HW Chapter 3 **Name(s):** Thomas Bailey

3.12 Including the initial parent process, how many processes are created by the program below?

```
#include <stdio.h>
#include <unistd.h>

int main()
{
     int i;

     for (i=0; i<4; i++)
          fork();

     return 0;
}
```

16

3.14 Using the program below, identify the values of `pid` and `pid1` that are output at lines A, B, C, and D. Assume that the actual pids of the parent and child are 2600 and 2603, respectively.

```
#include <sys/types.h>
#include <stdio.h>
#include <unistd.h>

int main()
{
     pid_t pid, pid1;

     pid = fork();
     if (pid < 0) {
          fprintf (stderr, "fork() failed\n");
          return(1);
     }
     else if (pid == 0)  {
          pid1 = getpid();
          printf ("pid = %d\n", pid);       // A
          printf ("pid1 = %d\n", pid1);     // B
     }
     else  {
          pid1 = getpid();
          printf ("pid = %d\n", pid);       // C
          printf ("pid1 = %d\n", pid1);     // D
          wait (NULL);
     }
     return 0;
}
```

A: 0

B: 2603

C: 2603

D: 2600

Practical:

Find `_do_fork()`, the fundamental routine for creating a new process (i.e. the main fork-routine)

- What is the purpose (give a high-level description) of `copy_process()`?

Copy process creates a new process as a copy of the old one, but does not start it. It copies the registers, and all appropriate parts of the process environment.

- Within `copy_process`, what exact code guards against `fork()` bombs?

```
retval = -EAGAIN;
if (nr_threads >= max_threads)
goto bad_fork_cleanup_count;
```