Submitter: dereky1(Yang,Derek)

HW #2

===============================start of the write-up===============================

Part1:

The difference between processes and threads is that processes are more expensive than thread. Not only their creation, but communications between processes are also expensive. Multiple threads can be run on a process, but not the other way around. Processes are essentially containers for threads, holding its storage and providing resources for threads to perform their jobs. Processes and threads share code, stack data and heap, but each thread has its own stack in memory. A situation where a new process is preferred over thread is when large software is being implemented because it provides more protection and organization for the structure of the software. Since threads can interact with other threads easily, there is less security in using threads. Threads are preferred when memory is limited or a lot of data is travelling back and forth between threads because their implementation is cheaper and communication is faster.

Part2: handle_signals.c

Submitter: dereky1(Yang,Derek)

Part3: send_signals.c

```
dereky1@odin:~/143a/hw/hw2                                    —    □    ⟩

Script started on Wed 12 Oct 2016 06:38:53 PM PDT
^[]0;dereky1@odin:~/143a/hw/hw2^G^[[?1034hdereky1@odin 18:38:53 ~/143a/hw/hw2 ^
^M
$ gcc send_signals.c -o send_signals^M
^[]0;dereky1@odin:~/143a/hw/hw2^Gdereky1@odin 18:39:11 ~/143a/hw/hw2 ^M^M
$ send_signals^M
^CSignals: 0^M
^[]0;dereky1@odin:~/143a/hw/hw2^Gdereky1@odin 18:39:26 ~/143a/hw/hw2 ^M^M
$ python^H^[[K test_send.py^M
Received SIGUSR2!^M
Received SIGUSR2!^M
Received SIGUSR2!^M
Your output should parse okay!^M
^[]0;dereky1@odin:~/143a/hw/hw2^Gdereky1@odin 18:39:41 ~/143a/hw/hw2 ^M^M
$ exit^M
exit^M

Script done on Wed 12 Oct 2016 06:39:44 PM PDT
~
~
~
~
~
                                              8,12-14        All
```

=================================end of the write-up=================================