

3 семестр

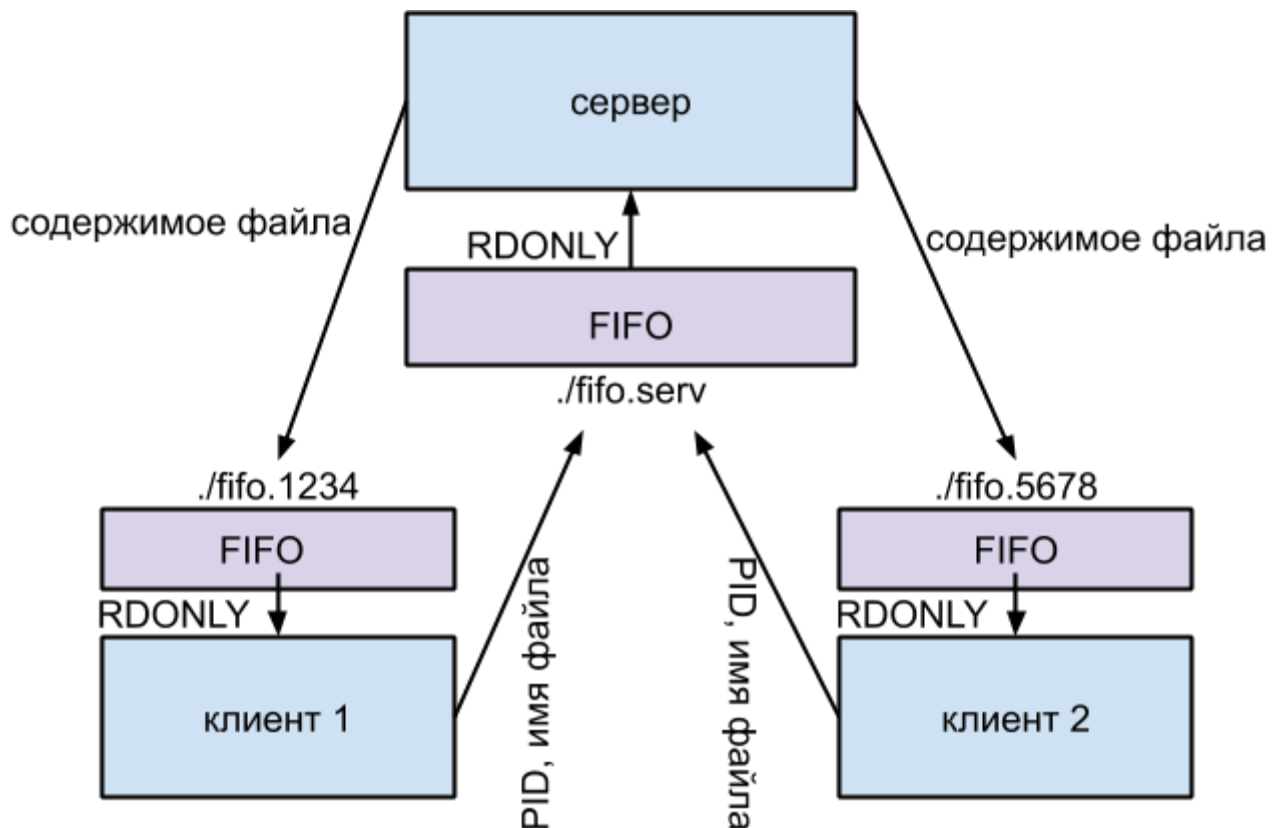
Задача №1

Написать программу, которая реализует корректную работу сервера с несколькими клиентами с помощью именованных каналов (FIFO).

Схема:

Сервер создает именованный канал с определенным именем, открывая его на чтение, а запускаемые после клиенты открывают его на запись и отправляют серверу команды и необходимые данные (имя файла/PID). Клиенты берут в качестве аргумента командной строки имя файла, передают его серверу, сервер считывает содержимое файла и передает его клиенту, который распечатывает переданное содержимое на экран. Сервер должен корректно завершаться по нажатию `ctrl+c`.

Программы должны освобождать выделенные ресурсы при любом исходе. Работа сервера с клиентами может идти поочередно в одном процессе.



Задача №2

Написать программу клиент-сервер, в которой:

- клиент считывает данные из файла (имя файла задается как аргумент командной строки => должна быть проверка на корректность аргументов) и записывает их в shared memory (4 Kb), выделенную этим процессам;
 - сервер считывает данные из shared memory и выводит на экран;
- Размер выделенной процессам shared memory ограничен 4 Kb;
Синхронизацию доступа к shared memory реализовать через семафоры;
Программа должна работать корректно, т.е. если один из процессов умер в процессе работы, то второй не должен зависнуть (журналирование/флаги и NO_WAIT и/или SEM_UNDO). Можно считать, что всегда сначала запускается сервер потом клиент.

Задача №3

Распараллелить на несколько тредов численное решение уравнения переноса.

Синхронизацию между тредом организовать с помощью очередей сообщений (System V IPC). Измерить среднее время работы тредов и построить график:

Oy – время работы одного тредов,

Ox – время работы n тредов, где n – количество тредов по степени 2.

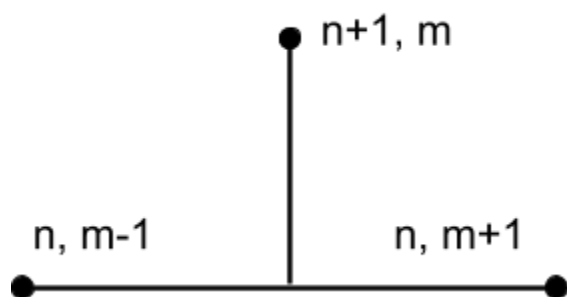
$$\frac{\partial u}{\partial t} = \frac{\partial u}{\partial x}$$

$$u(0, x) = \{1, 0 \dots 0\}$$

$$u(t, x = 0) = 1$$

$$u(t, x = L) = u(t, x = L - 1)$$

Численное решение с помощью явной центральной трехточечной схемы:



Здесь n – точки разбиения τ по шагам времени t

m – точки разбиения h по координате x .

Разностная схема запишется так:

$$\frac{u_m^{n+1} - \frac{1}{2}(u_{m+1}^n + u_{m-1}^n)}{\tau} = \frac{u_{m+1}^n - u_{m-1}^n}{2h}$$

Получается значение точки на следующем временном шаге можно найти так:

$$u_m^{n+1} = \frac{1}{2}(u_{m+1}^n + u_{m-1}^n) - \frac{\tau}{2h}(u_{m+1}^n - u_{m-1}^n)$$

Нужно разбить расчет по Oх между тредом максимально равномерно. Синхронизацию можно делать на каждом временном шаге (если придумаете оптимальнее и реализуете – это +1).

Последовательная версия transfer.c лежит здесь:

<https://github.com/ubuntik/4students.git>

Требования к программам:

- 1) должны компилироваться
- 2) должны запускаться
- 3) должны, на первый взгляд и по Вашему мнению, работать корректно, соответственно заданию
- 4) должны быть оформлены в стиле kernighan&ritchie с табуляцией в восемь пробелов (см. linux kernel coding style)

Полезные средства для отладки:

strace -- “разворачивает” системные вызовы

gdb -- отладчик