Multiprocessing_Assignments_1

1. Write a multiprocessing application to perform the following.
    a. Application to receive an input file name as command line argument.
    b. Store the filename in a buffer.
    c. Create a child using fork().
    d. Display the process id of both parent and child
    e. Let child prompt and read 2 lines from user and write to input file and exit.
    f. Let parent after fork() do the following
        - wait for child to complete.
        - open input file name, read and display the contents
    g. Add a function below and call it from parent after wait()
    /*Using WIFEXITED macro check if the child has exited and in such case extract exit code and display, Else, use WIFSIGNALLED macro to check if the child was terminated using signal (say using SIGINT) then the retrieve and display the signal number which caused the exit */
    process_display_exit_code(int exitstatus)

→ Ans: Refer directory Question01.

---

2. Refer the code snippet below and answer the queries.

```
    int g_value = 10; /* global variable */
    int main()
    {
        int pid;
        int l_value = 5;
        printf("Writing a sample code\n");
        pid = fork(); /* fork() returns 0 to child process and process id of child to parent
    process*/
        if(0 == pid)
        {
            printf("For child Local variable value=%d\n and global variable
value=%d\n",l_value,g_value);
        exit(0);
        }
        else
        {
            printf("For parent Local variable value=%d\n and global variable
value=%d\n",l_value,g_value);
        }
        printf("Code common for both parent process and child process\n");
        return 0;
    }
```

---

a. What will be the output of parent process and child processes?
   Output of Parent Process:
   Local variable value=5
   and global variable value=10
   Output of Child Process:
   Local variable value=5
   and global variable value=10

b. Find out whether the value of local variable and global variable value will be same for both parent process and child process.
   - Yes, the value of local variable and global variable value will be the same for both parent process and child process.

c. Will the order of execution be same always or could be different? Will it impact the output?
   - The order of execution could be different.
   - No, order of execution won't impact the output.

d. Why the first printf() statement will be executed only by parent process and not by child process?
   - Using wait() in the parent process can execute the child process at first.
   - But in this program, wait() function is not used, so the parent process is executed first not by child process.

Assignment No: 02
Done By: Rajesh Kannan M
Employee ID: 46252660