Report

Programming RED3.7

Ivan Trendafilov - s2309858
Vladi Avramov - s2292165

1. CheckStyle - the checkstyle did not work on neither mine nor my partner's computer and when the new version was uploaded we had to remake 3000 lines of code and there was nothing that we could do about it. We asked a TA and he told us to leave it as it is.

2. Discussions of overall design:

Entering a port on which the server to be opened is implemented in the AbaloneServer method run().

Entering a port that is already in use is implemented in the AbaloneServer method run().

The AbaloneServer is single threaded but the AbaloneClientHandler is multithreaded so it allows the clients to create and play multiple games.

After accepting a message the AbaloneServerTUI represents the message.

All the methods that are written in the protocol are fulfilled in the AbaloneServer/AbaloneClientHandler except for the exception duplicate username.We set up to directly give a username and later the people from our group changed the protocol without asking anyone and we could not rewrite our code in a way to suite that exception.

Giving IP and Port has been implemented in the class AbaloneClient and the server has a setUp which sets up the IP of the server on the IPV4 of the computer.

The AbaloneClientHandler is multithreaded and can support multiple clients per game.We have a different class ComputerAbaloneClient which implements the computer player with 2 types of strategies which can be selected.

The hint is implemented in the AbaloneClient.

After a game has finished the client can again create a new room that is implemented in AbaloneClient.

The quit functions is implemented in the AbaloneServer it sends to all players that the game has finished and it sends them again in the lobby.

The crashing of the server has been handled as an exception in the AbaloneClient and it prints a stack trace and it stops the connection.

The AbaloneClient support all requirements that have been listed in the protocol which are not for bonus points.

Also there is an error that is: if you disconnect from the server and then you connect you can write some commands that you should not be able to write.

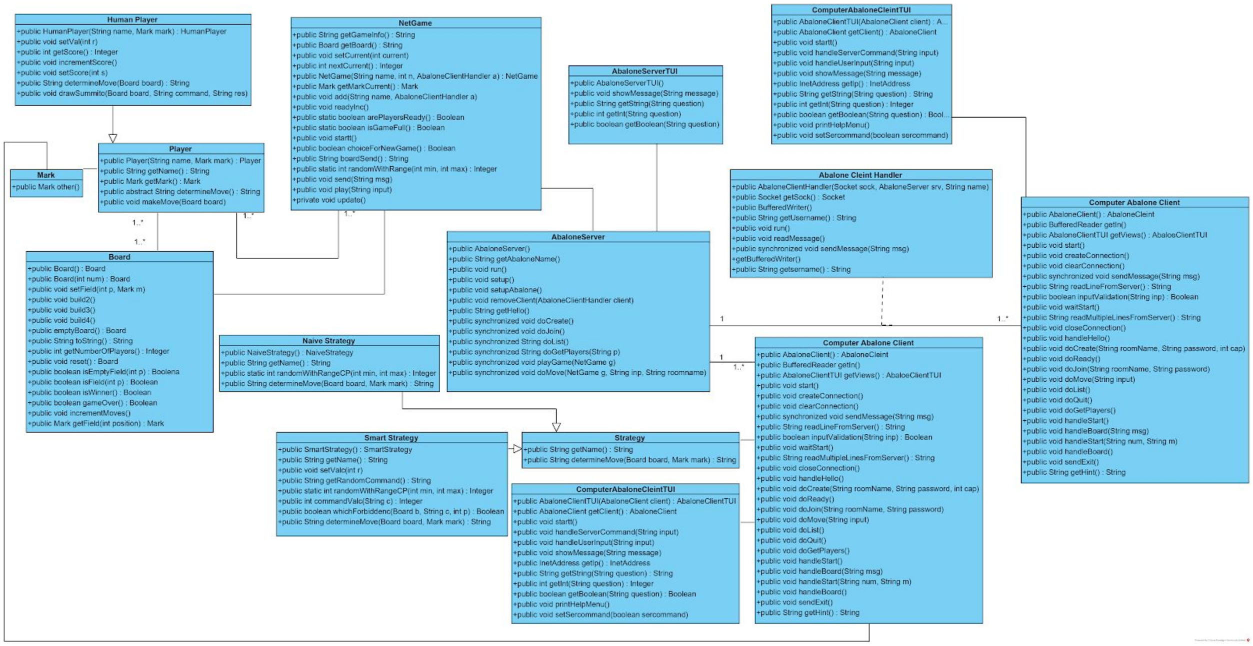
3 - The package server plays the role of MVC - and the classes that play the role of model-view-controller pattern are :

AbaloneClientView/AbaloneServerView/AbaloneClientTUI/AbaloneServerTUI/
ComputerAbaloneTUI - View

AbaloneClientHandler/ - Controller

AbaloneClient/AbaloneServer/ComputerAbalonePlayer - Model

Class Diagram:


