

Assignment 2

Problem Statement -

The demonstration of fork along with zombie and orphan states

- a) Implement C program in which main pg accepts the integers to be sorted. Parent process sorts the integers using sorting algo and wait for child process using WAIT system call to sort the integers using any sorting algo.
- b) Implement C program in which main program accepts an integer array. Parent process sorts an integer & passes the sorted array to child process through cmdline EXECVE system call.

Theory -

Fork call - used for creating a new process, which is called process, which runs concurrently with the process that makes the `fork()` call.

Execve call - executes the program pointed to by `pathname`.

Wait call - blocks the calling process until one of its child process exits or a signal is received. After child process after wait system call.

Algorithm - Part 1 -

- Accept the integer array to be sorted.
- call forkⁿ
- In child process call execve to binary search.
- In parent process display all sorted array.
- Use switch case to first sort, display, fork and call binary search program.
- Exit.

Part 2 -

- 1) ~~#~~ Find the elements of array from char* argv[1].
- 2) Convert string to int using atoi()
- 3) call ~~bubble sort~~ function to look for given element by use binary search
- 4) Print the found element.
- 5) Exit.

Conclusion -

Successful implementation of FORK, EXECVE and WAIT system call in c along with zombie & orphan state.