Venkat Prathipati
Shreya Kumar
Section #23

Lab 1 Report

Files Changed for exit system call signature:

1.) user.h
2.) defs.h
3.) sysproc.c
4.) proc.c

Files Changed for exit calls:

1.) cat.c
2.) echo.c
3.) forktest.c
4.) grep.c
5.) init.c
6.) kill.c
7.) ln.c
8.) ls.c
9.) mkdir.c
10.) proc.c
11.) rm.c
12.) sh.c
13.) stressfs.c
14.) sysproc.c
15.) trap.c
16.) usertests.c
17.) wc.c
18.) zombies.c

Files Changed for wait system call signature:

1.) user.h
2.) defs.h
3.) sysproc.c
4.) proc.c

Files Changed for wait calls:

1.) forktest.c
2.) init.c
3.) Proc.c
4.) sh.c

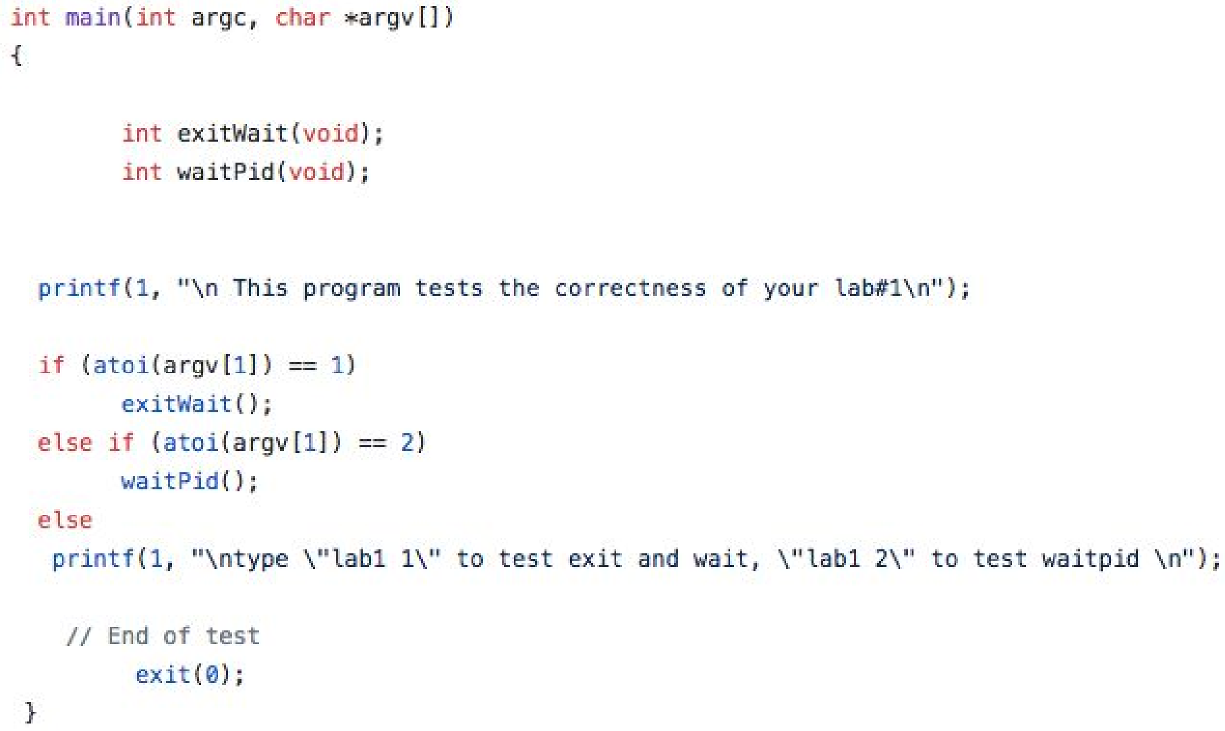
5.) stressfs.c
6.) sysproc.c
7.) usertests.c

Files Changed for waitpid system call signature:
1.) user.h
2.) defs.h
3.) sysproc.c
4.) proc.c

Waitpid calls:
1.) usys.S - added to syscall list
2.) syscall.c - added to vector table
3.) syscall.h

How to test the correctness of your work (you can use snapshots, put parts of code .. etc)

-We used a lab1 test file to test our work.



```
int main(int argc, char *argv[])
{

        int exitWait(void);
        int waitPid(void);


  printf(1, "\n This program tests the correctness of your lab#1\n");

  if (atoi(argv[1]) == 1)
        exitWait();
  else if (atoi(argv[1]) == 2)
        waitPid();
  else
   printf(1, "\ntype \"lab1 1\" to test exit and wait, \"lab1 2\" to test waitpid \n");

    // End of test
         exit(0);
 }
```

-We also created a waitpid test file which tests the hello world program.

- We ran it using
    -make qemu-nox
    -./lab1

To test exit and wait type “lab1 1”.

Venkat Prathipati
Shreya Kumar
Section #23

To test waitpid type “lab1 2”.