Derek Lee, Shine Li Computer Operating Systems Fall 2020

Professor Hakner PSET #3

Problem 1

a) i will be in the DATA segment. It is a static variable initialized to a non-zero value.

b) The first call to F1() prints 10 and increments i_parent, so i_parent=11. The fork() command creates a child process. The child process continues where the parent process was, so it enters the if statement because fork() returns 0 to the child process. However, the parent process does not enter the if statement because fork() returns a positive value representing the PID of the child. The child process prints out 11 and increments i_child, so i_child=12. Then, it prints out 12 and increments i_child. The parent process prints out 11 and increments i_parent. The child does not necessarily go before or after the parent.

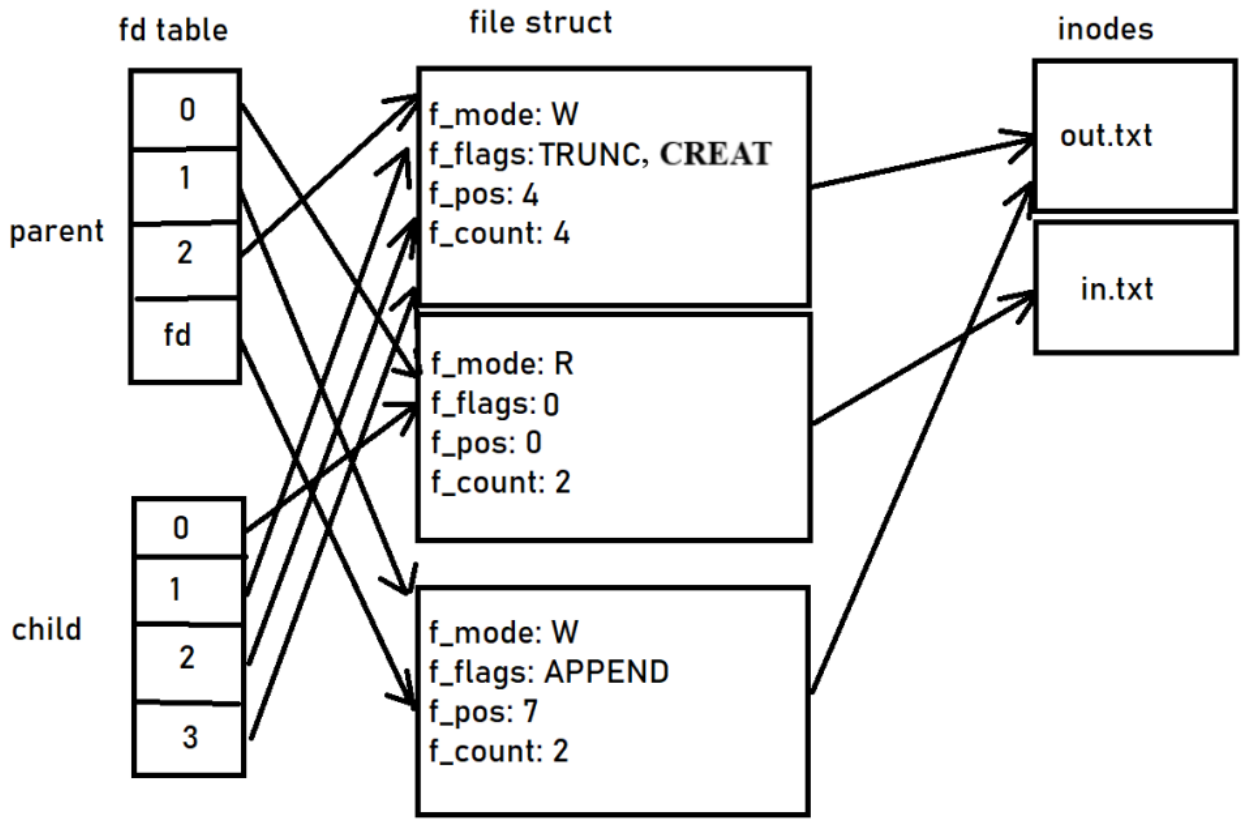
   Parent: 10 & 11

   Child: 11 & 12

c) No, because there is a race condition. It is undefined whether the parent process or the child process will execute first, or if they will execute simultaneously.

d) It prints 255 to STDOUT. echo $? prints the exit status of the last command, which is the return value of the parent process. The return value of the parent process operates on the parent copy of the integer ws, whose value was set in the wait(&ws) system call made earlier. This wait() syscall sets the integer pointed to in the address to have the same value as the child process' return value. The child process uses its own local int ws, which is still -1. The child process evaluates (-1>>8)&255 = 255, and returns that value, which is then stored in the parent's copy of ws via wait(). Since return values are stored in the first 8 bits of the integer, the parent's ws becomes 0b1111111100000000 = 65280. Finally, the return value of the parent process is evaluated: (65280>>8)&255 = 255. The wait() doesn't affect the child process.

Problem 2


fd table
file struct
inodes
parent
0
1
2
fd
child
0
1
2
3
f_mode: W
f_flags: TRUNC, CREAT
f_pos: 4
f_count: 4
f_mode: R
f_flags: 0
f_pos: 0
f_count: 2
f_mode: W
f_flags: APPEND
f_pos: 7
f_count: 2
out.txt
in.txt