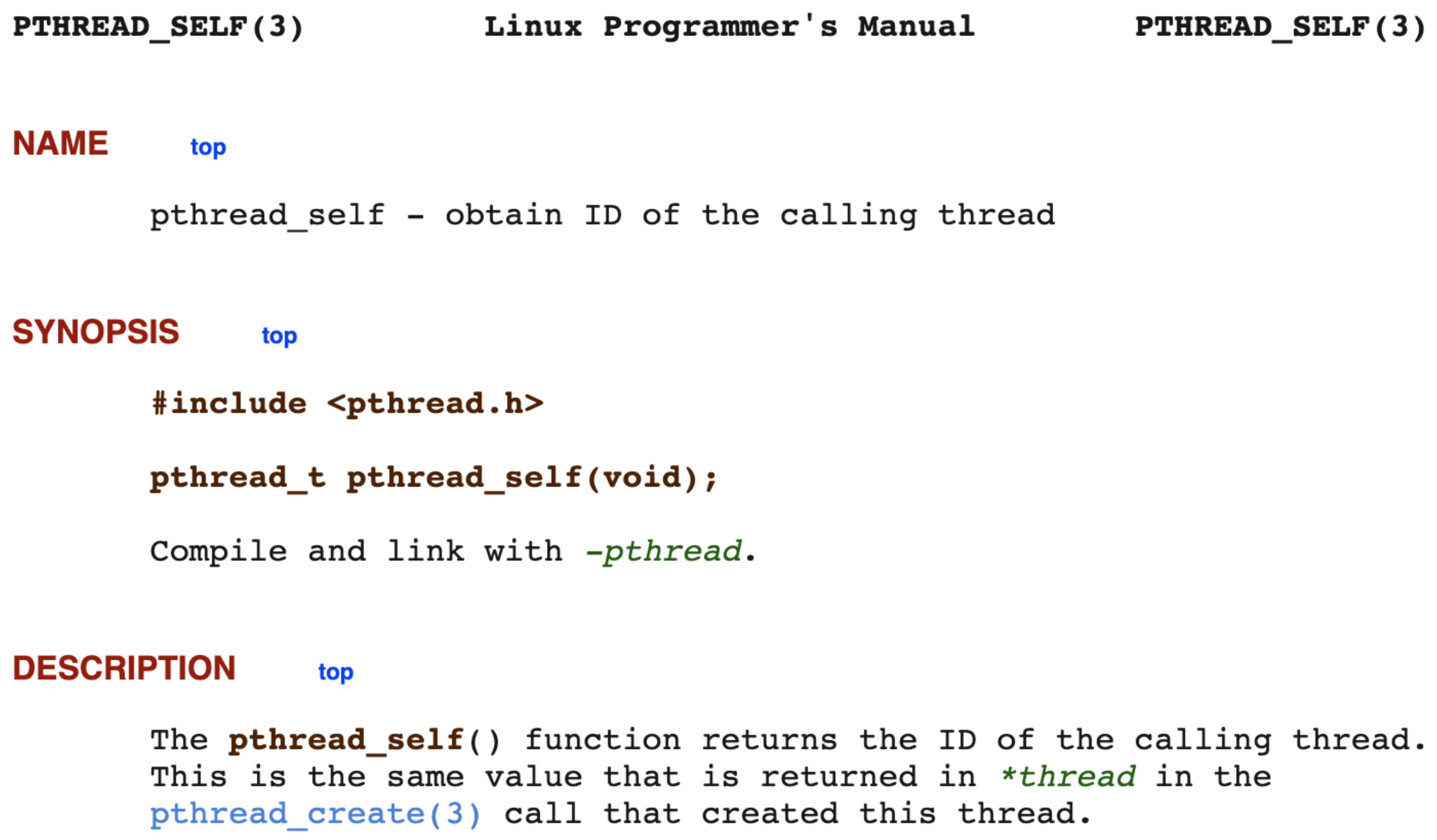
DESCRIPTION top

The **pthread_join**() function waits for the thread specified by *thread* to terminate. If that thread has already terminated, then **pthread_join**() returns immediately. The thread specified by *thread* must be joinable.

Thread join

- Argument #1: the thread whose termination we want to wait for
- Argument #2: if retval is not NULL, copies the exit status of the target thread (argument to pthread_exit) into the location pointed to by retval
- Return value: 0 on success else error number

Thread self identification



```
PTHREAD_SELF(3)          Linux Programmer's Manual          PTHREAD_SELF(3)
```

NAME top

pthread_self - obtain ID of the calling thread

SYNOPSIS top

#include <pthread.h>

pthread_t pthread_self(void);

Compile and link with *-pthread*.

DESCRIPTION top

The **pthread_self**() function returns the ID of the calling thread.
This is the same value that is returned in **thread* in the
pthread_create(3) call that created this thread.