

# Operating Systems. IHW 3

## [Operating Systems](#)

Александр Васюков | БПИ235

## Вариант 35

«Камень, ножницы, бумага» 1 – однокруговая система.

N студентов, изнывающих от скуки на лекции по операционным системам решили организовать однокруговой турнир в игру «Камень, ножницы, бумага». Каждый с каждым при этом играет только один раз. За победу дается два очка, за ничью – одно, за поражение – ноль.

Требуется создать клиент–серверное приложение, моделирующее турнир. Каждый студент — отдельный клиент. Генерация камня, ножниц и бумаги в каждом поединке формируется сервером случайно. Завершение работы программ осуществляется по завершению турнира. Количество участвующих студентов задается сервером.

Решение претендует на 10 баллов.

Программы можно найти на Github по ссылке: [https://github.com/vasyukov1/HSE-FCS-SE-2-year/tree/main/Operating%20Systems/Homeworks/IHW\\_3](https://github.com/vasyukov1/HSE-FCS-SE-2-year/tree/main/Operating%20Systems/Homeworks/IHW_3)

## Решение

### На 4-5 баллов

Разработать клиент–серверное приложение, в котором сервер (или серверы) и клиенты независимо друг от друга отображают только ту информацию, которая поступает им во время обмена. То есть, отсутствует какой-либо общий вывод интегрированной информации, отображающий поведение системы в целом

### Сервер

Управляет турниром: принимает подключения, генерирует ходы, вычисляет результаты, рассылает данные.

### Клиент

Представляет студента: подключается к серверу, получает и отображает результаты матчей.

1. Сервер создает TCP-сокеты, привязывается к указанному IP и порту, после чего ожидает подключения всех клиентов (по числу NUM\_STUDENTS).
2. Каждому клиенту присваивается уникальный ID, начиная с 1.
3. Проведение матчей:
  - Сервер генерирует все возможные пары игроков.

- Для каждой пары:
  - Генерируются случайные ходы (камень / ножницы / бумага).
  - Определяется результат (победа / ничья / поражение).
  - Начисляются очки.
  - Сервер отправляет клиентам результаты их матчей.

#### 4. Отправка результатов после завершения всех матчей:

- Сервер сортирует игроков по количеству очков.
- Рассылает клиентам итоговый рейтинг в формате: `RANKING <ID первого места>: <количество очков> .`
- Отправляет команду `END` с финальным счетом и закрывает соединения.

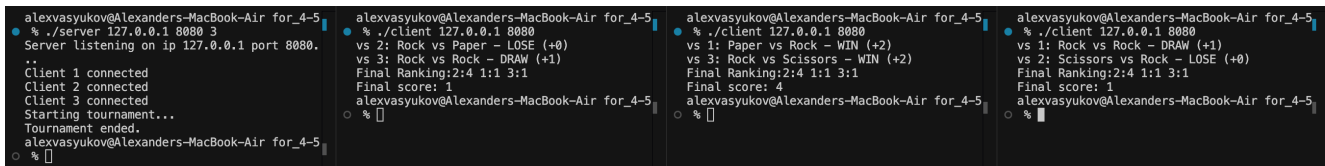
## Запуск

```
cd for_4-5

gcc server.c -o server
gcc client.c -o client

./server <IP> <PORT> <NUM_STUDENTS>
./client <IP> <PORT>
```

## Результат:



```
alexvasyukov@Alexanders-MacBook-Air for_4-5
% ./server 127.0.0.1 8080 3
Server listening on ip 127.0.0.1 port 8080.
..
Client 1 connected
Client 2 connected
Client 3 connected
Starting tournament...
Tournament ended.
alexvasyukov@Alexanders-MacBook-Air for_4-5
%

alexvasyukov@Alexanders-MacBook-Air for_4-5
% ./client 127.0.0.1 8080
vs 2: Rock vs Paper - LOSE (+0)
vs 3: Rock vs Rock - DRAW (+1)
Final Ranking:2:4 1:1 3:1
Final score: 1
alexvasyukov@Alexanders-MacBook-Air for_4-5
%

alexvasyukov@Alexanders-MacBook-Air for_4-5
% ./client 127.0.0.1 8080
vs 1: Paper vs Rock - WIN (+2)
vs 3: Rock vs Scissors - WIN (+2)
Final Ranking:2:4 1:1 3:1
Final score: 4
alexvasyukov@Alexanders-MacBook-Air for_4-5
%

alexvasyukov@Alexanders-MacBook-Air for_4-5
% ./client 127.0.0.1 8080
vs 1: Rock vs Rock - DRAW (+1)
vs 2: Scissors vs Rock - LOSE (+0)
Final Ranking:2:4 1:1 3:1
Final score: 1
alexvasyukov@Alexanders-MacBook-Air for_4-5
%
```

## На 6-7 баллов

В дополнение к программе на предыдущую оценку необходимо разработать отдельную клиентскую программу, подключаемую к серверу, которая предназначена для отображение комплексной информации о выполнении приложения в целом. То есть, данный программный модуль должен адекватно и в полном виде отображать поведение моделируемой системы (информацию передаваемую на сервер и информацию, порождаемую сервером), позволяя не пользоваться отдельными видами, предоставляемыми клиентами и серверами по отдельности.