Human Player
+public HumanPlayer(String name, Mark mark) : HumanPlayer
+public void setVal(int r)
+public int getScore() : Integer
+public void incrementScore()
+public void setScore(int s)
+public String determineMove(Board board) : String
+public void drawSummito(Board board, String command, String res)
NetGame
+public String getGameInfo() : String
+public Board getBoard() : String
+public void setCurrent(int current)
+public int nextCurrent() : Integer
+public NetGame(String name, int n, AbaloneClientHandler a) : NetGame
+public Mark getMarkCurrent() : Mark
+public void add(String name, AbaloneClientHandler a)
+public void readyInc()
+public static boolean arePlayersReady() : Boolean
+public static boolean isGameFull() : Boolean
+public void startt()
+public boolean choiceForNewGame() : Boolean
+public String boardSend() : String
+public static int randomWithRange(int min, int max) : Integer
+public void send(String msg)
+public void play(String input)
+private void update()
AbaloneServerTUI
+public AbaloneServerTUI()
+public void showMessage(String message)
+public String getString(String question)
+public int getInt(String question)
+public boolean getBoolean(String question)
ComputerAbaloneCleintTUI
+public AbaloneClientTUI(AbaloneClient client) : A...
+public AbaloneClient getClient() : AbaloneClient
+public void startt()
+public void handleServerCommand(String input)
+public void handleUserInput(String input)
+public void showMessage(String message)
+public InetAddress getIp() : InetAddress
+public String getString(String question) : String
+public int getInt(String question) : Integer
+public boolean getBoolean(String question) : Bool...
+public void printHelpMenu()
+public void setSercommand(boolean sercommand)
Mark
+public Mark other()
Player
+public Player(String name, Mark mark) : Player
+public String getName() : String
+public Mark getMark() : Mark
+public abstract String determineMove() : String
+public void makeMove(Board board)
1..*
1..*
1..*
1..*
Abalone Cleint Handler
+public AbaloneClientHandler(Socket sock, AbaloneServer srv, String name)
+public Socket getSock() : Socket
+public BufferedWriter()
+public String getUsername() : String
+public void run()
+public void readMessage()
+public synchronized void sendMessage(String msg)
+getBufferedWriter()
+public String getsername() : String
Computer Abalone Client
+public AbaloneClient() : AbaloneCleint
+public BufferedReader getIn()
+public AbaloneClientTUI getViews() : AbaloeClientTUI
+public void start()
+public void createConnection()
+public void clearConnection()
+public synchronized void sendMessage(String msg)
+public String readLineFromServer() : String
+public boolean inputValidation(String inp) : Boolean
+public void waitStart()
+public String readMultipleLinesFromServer() : String
+public void closeConnection()
+public void handleHello()
+public void doCreate(String roomName, String password, int cap)
+public void doReady()
+public void doJoin(String roomName, String password)
+public void doMove(String input)
+public void doList()
+public void doQuit()
+public void doGetPlayers()
+public void handleStart()
+public void handleBoard(String msg)
+public void handleStart(String num, String m)
+public void handleBoard()
+public void sendExit()
+public String getHint() : String
Board
+public Board() : Board
+public Board(int num) : Board
+public void setField(int p, Mark m)
+public void build2()
+public void build3()
+public void build4()
+public emptyBoard() : Board
+public String toString() : String
+public int getNumberOfPlayers() : Integer
+public void reset() : Board
+public boolean isEmptyField(int p) : Boolena
+public boolean isField(int p) : Boolean
+public boolean isWinner() : Boolean
+public boolean gameOver() : Boolean
+public void incrementMoves()
+public Mark getField(int position) : Mark
AbaloneServer
+public AbaloneServer()
+public String getAbaloneName()
+public void run()
+public void setup()
+public void setupAbalone()
+public void removeClient(AbaloneClientHandler client)
+public String getHello()
+public synchronized void doCreate()
+public synchronized void doJoin()
+public synchronized String doList()
+public synchronized String doGetPlayers(String p)
+public synchronized void playGame(NetGame g)
+public synchronized void doMove(NetGame g, String inp, String roomname)
1
1..*
1
1..*
Naive Strategy
+public NaiveStrategy() : NaiveStrategy
+public String getName() : String
+public static int randomWithRangeCP(int min, int max) : Integer
+public String determineMove(Board board, Mark mark) : String
Computer Abalone Client
+public AbaloneClient() : AbaloneCleint
+public BufferedReader getIn()
+public AbaloneClientTUI getViews() : AbaloeClientTUI
+public void start()
+public void createConnection()
+public void clearConnection()
+public synchronized void sendMessage(String msg)
+public String readLineFromServer() : String
+public boolean inputValidation(String inp) : Boolean
+public void waitStart()
+public String readMultipleLinesFromServer() : String
+public void closeConnection()
+public void handleHello()
+public void doCreate(String roomName, String password, int cap)
+public void doReady()
+public void doJoin(String roomName, String password)
+public void doMove(String input)
+public void doList()
+public void doQuit()
+public void doGetPlayers()
+public void handleStart()
+public void handleBoard(String msg)
+public void handleStart(String num, String m)
+public void handleBoard()
+public void sendExit()
+public String getHint() : String
Smart Strategy
+public SmartStrategy() : SmartStrategy
+public String getName() : String
+public void setValc(int r)
+public String getRandomCommand() : String
+public static int randomWithRangeCP(int min, int max) : Integer
+public int commandValc(String c) : Integer
+public boolean whichForbiddenc(Board b, String c, int p) : Boolean
+public String determineMove(Board board, Mark mark) : String
Strategy
+public String getName() : String
+public String determineMove(Board board, Mark mark) : String
ComputerAbaloneCleintTUI
+public AbaloneClientTUI(AbaloneClient client) : AbaloneClientTUI
+public AbaloneClient getClient() : AbaloneClient
+public void startt()
+public void handleServerCommand(String input)
+public void handleUserInput(String input)
+public void showMessage(String message)
+public InetAddress getIp() : InetAddress
+public String getString(String question) : String
+public int getInt(String question) : Integer
+public boolean getBoolean(String question) : Boolean
+public void printHelpMenu()
+public void setSercommand(boolean sercommand)

