

Московский авиационный институт
(Национальный исследовательский университет)

Факультет: «Информационные технологии и прикладная
математика»

Кафедра: 806 «Вычислительная математика и программирование»

Отчет по Курсовому проекту
по курсу "Операционные системы"

Студент Петросян В.А.

группы М8О-208Б-17

№ по списку 17

Работа выполнена 04.01.2019

Преподаватель Соколов А.А.

Отчет сдан

Оценка

Москва 2019

Тема: Консоль серверная игра

Цель:

- Управление серверами сообщений
- Применение отложенных знаний за весь курс

Задание:

Реализовать консоль-серверную игру крестики-нолики. Необходимо составить программы сервера и клиента. Должны быть поддержка нескольких игр сразу. То есть если на сервер зашло 7 человек, и 6 игроков начали игру, то один игрок ждет своего соперника. Берите правила крестиков-ноликов: <https://habr.com/post/183764/>

Также ведите рейтинг победы и поражений игроков.

Тип технологии по обмену данными между приложениями остается за Вами (пайпы, очереди, сокеты)

Если клиент выключился (сам или завершился неожиданно процесс), то должно быть засчитано игроку поражение.

Решение:

Изначально правила были очень не понятными. Потратил часов 30 где-то только на написание программы без ZMQ, которая бы удовлетворяла всем условиям и правилам с сайта habr.

Для написания данной лабораторной опять использовал паттерн REQUEST - REPLY , так как он намного понятнее и освоен мною. Уверен , что это далеко не самое оптимальное решение.

Структуру хранения игроков - односвязный список . Элемент списка хранит ID игрока , указатель на след элемент списка ,статус(в игре или ожидает присоединения)и др.

Так же были реализованы все необходимые функции для работы с игрой : Add(совершить ход) Check*(Много функций с началом check проверяющих разные условия.Одни проверяли наступила ли победа,другие проверяли поражение,третьи дополнительные условия возвращаемые на сервер для успешной реализации более сложных функций из меню клиента)

Структуру программы клиента : в начале создается пустой контекст с помощью zmq_ctx_new() , затем в данном контексте создается сокет , который конектится к удаленному адресу (endpoint) . Далее программа запускает диалог между собой и клиентом . С помощью zmq_msg_send клиент отправляет запрос , затем получает ответ с помощью zmq_msg_recv и печатается сообщение от сервера .

Программа же сервера отличается следующим : вместо zmq_connect используется zmq_bind , который присоединяет созданный сокет к локальному адресу .

Для реализации остановки сервера использовал функции signal и kill из библиотеки signal.h . Также сделал возможность остановки сервера с помощью SIGINT , в результате чего после нажатия комбинации ctrl + C программа благополучна заканчивалась возвратом нулевого значения . Иначе , мы бы не смогли больше запустить данный сервер , так как его состояние будет : LIMIT , что можно увидеть с помощью утилиты nenstat с ключем -at . Убить данные процесс можно таким способом : fuser -k "имя tcp порта"/tcp , но не самый лучший способ завершения процесса , можно также использовать утилиту tcpkill.

Код программы очень большой.Я пытался его вставить в отчет,но из этого ничего хорошего не вышло.Получалось около 110 страниц.Было два выхода либо уменьшить очень сильно шрифт ,но тогда в чем смысл печатать код программы,если его не видно,поэтому просто с отчетом дополнительно прилагаю архив с программой.

Сценарий выполнения работы:

Второй игрок

Starting....

Enter the address of server : 7777

-----MENU-----

1 - To play with a certain player

2 - To play with a random player

3 - To view personal statistics

4 - To view general statistics

5 - To view database

6 - To delete your profile from DB

Enter your decision

2

Searching a player...

Player is found

You are playing against 5 player

3

2

1

Game Starts

1 2 3	10 11 12	19 20 21
4 5 6	13 14 15	22 23 24
7 8 9	16 17 18	25 26 27
28 29 30	37 38 39	46 47 48
31 32 33	40 41 42	49 50 51
34 35 36	43 44 45	52 53 54
55 56 57	64 65 66	73 74 75
58 59 60	67 68 69	76 77 78
61 62 63	70 71 72	79 80 81

.

.

.

.
.
.

.
.
.

Opponent turn

.
.
.

.
.
.

.
. . . . x
.

Enter index [1 ... 9]
5

.
.
.

.
. 0
.

.
. . . . x
.

Opponent turn

.
.
.

.
. 0 . . x

.

.
. . . . x
.

Enter index [1 ... 9]

7

.
.
.

.
. 0 . . x
. . . . 0

.
. . . . x
.

Opponent turn

You won by resignation

Quitting the game

Просмотр общей статистики игроков на сервере

victor@victor:~/Рабочий стол/FINAL_VERSION

2-20190219T154654Z-001/FINAL_VERSION 2\$./player 17

Starting....

