Ruth Feinstein
COSC 439
Homework 3

Code for parent program:

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

int main() {
   pid_t pid;
   int i;

   // Loop to create 5 child processes
   for (i = 0; i < 5; i++) {
      pid = fork();  // Fork a child process

      if (pid < 0) {  // Error occurred
         fprintf(stderr, "Fork Failed\n");
         return 1;
      }
      else if (pid == 0) {  // Child process
         printf("Child %d, PID: %d\n", i + 1, getpid());
         execlp("./hello", "hello", NULL);  // Replace the child process with "Hello" program
         fprintf(stderr, "Exec Failed\n");  // Only reached if exec fails
         return 1;
      }
      else {  // Parent process
         // Parent continues to next iteration of the loop
      }
   }

   // Parent waits for all child processes to complete
   for (i = 0; i < 5; i++) {
      wait(NULL);  // Wait for each child to terminate
   }

   printf("All children complete.\n");
   return 0;
}
```

Code for hello program:

```
#include <stdio.h>

int main() {
   printf("Hello from process with PID: %d\n", getpid());

   return 0;
}
```

Output:

```
Hello from process with PID: 6611
Hello from process with PID: 6612
Hello from process with PID: 6610
Hello from process with PID: 6613
Hello from process with PID: 6614
All children complete.
```