For full resolution version:
https://drive.google.com/open?id=1nCLrItCSeFMfIyi9Wv5Wen2icMNnxchK

Overall, the design of the application in regards to the game-server is based on the communication client-server-execution on board, and then sending the new board back to the client for the next move. In that way part of the validation can be done on client, because some of the rules can be checked on client which makes the game faster because no commands will be sent in good amount of cases. After a move is sent a full validation is made on server in-game and it returns a response fail if it is not a possible move and otherwise sends the updated board(via NetGame) to all clients and then gives the turn to the next one. The assignment of colors is based on the room joining(first come, first served Red, then yellow, then green, then pink, and this all-over again). A map of games(NetGame) played is being kept on server with key name of game(room), capacity with key name of game(room), also passwords with key name of game(room), players are kept in a list in each NetGame. Most commands are kept and validated on client level, because some functions like creation and joining can't be done while in a room. The quit function works correctly in game, not entirely perfect while just joined in a room. If a person leaves mid-game, everyone gets a message. If anyone wins, or a game is draw everyone get a corresponding message. Those things are checked after every correct move. We tested at best two games played at the same time, because our PCs run out of CPU power. We also tested two computer players playing against each other, it works correctly.

The game design is based on the Tic-Tac-Toe from week 4. We started to work on everything, brute-forcing it, but it was way too slow and big, therefore we reworked 90% of the HumanPlayer which decides everything about a move(without the sumito for 3 players which is still original code, but we were a week before the deadline so we left it working). The human player gets the input and does a full validation, is a command valid, is it some kind of summito, is it a regular move, and if so does the recolor and returns “ok”. The board is kept in an array and visualised via toString, we wrote additional functions for validation on board if a field is forbidden in order to check for validity of move, or whether a marble should be pushed. Marks are kept in class Mark, and there we can get our partner’s mark(for 4-player games). The human player is called from the server via NetGame for those functions, it is called with the board it should operate on and the move as given by the player. This is the most complex of all our classes, but it was way too late in order to break it into smaller parts, although it would have been good. Otherwise the game operates in a similar fashion to the tic-tac-toe(as you can see from the diagram too).

The client-server relationship is based on the changes between listening mode and command giving mode on the client. The server is always listening and reacting. The whole server-client structure is based on the code from week 7, so similarities with it are possible. The client changes its state based on the command given. For example, after joined in a room, a player enters ready and that launches him into listening state, waiting for the start command from server. After receiving it, the client goes back to read a command, after receiving the Turn command, while other clients still wait for their turn to come. The client stays in this mode until a valid move is sent, then the turn is given to the next player until the game is over. A menu is printed to every client on his turn. After a client quits or disconnects, the server informs all other players and kicks them out of a room(game), and leaves them to decide their next move. A client can at all times decide to disconnect and that will also kick people out of a room.

Some of the errors that we found were:
When moving 2 marbles of different teammates they would duplicate let’s say we move red/yellow the end result would be red/red.
When making a sumito for 3vs1 marbles the end marble would just disappear.
When making a sumito for 3vs2 marbles the pushed marbles would not colour the correct way.
When pushing an opponent marble out of the board the score of the one who has pushed the marbles would not increment.
When asking for a list of all the rooms in the game if there were more than one room we would receive only 1 room and not all.
When creating a board it would not paint it the correct way for the number of players.
When adding the computer player to the server he would not output the correct commands.
When one player quits the others do not get notified or kicked out of the room.
When the server receives a command for moving a marble and updates the board, the server would not send the new board to the players.