Разработан отдельный клиентский модуль, отображающий информацию о турнире:

- Подключение участников
- Начало матчей
- Результаты матчей
- Финальные результаты

В сервере добавлена структура для наблюдателей.

Сервер отправляет наблюдателям сообщения формата:

- `PLAYER_CONNECTED <ID>` – подключение игрока
- `MATCH_START <ID1> <ID2>` – начало матча
- `MATCH_RESULT <ID1> <Ход1> <ID2> <Ход2> <Результат>` – результат матча
- `RANKING` – финальный рейтинг

Клиент-игрок и клиент-наблюдатель работают независимо.

## Запуск

```
cd for_6-7

gcc server.c -o server
gcc client.c -o client
gcc observer.c -o observer

./server <IP> <PORT> <NUM_STUDENTS>
./observer <IP> <PORT>
./client <IP> <PORT>
```

## Результат:

```
alexvasyukov@Alexanders-MacBook-Air for_6-7 % ./server 127.0.0.1 8080 3
Server listening on ip 127.0.0.1 port 8080...
Observer connected
Player 1 connected
Player 2 connected
Player 3 connected
Starting tournament...
Waiting for observers...
Closing connections...
Tournament ended.
alexvasyukov@Alexanders-MacBook-Air for_6-7 %

alexvasyukov@Alexanders-MacBook-Air for_6-7 % ./observer 127.0.0.1 8080
[System] Player 1 connected
[System] Player 2 connected
[System] Player 3 connected
[Match] Player 1 vs Player 2 started
[Result] 1 (Paper) vs 2 (Scissors) -> WIN2
[Match] Player 1 vs Player 3 started
[Result] 1 (Paper) vs 3 (Scissors) -> WIN2
[Match] Player 2 vs Player 3 started
[Result] 2 (Scissors) vs 3 (Paper) -> WIN1
[Ranking] 2:4 3:2 1:0
alexvasyukov@Alexanders-MacBook-Air for_6-7 %

alexvasyukov@Alexanders-MacBook-Air for_6-7 % ./client 127.0.0.1 8080
vs 2: Paper vs Scissors - LOSE (+0)
vs 3: Paper vs Scissors - LOSE (+0)
Final Ranking: 2:4 3:2 1:0
alexvasyukov@Alexanders-MacBook-Air for_6-7 %

alexvasyukov@Alexanders-MacBook-Air for_6-7 % ./client 127.0.0.1 8080
vs 1: Scissors vs Paper - WIN (+2)
vs 3: Scissors vs Paper - WIN (+2)
Final Ranking: 2:4 3:2 1:0
alexvasyukov@Alexanders-MacBook-Air for_6-7 %

alexvasyukov@Alexanders-MacBook-Air for_6-7 % ./client 127.0.0.1 8080
vs 1: Scissors vs Paper - WIN (+2)
vs 2: Paper vs Scissors - LOSE (+0)
Final Ranking: 2:4 3:2 1:0
alexvasyukov@Alexanders-MacBook-Air for_6-7 %
```

## На 8 баллов

В дополнение к предыдущей программе реализовать возможность, подключения множества клиентов, обеспечивающих отображение информации о работе приложения. Это должно позволить осуществлять наблюдение за поведением программы с многих независимых компьютеров.

- Клиент и наблюдатель остались без изменений.
- Сервер поддерживает подключение нескольких наблюдателей.
- Наблюдатели могут динамически подключаться и отключаться во время и после турнира.

- Все новые наблюдатели получают полную историю событий - для этого сервер запоминает все события.
- Сервер корректно обрабатывает множественные подключения

## Запуск

```
cd for_8

gcc server.c -o server
gcc client.c -o client
gcc observer.c -o observer

./server <IP> <PORT> <NUM_STUDENTS>
./observer <IP> <PORT>
./client <IP> <PORT>
```

## Результат:

```
alexvasyukov@Alexanders-MacBook-Air for_8 % ./server 127.0.0.1 8080 3
1 8080 3
Server listening on ip 127.0.0.1 port 8080...
Observer connected
Player 1 connected
Player 2 connected
Observer connected
Player 3 connected
Starting tournament...
Tournament finished. Waiting for observers...
Closing connections...
alexvasyukov@Alexanders-MacBook-Air for_8 %

alexvasyukov@Alexanders-MacBook-Air for_8 % ./observer 127.0.0.1 8080
0.1 8080
[System] Player 1 connected
[System] Player 2 connected
[System] Player 3 connected
[Match] Player 1 vs Player 2 started
[Result] 1 (Rock) vs 2 (Scissors) -> WIN1
[Match] Player 1 vs Player 3 started
[Result] 1 (Scissors) vs 3 (Rock) -> WIN2
[Match] Player 2 vs Player 3 started
[Result] 2 (Scissors) vs 3 (Paper) -> WIN1
[Ranking]1:2 2:2 3:2
alexvasyukov@Alexanders-MacBook-Air for_8 %

alexvasyukov@Alexanders-MacBook-Air for_8 % ./client 127.0.0.1 8080
0.1 8080
vs 2: Rock vs Scissors - WIN (+2)
vs 3: Scissors vs Rock - LOSE (+0)
Final Ranking:1:2 2:2 3:2
alexvasyukov@Alexanders-MacBook-Air for_8 %
```

