Jacob Stone

CIS 452

LAB 2

9/9/21

1. **Open other man pages from section 2 or 3 and look for more error codes and explain the meaning of each code.**

EILSEQ is the error code to show that the input byte sequence does not form a valid character

ENOMEM is error message for out of memory

1. **Open the man page of sqrt(), what is the symbolic error code that may be set into errno by this math function?**

The Symbolic error is because the input value is less than -0.

1. **By comparing the manual pages of puts() and printf() (both are from section list *at least* two differences between the two library functions.**

-Printf is used to output and format data (regardless) while puts output of characters and strings

-printf can have modifiers or field characters while puts will only print the string/char.

1. **How many lines are printed by the program?**
   1. **How many lines by the parent process?**

-two lines are printed by the parent

* 1. **How many lines by the child process?**
* One line is printed by the child

1. **Describe what is happening to produce the answer observed for the above question**

The parent will print the first statement then fork and copy itself into the child, while both of them are running, they both print out the second print statement, therefore there are three statements in all.

1. **Create a diagram illustrating how Sample Program 2 executes (i.e. give a timeline diagram, similar to used in lecture).**

1. **In the context of our classroom discussions on process state, process operations, process scheduling, etc., describe what you observed and try explain what is happening to produce the observed results. This is primarily an experiment; look for apparent anomalies and try to explain them based on classroom discussion of process operations.**

What has been observed is the parent will fork twice in this program and then start to print out its other commands. While the 1st child is forking for the second time, the 2nd child is starting print since it has no forks. The 1st child can then start to print and the last child is ready to print at this time too but is still multiple iterations behind the parent and other children.

1. **What line of code did you insert for the wait() system call?**

It is inserted directly before the printf statement since it needs to be terminated .

1. **Who prints first, the child or the parent? Why?**

The child prints since the parent is waiting

1. **What two values are printed out by the parent in Sample Program 3 (No, not the actual numbers, but what they mean.) In other words, describe the interaction between the exit() function and the wait() system call. You may want to experiment by changing the value in the exit() call to better understand the interaction.**

The childs PID is printed out as the first print statement then the first number printed in the parent is the 0 from the pid given to the child at the beginning of the program. lastly the exit return value is printed as the last value.

1. **Do you see the output "Just checking" or perror() output after the 3-month calendar is printed? Why or why not? Explain what happened.**
2. **Make the following changes to Sample 4:**
   1. **Replace /usr/bin/cal with /usr/bin/calculus (or any non-existent file)**
   2. **Recompile and rerun the program. What output do you see now?**
   3. **Explain what just happened, compare this to your observation at the previous question.**
3. **Make the following changes to Sample 4**
   1. **Comment out the execl() call and add the following execlp() call:**
   2. **Recompile and rerun**
4. **Make the following changes to Sample 4**
   1. **Comment out the execlp() call and add the following execvp() call:**
   2. **Recompile and rerun**

1. **Explain how the second argument passed to execvp() is used?**