БГУИР

Кафедра ЭВМ

Отчет по лабораторной работе № 3

Тема: «Взаимодействие и синхронизация процессов»

Выполнил:

студент группы 150502 Альхимович Н.Г.

Проверил:

старший преподаватель Поденок Л.П.

Минск

2023

**1 ЦЕЛЬ РАБОТЫ**

Изучить принципы синхронизации процессов с помощью сигналов и обработку сигналов таймера.

**2 ПОСТАНОВКА ЗАДАЧИ**

Разработать две программы ‒ parent и child.

По нажатию клавиши «+» родительский процесс (P) порождает дочерний процесс (C\_k) и сообщает об этом.

По нажатию клавиши «-» P удаляет последний порожденный C\_k, сообщает об этом и о количестве оставшихся.

При вводе символа «l» выводится перечень родительских и дочерних процессов. При вводе символа «k» P удаляет все C\_k и сообщает об этом.

При вводе символа «s» P запрещает всем C\_k выводить статистику (см. ниже).

При вводе символа «g» P разрешает всем C\_k выводить статистику.

При вводе символов «s<num>» P запрещает C\_<num> выводить статистику.

При вводе символов «g<num>» P разрешает C\_<num> выводить статистику.

При вводе символов «p<num>» P запрещает всем C\_k вывод и запрашивает C\_<num> вывести свою статистику. По истечению заданного времени (5 с, например), если не введен символ «g», разрешает всем C\_k снова выводить статистику.

По нажатию клавиши «q» P удаляет все C\_k, сообщает об этом и завершается.

Дочерний процесс во внешнем цикле заводит будильник (nanosleep(2)) и входит в вечный цикл, в котором заполняет структуру, содержащую пару переменных типа int, значениями {0, 0} и {1, 1} в режиме чередования.

При получении сигнала от будильника проверяет содержимое структуры, собирает статистику и повторяет тело внешнего цикла.

Через заданное количество повторений внешнего цикла (например, через 101) дочерний процесс, если ему разрешено, выводит свои PPID, PID и 4 числа — количество разных пар, зарегистрированных в момент получения сигнала от будильника.

C\_k запрашивает доступ к stdout у P и осуществляет вывод после подтверждения. По завершению вывода C\_k сообщает P об этом.

**3 РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММЫ**

Результат работы программы с опцией +:

[nina@fedora Lab03]$ make

gcc -W -Wall -Wextra -Wno-unused-parameter -Wno-unused-variable -std=c11 -pedantic parent.c -o parent

gcc -W -Wall -Wextra -Wno-unused-parameter -Wno-unused-variable -std=c11 -pedantic child.c -o child

[nina@fedora Lab03]$ ./parent

Выберите опцию: + - создать дочерний процесс;

- - удалить последний дочерний процесс;

l - вывести перечень процессов;

k - удалить все дочерние процессы;

s - запретить вывод статистики всем;

g - разрешить вывод статистики всем;

s<num> - запретить вывод статистики;

g<num> - разрешить вывод статистики;

p<num> - запретить вывод статистики всем и запросить у одного;

q - удалить все дочерние процессы и завершить работу;

h - вывести меню повторно.

+

Дочерний процесс 3286 создан

Идентификатор родительского процесса: 3285

Идентификатор дочернего процесса: 3286

Статистика: {0,0} - 26, {0,1} - 25, {1,0} - 25, {1,1} - 25

Дочерний процесс 3286 завершил вывод статистики

Результат работы программы с опцией l:

l

Перечень всех процессов:

1. Процесс 3286;

Результат работы программы с опцией -:

-

Дочерний процесс 3312 удален

Осталось 1 дочерних процессов

Результат работы программы с опцией k:

k

Дочерний процесс 3286 удален

Все дочерние процессы удалены

Результат работы программы с опцией s:

Дочернему процессу 3286 вывод статистики запрещен

Дочернему процессу 3312 вывод статистики запрещен

Результат работы программы с опцией p<num>:

p<1>

Дочернему процессу 3286 вывод статистики запрещен

Дочернему процессу 3312 вывод статистики запрещен

g

Идентификатор родительского процесса: 3285

Идентификатор дочернего процесса: 3286

Статистика: {0,0} - 480, {0,1} - 480, {1,0} - 480, {1,1} - 479

Дочерний процесс 3286 завершил вывод статистики

**4 ВЫВОД**

В ходе лабораторной работы были разработаны две программы: родительский процесс (parent), порождающий дочерние процессы и в зависимости от выбранной опции управляющий его работой; и дочерний процесс (child), сообщающий характеризующие его параметры и статистические данные при наличии разрешающего сигнала от родительского процесса.