## На 9-10 баллов

В дополнение к программам на предыдущие оценки необходимо разработать приложение, позволяющее отключать и подключать различных клиентов с сохранением работоспособности сервера.

Расширить предыдущую программу таким образом, чтобы при завершении работы сервера происходило корректное завершение работы всех подключенных клиентов. То есть, данная программа должна являться модификацией программы на оценку в 9 баллов.

Отдельную программу на 9 баллов в этом случае сдавать не нужно

- Клиент и наблюдатель остались без изменений.
- В сервере у структуры клиента добавлен флаг `active`.
- Также в сервере добавлен обработчик отключения клиента `handle_disconnected_player(int player_id, int server_sock)`, который ждёт переподключения клиента.
- При завершении работы сервера корректно завершается работа клиентов и наблюдателя.

## Запуск

```
cd for_9-10

gcc server.c -o server
gcc client.c -o client
gcc observer.c -o observer

./server <IP> <PORT> <NUM_STUDENTS>
./observer <IP> <PORT>
./client <IP> <PORT>
```

## Результат:

Корректная работа:

```
alexvasyukov@Alexanders-MacBook-Air for_9 % ./server 127.0.0.1 8080 3
Server listening on ip 127.0.0.1 port 8080...
Observer connected
Player 1 connected
Player 2 connected
Player 2 disconnected
Waiting for new connection to replace player 2...
Player 2 reconnected
Player 3 connected
Starting tournament...
Tournament finished. Waiting for observers...
Closing connections...
alexvasyukov@Alexanders-MacBook-Air for_9 %

alexvasyukov@Alexanders-MacBook-Air for_9 % ./observer 127.0.0.1 8080
[System] Player 1 connected
[System] Player 2 connected
[System] Player 3 connected
[Match] Player 1 vs Player 2 started
[Result] 1 (Rock) vs 2 (Scissors) - WIN1
[Match] Player 1 vs Player 3 started
[Result] 1 (Scissors) vs 3 (Rock) - WIN2
[Match] Player 2 vs Player 3 started
[Result] 2 (Rock) vs 3 (Scissors) - WIN1
[Ranking]1:2 2:2 3:2
alexvasyukov@Alexanders-MacBook-Air for_9 %

alexvasyukov@Alexanders-MacBook-Air for_9 % ./client 127.0.0.1 8080
vs 2: Rock vs Scissors - WIN (+2)
vs 3: Scissors vs Rock - LOSE (+0)
Final Ranking:1:2 2:2 3:2
alexvasyukov@Alexanders-MacBook-Air for_9 %

alexvasyukov@Alexanders-MacBook-Air for_9 % ./client 127.0.0.1 8080
vs 1: Scissors vs Rock - LOSE (+0)
vs 3: Rock vs Scissors - WIN (+2)
Final Ranking:1:2 2:2 3:2
alexvasyukov@Alexanders-MacBook-Air for_9 %

alexvasyukov@Alexanders-MacBook-Air for_9 % ./client 127.0.0.1 8080
vs 1: Rock vs Scissors - WIN (+2)
vs 2: Scissors vs Rock - LOSE (+0)
Final Ranking:1:2 2:2 3:2
alexvasyukov@Alexanders-MacBook-Air for_9 %
alexvasyukov@Alexanders-MacBook-Air for_9 %
```

Завершение работы сервера:

```
alexvasyukov@Alexanders-MacBook-Air for_9 % ./server 127.0.0.1 8080 3
Server listening on ip 127.0.0.1 port 8080...
Observer connected
Player 1 connected
Player 2 connected
^C
alexvasyukov@Alexanders-MacBook-Air for_9 %

alexvasyukov@Alexanders-MacBook-Air for_9 % ./observer 127.0.0.1 8080
[System] Player 1 connected
[System] Player 2 connected
alexvasyukov@Alexanders-MacBook-Air for_9 %

alexvasyukov@Alexanders-MacBook-Air for_9 % ./client 127.0.0.1 8080
alexvasyukov@Alexanders-MacBook-Air for_9 %

alexvasyukov@Alexanders-MacBook-Air for_9 % ./client 127.0.0.1 8080
alexvasyukov@Alexanders-MacBook-Air for_9 %

alexvasyukov@Alexanders-MacBook-Air for_9 %
```