Homework 1

BRIAN ACKLEY
CSE 460

1. The code will have 8 processes at the end. The first fork() call makes a duplicate of the original process, giving us 2 processes. The second fork() ends up giving us duplicates of this which is now 4 processes. Lastly the final fork() duplicates again, which gives us 8 processes.

2. A.)

```
#include <iostream>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>
#include <stdio.h>
using namespace std;

int main()
{
   pid_t pid;
   pid = fork();
   for (int i =0; i < 10; i++)
   {
      if (pid == -1)
      {
         cout << "fork failure" << endl;
      }
      else if (pid == 0)
      {
         printf("pid in child = %d and parent = %d\n",getpid(),getppid());
      }
      else
      {
         printf("pid in parent = %d and childid= %d\n",getpid(),pid);
      }
   if (pid != 0)
   {
      int stat_val;
      pid_t child_pid
      child_pid = wait (&stat_val);
   }
   }
}
```

B.)

```
#include <iostream>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>
#include <stdio.h>
using namespace std;

int main()
{
   pid_t pid;
   pid = fork();

   printf("pid in parent = %d ;

   for (int i =0; i < 10; i++)
   {
       printf("pid in child = %d and parent = %d\n",getpid(),getppid());
   }
}
```