Enter the address of server : 7777

-----MENU-----

1 - To play with a certain player

2 - To play with a random player

3 - To view personal statistics

4 - To view general statistics

5 - To view database

6 - To delete your profile from DB

Enter your decision

4

-----Statistics-----

ID	17	won 1 times and lost 0 times
ID	1	won 0 times and lost 0 times
ID	2	won 0 times and lost 0 times
ID	3	won 0 times and lost 0 times
ID	4	won 0 times and lost 0 times

Первый игрок

victor@victor:~/Рабочий стол/FINAL_VERSION 2-20190219T154654Z-001/FI 2\$./

player 5

Starting....

Enter the address of server : 7777

-----MENU-----

1 - To play with a certain player

2 - To play with a random player

3 - To view personal statistics

4 - To view general statistics

5 - To view database

6 - To delete your profile from DB

Enter your decision

2

Searching a player...

Player is found

You are playing against 17 player

3

2

1

Game Starts

1 2 3 10 11 12 19 20 21

4 5 6 13 14 15 22 23 24

7 8 9 16 17 18 25 26 27

28 29 30 37 38 39 46 47 48

31 32 33 40 41 42 49 50 51

34 35 36 43 44 45 52 53 54

55 56 57 64 65 66 73 74 75

58 59 60 67 68 69 76 77 78

61 62 63 70 71 72 79 80 81

.

.

.

.

.

.

.

.

.

Enter index [1 ... 81]

67

.

.

.

.

.

.

.

. . . . X

.

Opponent turn

.
.
.

.
. 0
.

.
. . . x
.

Enter index [1 ... 9]

47

Inccorect index.Please type again

Enter index [1 ... 9]

ewfqq

Inccorect index.Please type again

Enter index [1 ... 9]

qe

Inccorect index.Please type again

Enter index [1 ... 9]

5

.
.
.

.
. 0 . . x
.

.
. . . x
.

Opponent turn

.
.
.

.

. 0 . . x
. . . . 0

.
. . . . x
.

Enter index [1 ... 9]

exit

You lost by resignation

Quitting the game

Просмотр личной статистики.

victor@victor:~/Рабочий стол/FINAL_VERSION 2-20190219T154654Z-001/FI 2\$./

player 5

Starting....

Enter the address of server : 7777

-----MENU-----

1 - To play with a certain player

2 - To play with a random player

3 - To view personal statistics

4 - To view general statistics

5 - To view database

6 - To delete your profile from DB

Enter your decision

3

victor@victor:~/Рабочий стол/FINAL_VERSION
2-20190219T154654Z-001/FINAL_VERSION 2\$

Сервер

victor@victor:~/Рабочий стол/FINAL_VERSION

2-20190219T154654Z-001/FINAL_VERSION 2\$./server 7777

ID	STATUS	TYPE	ID_OPPONENT
5	1	1	0

ID	STATUS	TYPE	ID_OPPONENT
5	1	1	0
17	1	2	0

ID	STATUS	TYPE	ID_OPPONENT
5	2	1	17
17	2	2	5

ID	STATUS	TYPE	ID_OPPONENT
5	2	1	17
17	2	2	5

ID	STATUS	TYPE	ID_OPPONENT
5	3	1	17
17	3	2	5

Вывод:

Это были очень интересные зимние каникулы.

Более чем уверен, что программа будет оценена любым опытным программистом как провальная. Почему?

Потому что она далеко не оптимальна как по памяти, так и по времени. К сожалению не смог оптимизировать программу как следует, но хочу подчеркнуть, что были предприняты некоторые меры оптимизации.

1) Если в базу данных решили добавить нового игрока и у нас в списке был удаленный игрок, то новый (добавляемый) встанет на место старого игрока.

Какие слабости у этого алгоритма?

Для данной реализации самый плохой вариант если создать очень много игроков. Потом их всех кроме двух удалить и играть только этими двумя на сервере. Конечно на быстрое действие их игры это не сильно повлияет, так как у

них уже установилось соединение и им больше не нужно обращаться к базе данных,но у нас будет висеть очень много памяти ожидающей своего череда.

2) Насколько смог сократил количество вспомогательных локальных переменных в файле functions.c.

3) Если под конкретным ID в данный момент ведется игра, то кто-то другой не сможет зайти под его ID на сервер пока игра не завершится.

Также у меня есть пару пунктов в TODO

1)Не вводить ID игрока с клавиатуры.Возможно это более удобно с точки зрения запуска программы,но с другой стороны вдруг игрок хочет поиграть с каким-то своим другом.Как он тогда узнает ID друга ,что конкретно с ним играть?

Конечно можно будет ввести доп функции ,чтобы они возвращали клиенту при запросе его автоматически определенный ID с сервера , но это дополнительно нагрузит программу,чего бы не хотелось.

2)Профилировать код с помощью gprof .Проанализировать граф вывода и хоть как-то еще оптимизировать программу.

3)Подключить графическую библиотеку СИ.

Моя программа не идеальная конечно,но и полным провалом её не назовешь. Она придала мне уверенности в себе и моих возможностях в программировании.