

ИГРА шахматы

Игра предполагает наличие 2х игроков, находящихся в одной комнате. Поэтому на сервере при заходе игрока создается комната и он начинает играть за белых. Может сделать один ход. Далее сервер ожидает захода еще одного игрока, который укажет id комнаты первого человека. Зайти можно только в комнату, где есть ровно 1 человек или в комнату, в которую уже заходил раньше. После захода двух игроков начинается игра. Игроки могут ходить строго по очереди. Иначе они будут получать ошибку, что хотят сделать ход не в свою очередь.


Игроки определяются по логину и паролю. Есть несколько предзарегистрированных игроков. Можно дополнительно дозарегистрироваться.

В игре может быть несколько комнат и несколько клиентов одновременно могут играть.

Протокол:


Registration:

CLIENT	SERVER
REG please LOGIN %login% PASSW %password% ---->	Регистрация пользователя, если л
<----	SUCC REG, если регистрация успешна / ERROR NON_UNIC
	если логин уже есть




Login:

CLIENT	SERVER
LOGIN %login% PASSW %password% ---->	аутентификация пользователя,
<----	SUCC AUTH, если аутентификация прошла успешно /
	один из ответов в зависимости от ситуации: /
	ERROR NO_LOGIN, ERROR NO_PASSWORD, ERROR LOGIN_OR_PASSW



Создание комнаты (возможно только после регистрации или аутентификации, иначе вернется ERROR UNKNOWN_COMMAND):

CLIENT	SERVER
ROOM NEW	----> Создает новую комнату
<----	GAME_ID %id% возвращает id, по которому можно попасть в ко



Подключение к существующей комнате (возможно только после регистрации или аутентификации, иначе вернется ERROR UNKNOWN_COMMAND):

CLIENT

SERVER

ROOM %id%

----> Проверяет, что комната существует
<---- SUCC ROOM, если id верный, иначе ERROR INCORRECT_ROOM_ID.
если есть комната, у которой только один человек, то человек
добавляется туда если по id есть комната, но login пользователя
не совпадает ни с одним из участников, \
возвращается ERROR INCORRECT_ROOM_ID_ERROR

Попав в комнату можно начинать делать ходы:

CLIENT

SERVER

MOVE 11-12 (переместить фигуру с поля 1,1 на 1,2) ----> Делает ход и меняет местопо.
(проверяет корректность хода в перспективе)
Возвращает ERROR INCORRECT_MOVE, если ход
недопустимый
иначе возвращает карту(описание см ниже):
<---- [
["r","kn","b","q","kg","b","kn","r"],
[".",".","p","p","p","p","p","p"],
[".",".",".",".",".",".",".","."],
[".",".",".",".",".",".",".","."],
[".",".",".",".",".",".",".","."],
[".",".",".",".",".",".",".","."],
["P","P","P","P","P","P","P","p"],
["R","KN","B","KG","Q","B","KN","r"]
]
Если сейчас ход другого игрока, то возвращает
ERROR NOT_YOUR_TURN

Есть пользователь admin, который может получить всех игроков на сервере, для этого нужно
отправить логин и пароль от админа (admin, admin)

CLIENT	SERVER
LOGIN admin PASSW admin	----> Проверяет логин и пароль админа
	<---- SUCC AUTH (возвращает ERROR NON_UNIQUE_LOGIN если логин или пароль не существуют)
GET PLAYERS	----> Собирает информацию о всех пользователях на сервере и возвращает логины зарегистрированных пользователей
	<---- admin
	lala
	mylogin

Описание карты: r - rook, kn - knight, b - bishop, kg - king, q - queen. Большая буква - белые, маленькая - черные.

Команды заканчиваются на \n, порядок команд строгий

Пишется лог запросов и команд в файл, из-за чего игры можно восстановить.