

Fall 2015
CS3413
Lab1

Processes

1. Write a C program to fork a child process with a *fork()* and print the process id for a child and for a parent process using a C library function *getpid()*. Create an *int* variable that should be incremented by a child process and decremented by a parent process. Print the value of the variable for both a child and a parent.¹
Note: a child process id upon returning from *fork()* is 0.
2. Run a program, comment on the received ids and the values of the variable. Explain.
3. Trace system calls used by your program using *strace nameofyourfile* (-c option gives a summary of system calls only)
4. Identify a system call responsible for creation of your child process.
5. Modify your program so that your child process creates a thread with the *pthread_create* call. In a function that a thread will execute print the thread id using the *pthread_self()* call.
6. Submit your program and explanation.

Thread Information

To include the *pthread.h* library:

```
#include <pthread.h>
```

To declare a variable of type *pthread_t* :

```
pthread_t mythread;
```

The function *pthread_create()* is used to create a new thread, and a thread to terminate itself uses the function *pthread_exit()*. A thread to wait for termination of another thread uses the function *pthread_join*.

```
int pthread_create
```

```
(  
pthread_t * threadhandle, /* Thread handle returned by reference */  
pthread_attr_t *attribute, /* Special Attribute for starting thread */ keep it as NULL  
void *(*start_routine)(void *), /* Main Function which thread executes */  
void *arg /* An extra argument passed as a pointer */ keep it as NULL  
);
```

```
int pthread_join
```

```
(
```

¹ You can read more about *fork()*, *getpid()* and other C library functions by using “*man fork*”, “*man getpid*” etc.

```
pthread_t threadhandle, /* Pass threadhandle */  
void **returnvalue /* Return value is returned by ref. */      keep it as NULL  
);
```

Example:

```
pthread_create (&mythread, NULL, &myfunc, NULL)  
pthread_join(mythread, NULL)
```

To compile, add -lpthread to the linker flags:
gcc -lpthread

Selected web references:

- <http://www.digilife.be/quickreferences/QRC/LINUX%20System%20Call%20Quick%20Reference.pdf>
- http://www.gnu.org/software/libc/manual/html_node/System-Calls.html
- <http://www.kernel.org/doc/man-pages/online/pages/man2/syscall.2.html>
- <http://src.opensolaris.org/source/xref/onnv/onnv-gate/usr/src/uts/common/sys/syscall.h>