Programming Assignment 1 Report

Anthony Carvalho

Thales Moraes

This program will read numbers from a separate txt file and use those numbers in 4 different child processes, which are created by using “fork()”. Each child will then find all prime numbers from 0 up to a given number from the txt file. When the process is completed, each child will display a message stating the task is done and how long it took to finish the task. After all child processes finish their tasks, the parent process then displays the final message and, as an extra, we created a leaderboard of the top 3 fastest child processes.

With this assignment, we were able to learn how child processes are created and how related functions work, such as fork() and wait() . The biggest challenges were creating and deleting a temp file to store the times for each process, as well as figuring out the formatting for the leaderboard.

Reads file by X lines, get numbers

Start Threads by X children

Child 1 | Child 2 | Child 3

Performs Operation | Performs Operation | Performs Operation

Save results on temp file | Save results on temp file | Save results on temp file

```c
// Save child info in a file
FILE *file = fopen("child_info.txt", "a");
if (file != NULL) {
    fprintf(file, "%d %.6f\n", child_num, elapsed);
```

Numbers.txt to be read

numbers.txt

File Edit View

```
15
17
23
28
31
```

Starts threads

```
Creating 4 child processes.
```

Operations Consists of finding prime numbers ⚙️

```
Child 1 (PID: 1523808) finding primes up to 15:
2 3 5 7 11 13
Child 2 (PID: 1523809) finding primes up to 17:
2 3 5 7 11 13 17
Child 3 (PID: 1523810) finding primes up to 23:
2 3 5 7 11 13 17 19 23
Child 4 (PID: 1523811) finding primes up to 28:
2 3 5 7 11 13 17 19 23
```

```c
// Function to check if a number is prime
int is_prime(int num) {
    if (num <= 1) return 0;
    for (int i = 2; i * i <= num; i++) {
        if (num % i == 0) return 0;
    }
    return 1;
}
```

Read and display results and leaderboard, delete temp file ✅

```c
// Display the leaderboard
display_leaderboard();

// Clean up
remove("child_info.txt");
```

Display leaderboard ✅

```
Winner: Child 1 in 0.000036 seconds

Child Leaderboard Podium:

                    Child 1 (Winner)
                    _____________
       Child 4     |    1st      |
      ____________ |             |  Child 3
     |    2nd     |             |_________
     |            |                 3rd     |
     |            |_____________________    |
Child 1 took (0.000036 s)
Child 4 took (0.000037 s)
Child 3 took (0.000037 s)
Child 2 took (0.000043 s)
acarv048@ocelot:~/COP4610 293%
```

```c
// Function to display the leaderboard
void display_leaderboard() {
    FILE *file = fopen("child_info.txt", "r");
```