Identify the values of pid at lines A, B, C, and D.

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

int main()
{
        pid_t pid, forkReturn;

        /* fork a child process */
        forkReturn = fork();

        if (forkReturn < 0) { /* error occurred */
                fprintf(stderr, "Fork Failed");
                return 1;
        }
        else if (forkReturn == 0) { /* child process */
                pid = getpid();
                printf("child: return value of fork = %d", forkReturn); /* A */
                printf("\nchild: pid = %d", pid); /* B */
        }
        else { /* parent process */
                pid = getpid();
                printf("parent: return value of fork is actually the new child's pid =
                %d", forkReturn); /* C */
                printf("\nparent: pid = %d", pid); /* D */
                wait(NULL);
        }
        printf("\n");
        return 0;
}
```

Calling fork( ) system call creates a new process. This system call duplicates the current process, creating a new entry in the process table with many of the same attributes as the current process.

When fork() is called, the program divides into two separate processes.

1) The parent process is identified by a nonzero return from fork
2) The child process is identified by a zero return.

Remember, zero is not the pid of the child process. The very first process i.e., the root process has the pid number 1. No process has a pid < 1.

1) Zero is just the return code for the new (child) process, not its process id (pid).
2) The nonzero return value of the fork( ) is returned to the parent. This number happens to be the actual pid of the child.

The getpid() call gets the pid of the calling process. When called by the child process, getpid() returns the nonzero pid of the child process.

When called by the parent process, getpid() returns the nonzero pid of the parent process.

Note that the parents' pid number is < the child's pid, because the parent process was created before the child process