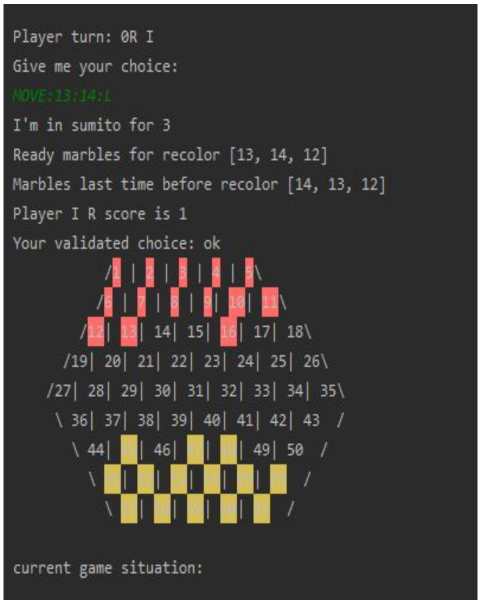
The JUNIT tests that have full code coverage are the BoardTest and the SmartStrategyTest both these tests have 100% coverage.
So some of the tests that we wanted to try can't be done with a JUNIT test (Human player we also could not get 100% test coverage) because they need an input(we even asked TA and they told us to leave them like that) and we decided to just try different end case tests to test if they work, so some of the tests that we tried are:
1 marble move.
Move:27:L - not possible,Move:35:R - not possible,Move:19:UL,Move:18:UR - not possible,Move:50:DR - not possible,Move:44:DL - not possible.
2 marbles move.
Sumito and push for 2 marbles vs 1 for(2,3 players).
Move:1:2:R(marbles of red player) vs 1marble(of yellow player on position 3 and position 4 empty) - push successful,marbles 2:3 painted red marble 4 painted yellow.
Move:3:4:R(marbles of red player) vs 1marble(of yellow player on position 5 end position on board) - push successful,marbles 4:5 painted red and score of red player incremented with 1.
Move:3:4:L(marbles of yellow player) vs 1marble(of red player on position 2 and position 1 empty) - push successful,marbles 2:3 painted yellow marble 1 painted red.
Move:2:3:L(marbles of yellow player) vs 1marble(of red player on position 1 end spot on board) - push successful,marbles 1:2 painted red marble and score of yellow player incremented by 1.
Move:1:7:DR(marbles of red player) vs 1marble(of yellow player on position 14 and position 22 empty) - push successful,marbles 7:14 painted red and marble 22 painted yellow.
Move:33:42:DR(marbles of red player) vs 1marble(of yellow player on position 50 end spot on board) - push successful,marbles 42:50 painted red and score of red player incremented by 1.
Move:21:29:DL(marbles of red player) vs 1marble(of yellow player on position 37 and position 44 empty) - push successful,marbles 29:37 painted red and marble 44 painted yellow.
Move:29:37:DL(marbles of red player) vs 1marble(of yellow player on position 44 end spot on board) - push successful,marbles 37:44 painted red and score of red player incremented by 1.
Move:21:30:UL(marbles of red player) vs 1marble(of yellow player on position 13 and position 6 empty) - push successful,marbles 13:21 painted red and marble 6 painted yellow.
Move:13:21:UL(marbles of red player) vs 1marble(of yellow player on position 6 end spot on board) - push successful,marbles 6:13 painted red and score of red player incremented by 1.
Move:32:24:UR(marbles of red player) vs 1marble(of yellow player on position 17 and position 11 empty) - push successful,marbles 17:24 painted red and marble 11 painted yellow.
Move:17:24:UR(marbles of red player) vs 1marble(of yellow player on position 11 end spot on board) - push successful,marbles 11:17 painted red and score of red player incremented by 1.
Sumito and push for 3 marbles vs 1 for(2,3 players).
Move:1:2:3:R(marbles of red player) vs 1marble(of yellow player on position 4 and position 5 empty) - push successful, marbles 2:3:4 painted red and marble 5 painted yellow.
Move:2:3:4:R(marbles of red player) vs 1marble(of yellow player on position 5 end position) - push successful, marbles 3:4:5 painted red and score of red player incremented by 1.
Move:3:4:5:L(marbles of red player) vs 1marble(of yellow player on position 2 and position 1 empty) - push successful, marbles 2:3:4 painted red and marble 1 painted yellow.

Move:2:3:4:L(marbles of red player) vs 1marble(of yellow player on position 1 end position) - push successful, marbles 1:2:3 painted red and score of red player incremented by 1.
Move:1:6:12:DL(marbles of red player) vs 1marble(of yellow player on position 19 and position 27 empty) - push successful, marbles 6:12:19 painted red and marble 27 painted yellow.
Move:6:12:19:DL(marbles of red player) vs 1marble(of yellow player on position 27 end position) - push successful, marbles 12:19:27 painted red and score of red player incremented by 1.
Move:5:11:18:DR(marbles of red player) vs 1marble(of yellow player on position 26 and position 35 empty) - push successful, marbles 11:18:26 painted red and marble 35 painted yellow.
Move:11:18:26:DR(marbles of red player) vs 1marble(of yellow player on position 35 end position) - push successful, marbles 18:26:35 painted red and score of red player incremented by 1.
Move:44:51:57:UL(marbles of red player) vs 1marble(of yellow player on position 36 and position 27 empty) - push successful, marbles 36:44:51 painted red and marble 27 painted yellow.
Move:36:44:51:UL(marbles of red player) vs 1marble(of yellow player on position 27 end position) - push successful, marbles 27:36:44 painted red and score of red player incremented by 1.
Move:61:56:50:UR(marbles of red player) vs 1marble(of yellow player on position 43 and position 35 empty) - push successful, marbles 43:50:56 painted red and marble 35 painted yellow.
Move:43:50:56:UR(marbles of red player) vs 1marble(of yellow player on position 35 end position) - push successful, marbles 35:43:50 painted red and score of red player incremented by 1.
Sumito and push for 3 marbles vs 2 for(2,3 players).
Move:1:2:3:R(marbles of red player)vs 2marbles(of yellow player on positions 4:5(end position)) - push successful,marbles 2:3:4 painted red marble 5 painted yellow and score of red player incremented by 1.
Move:3:4:5:L(marbles of red player)vs 2marbles(of yellow player on positions 1:2(end position)) - push successful,marbles 2:3:4 painted red marble 1 painted yellow and score of red player incremented by 1.
Move:29:38:46:UL(marbles of red player)vs 2marbles(of yellow player on positions 12:20(end position)) - push successful,marbles 20:29:38 painted red marble 12 painted yellow and score of red player incremented by 1.
Move:50:56:61:UR(marbles of red player)vs 2marbles(of yellow player on positions 43:35(end position)) - push successful,marbles 43:50:56 painted red marble 35 painted yellow and score of red player incremented by 1.
Move:23:32:41:DR(marbles of red player)vs 2marbles(of yellow player on positions 49:56(end position)) - push successful,marbles 35:41:49 painted red marble 56 painted yellow and score of red player incremented by 1.
Move:22:30:38:DL(marbles of red player)vs 2marbles(of yellow player on positions 45:51(end position)) - push successful,marbles 30:38:45 painted red marble 51 painted yellow and score of red player incremented by 1.

