

**fork() system call-example**

**[17619] parent process id: 12729**

**[17619] parent process id: 12729**

**[2372] parent process id: 17619**

# fork()- program structure

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

main( int argc, char* argv[] )
{
    pid_t pid;

    if ( ( pid = fork() ) > 0 ) {
        /*parent*/
    }
    else if ( pid == 0 ) {
        /*child*/
    }
    else {
        /*cannot fork*/
    }
    exit( 0 );
}
```

# wait() system call

wait() - wait for the process whose pid reference is passed to finish executing

## SYNOPSIS

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

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

```
→pid_t wait ( int *stat_loc )
```

The unsigned decimal integer process ID for which to wait.

## RETURN VALUE

success- child pid

failure- -1 and errno is set