Logotipo

Descripción generada automáticamente con confianza media

**Universidad Autónoma de Guadalajara**

**C programming Language**

**STUDENT:** Erick Mathew García Sánchez

**CAREER:** Software Engineering and Data Mining

**COURSE:** Operating Systems

**REGISTRATION NUMBER:** 2818812

**TEACHER:** Agustín Villarreal

**DATE:** 13 – 09 – 25

1. A screen shot of a computer code

   AI-generated content may be incorrect.Bookstores:

1

2

3

4

5

1. Functions such as fork(), execvp(), getpid()
2. For wait() and macros related to process states
3. Time-related functions
4. String manipulation functions, not used directly here
5. For gettimeofday() functions and more accurate
6. A screen shot of a computer code

   AI-generated content may be incorrect.Command definition

An Array of arrays of character pointers is declared; each sub-array contains a command and its arguments to execute. The las element is always NULL, required by execvp to mark the end of arguments.

c) variables to control processes and times

A black background with white text

AI-generated content may be incorrect.

num\_commands: Number of commands to be executed (in this case 5).

child\_pids: Array to store the process identifiers (PIDs) of the children.

start\_total and end\_total: variables to measure the total execution time using gettimeofday().

A screen shot of a computer

AI-generated content may be incorrect.d) Measure total time at start and display information

gettimeofday obtains the current time with microsecond precision. A header and the parent process PID are printed.

A screen shot of a computer

AI-generated content may be incorrect.e) create child processes with fork()

A loop is created to generate a child for each command. fork() duplicates the current process, returning:

* < 0 if it fails
* 0 in the child process
* PID of the child in the parent process

f) Error handling and execution in child processes

A screen shot of a computer code

AI-generated content may be incorrect.

If fork() fails, an error is output and the program terminates. In the child process (rc == 0), an output is generated indicating that a command is being executed. The command is executed with execvp(). If execvp() fails, an error message is output and the child process terminates to prevent further execution of faulty code.

g) store child PIDs in the parent process

A close-up of words

AI-generated content may be incorrect.

in the parent, store the child’s PID in the array. Print confirmation of the child’s creation.

A screen shot of a computer code

AI-generated content may be incorrect.h) wait for the children to finish and obtain their status

Use a loop to wait for all children to finish with wait(). wait() returns the PID of the finished child ant its status. If wait() fails (-1), an error is reported and the program continues.

i) Find the index of the finished child

A screen shot of a computer program

AI-generated content may be incorrect.

Search the child\_pids array to find out which child finished (by its PID). This allows you to relate the PID to the corresponding index and command.

j) Display execution result for each child

A screenshot of a computer

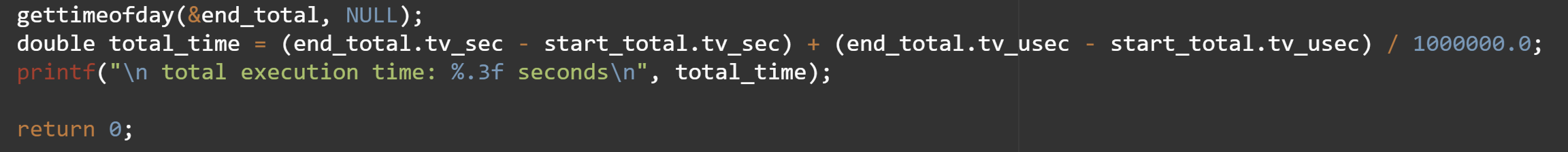
AI-generated content may be incorrect.

WIFEXITED(status): checks whether the child terminated normally.

WEXITSTATUS(status): obtains the exit code of the child.

If the child terminated abnormally, this is reported.

k) Measure total execution time and display it



The time is captured when all child processes are finished. The difference in seconds is calculated, considering seconds and microseconds. The total execution time of the parent program is printed.