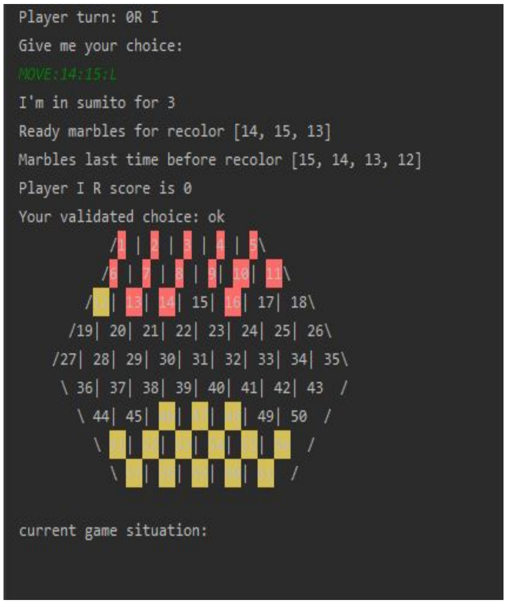
Sumito and push for 2 vs 1 marble(4 players)(Players red/green and yellow/pink are teammates)
Move:1:2:R(first marble of red player second of green) vs 1marble(of yellow player on position 3 and position 4 empty) - push successful,marbles 2:3 painted red and green marble 4 painted yellow.
Move:3:4:R(first marble of red player second of green) vs 1marble(of yellow player on position 5 end position on board) - push successful,marbles 4:5 painted red and green and score of red/green players incremented with 1.
Move:3:4:L(first marble of yellow player second of green) vs 1marble(of red player on position 2 and position 1 empty) - push successful,marbles 2:3 painted yellow/green marble 1 painted red.
Move:2:3:L(first marble of yellow player second of green) vs 1marble(of red player on position 1 end spot on board) - push successful,marbles 1:2 painted red/green marble and score of yellow/pink players incremented by 1.
Move:1:7:DR(first marble of red player second of green) vs 1marble(of yellow player on position 14 and position 22 empty) - push successful,marbles 7:14 painted red/green and marble 22 painted yellow.
Move:33:42:DR(first marble of red player second of green) vs 1marble(of yellow player on position 50 end spot on board) - push successful,marbles 42:50 painted red/green and score of red/green players incremented by 1.
Move:21:29:DL(first marble of red player second of green) vs 1marble(of yellow player on position 37 and position 44 empty) - push successful,marbles 29:37 painted red/green and marble 44 painted yellow.
Move:29:37:DL(first marble of red player second of green) vs 1marble(of yellow player on position 44 end spot on board) - push successful,marbles 37:44 painted red/green and score of red/green players incremented by 1.
Move:21:30:UL(first marble of red player second of green) vs 1marble(of yellow player on position 13 and position 6 empty) - push successful,marbles 13:21 painted red/green and marble 6 painted yellow.
Move:13:21:UL(first marble of red player second of green) vs 1marble(of yellow player on position 6 end spot on board) - push successful,marbles 6:13 painted red and score of red players incremented by 1.
Move:32:24:UR(marbles of red player) vs 1marble(of yellow player on position 17 and position 11 empty) - push successful,marbles 17:24 painted red/green and marble 11 painted yellow.
Move:17:24:UR(first marble of red player second of green) vs 1marble(of yellow player on position 11 end spot on board) - push successful,marbles 11:17 painted red/green and score of red players incremented by 1.
Sumito and push for 3 marbles vs 1 for(2,3 players).
Move:1:2:3:R(first marble of red player the other 2 of green) vs 1marble(of yellow player on position 4 and position 5 empty) - push successful, marbles 2:3:4 painted red/green and marble 5 painted yellow.
Move:2:3:4:R(first marble of red player the other 2 of green) vs 1marble(of yellow player on position 5 end position) - push successful, marbles 3:4:5 painted red/green and score of red/green player incremented by 1.

Move:3:4:5:L(first marble of red player the other 2 of green) vs 1marble(of yellow player on position 2 and position 1 empty) - push successful, marbles 2:3:4 painted red/green and marble 1 painted yellow.
Move:2:3:4:L(first marble of red player the other 2 of green) vs 1marble(of yellow player on position 1 end position) - push successful, marbles 1:2:3 painted red/green and score of red/green player incremented by 1.
Move:1:6:12:DL(first marble of red player the other 2 of green) vs 1marble(of yellow player on position 19 and position 27 empty) - push successful, marbles 6:12:19 painted red/green and marble 27 painted yellow.
Move:6:12:19:DL(first marble of red player the other 2 of green) vs 1marble(of yellow player on position 27 end position) - push successful, marbles 12:19:27 painted red/green and score of red/green player incremented by 1.
Move:5:11:18:DR(first marble of red player the other 2 of green) vs 1marble(of yellow player on position 26 and position 35 empty) - push successful, marbles 11:18:26 painted red/green and marble 35 painted yellow.
Move:11:18:26:DR(first marble of red player the other 2 of green) vs 1marble(of yellow player on position 35 end position) - push successful, marbles 18:26:35 painted red/green and score of red/green player incremented by 1.
Move:44:51:57:UL(first marble of red player the other 2 of green) vs 1marble(of yellow player on position 36 and position 27 empty) - push successful, marbles 36:44:51 painted red/green and marble 27 painted yellow.
Move:36:44:51:UL(first marble of red player the other 2 of green) vs 1marble(of yellow player on position 27 end position) - push successful, marbles 27:36:44 painted red/green and score of red/green player incremented by 1.
Move:61:56:50:UR(first marble of red player the other 2 of green) vs 1marble(of yellow player on position 43 and position 35 empty) - push successful, marbles 43:50:56 painted red/green and marble 35 painted yellow.
Move:43:50:56:UR(first marble of red player the other 2 of green) vs 1marble(of yellow player on position 35 end position) - push successful, marbles 35:43:50 painted red/green and score of red/green player incremented by 1.
Sumito and push for 3 marbles vs 2 for(2,3 players).
Move:1:2:3:R(first marble of red player the other 2 of green)vs 2marbles(first of yellow player second of pink on positions 4:5(end position)) - push successful,marbles 2:3:4 painted red/green marble 5 painted yellow and score of red/green player incremented by 1.
Move:3:4:5:L(first marble of red player the other 2 of green)vs 2marbles(first of yellow player second of pink on positions 1:2(end position)) - push successful,marbles 2:3:4 painted red/green marble 1 painted yellow and score of red/green player incremented by 1.
Move:29:38:46:UL(first marble of red player the other 2 of green)vs 2marbles(first of yellow player second of pink on positions 12:20(end position)) - push successful,marbles 20:29:38 painted red/green marble 12 painted yellow and score of red/green player incremented by 1.
Move:50:56:61:UR(first marble of red player the other 2 of greenr)vs 2marblesfirst of yellow player second of pink on positions 43:35(end position)) - push successful,marbles 43:50:56 painted red/green marble 35 painted yellow and score of red/green player incremented by 1.

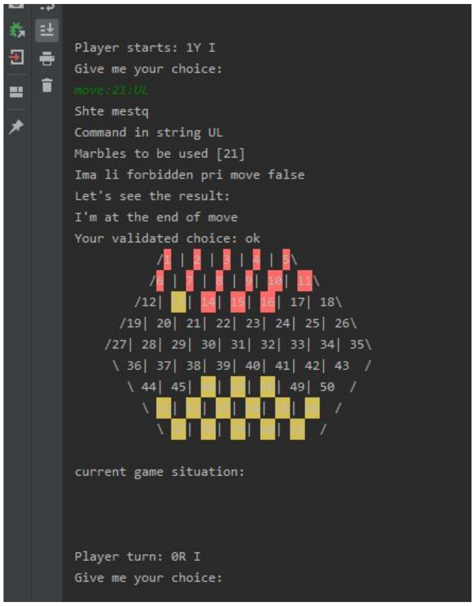
Move:23:32:41:DR(first marble of red player the other 2 of green)vs 2marbles(first of yellow player second of pink on positions 49:56(end position)) - push successful,marbles 35:41:49 painted red/green marble 56 painted yellow and score of red/green player incremented by 1.
Move:22:30:38:DL(first marble of red player the other 2 of green)vs 2marbles(first of yellow player second of pink on positions 45:51(end position)) - push successful,marbles 30:38:45 painted red/green marble 51 painted yellow and score of red/green player incremented by 1.
Here is some visual representation:



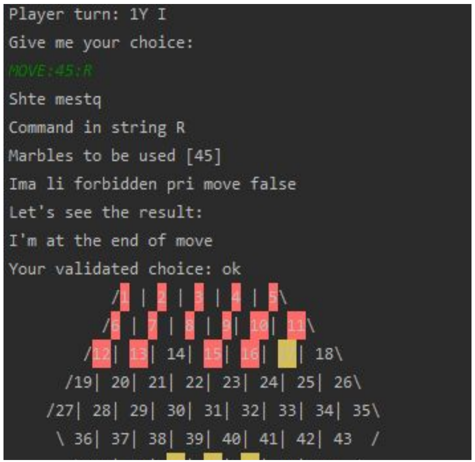
```
Player turn: 0R I
Give me your choice:
MOVE:13:14:L
I'm in sumito for 3
Ready marbles for recolor [13, 14, 12]
Marbles last time before recolor [14, 13, 12]
Player I R score is 1
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
         \ 51| 52| 53| 54| 55| 56  /
           \ 57| 58| 59| 60| 61  /

current game situation:
```



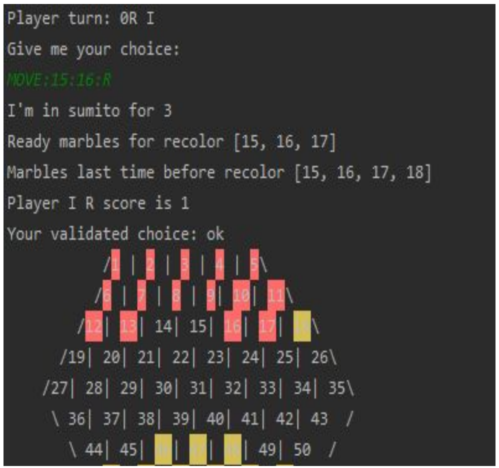
```
Player turn: 0R I
Give me your choice:
MOVE:14:15:L
I'm in sumito for 3
Ready marbles for recolor [14, 15, 13]
Marbles last time before recolor [15, 14, 13, 12]
Player I R score is 0
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
         \ 51| 52| 53| 54| 55| 56  /
           \ 57| 58| 59| 60| 61  /

current game situation:
```



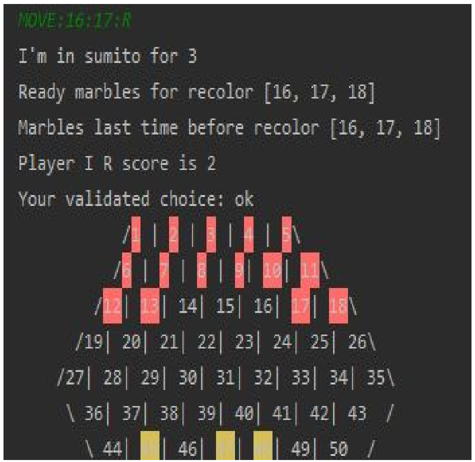
```
Player starts: 1Y I
Give me your choice:
move:21:UL
Shte mestq
Command in string UL
Marbles to be used [21]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
         \ 51| 52| 53| 54| 55| 56  /
           \ 57| 58| 59| 60| 61  /

current game situation:



Player turn: 0R I
Give me your choice:
```



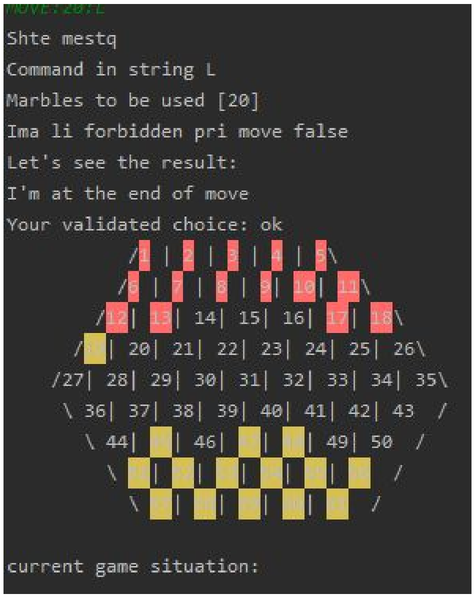
```
Player turn: 1Y I
Give me your choice:
MOVE:45:R
Shte mestq
Command in string R
Marbles to be used [45]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
```



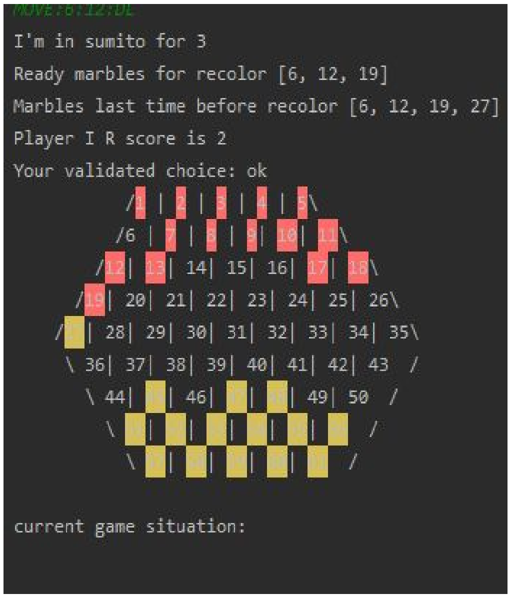
```
Player turn: 0R I
Give me your choice:
MOVE:15:16:R
I'm in sumito for 3
Ready marbles for recolor [15, 16, 17]
Marbles last time before recolor [15, 16, 17, 18]
Player I R score is 1
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
```



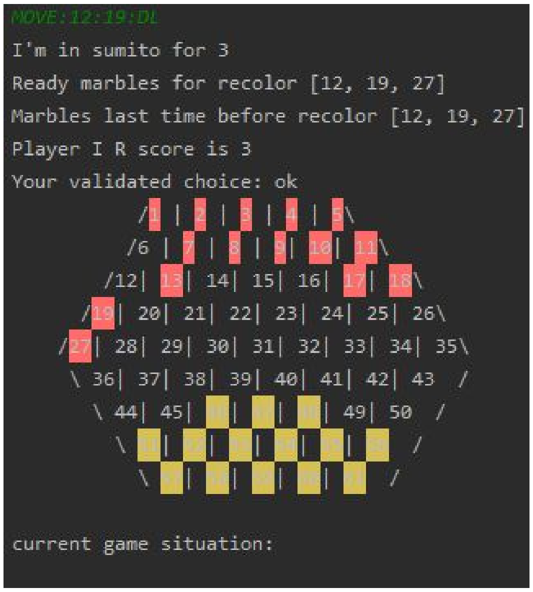
```
MOVE:16:17:R
I'm in sumito for 3
Ready marbles for recolor [16, 17, 18]
Marbles last time before recolor [16, 17, 18]
Player I R score is 2
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
```



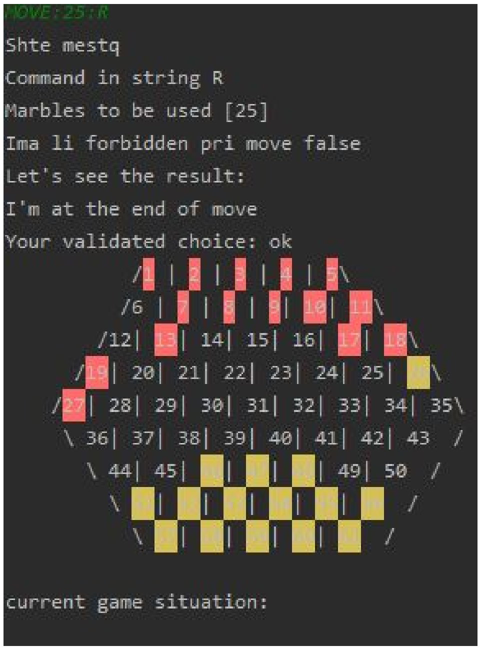
```
MOVE:20:L
Shte mestq
Command in string L
Marbles to be used [20]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
          /6 | 7 | 8 | 9| 10| 11\
        /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
         \ 51| 52| 53| 54| 55| 56  /
           \ 57| 58| 59| 60| 61  /

current game situation:
```



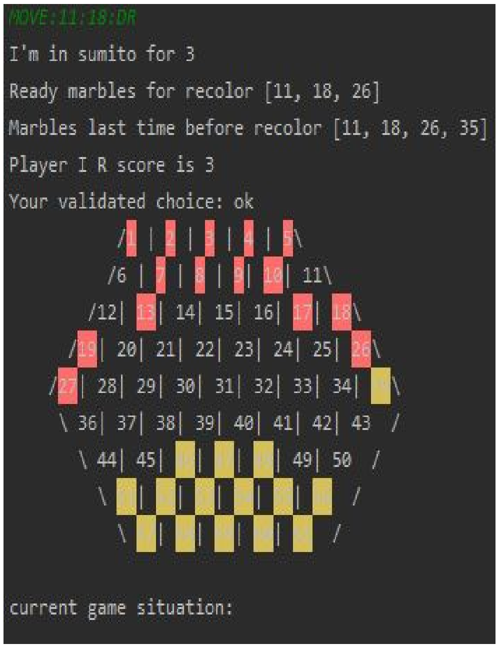
```
MOVE:6:12:DL
I'm in sumito for 3
Ready marbles for recolor [6, 12, 19]
Marbles last time before recolor [6, 12, 19, 27]
Player I R score is 2
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
          /6 | 7 | 8 | 9| 10| 11\
        /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
         \ 51| 52| 53| 54| 55| 56  /
           \ 57| 58| 59| 60| 61  /

current game situation:
```



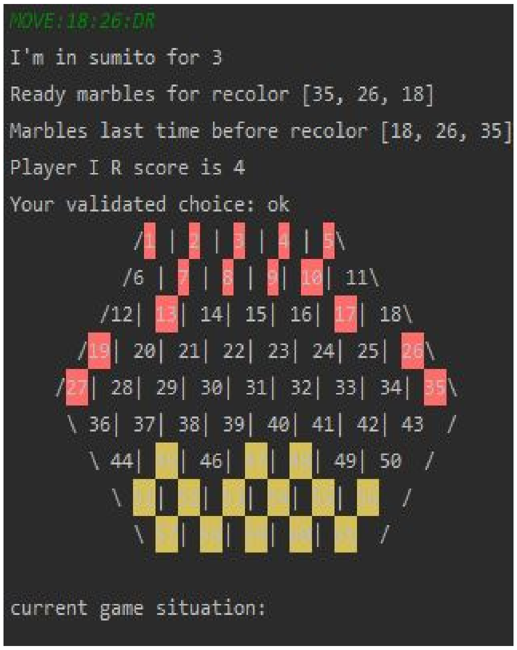
```
MOVE:12:19:DL
I'm in sumito for 3
Ready marbles for recolor [12, 19, 27]
Marbles last time before recolor [12, 19, 27]
Player I R score is 3
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
          /6 | 7 | 8 | 9| 10| 11\
        /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
         \ 51| 52| 53| 54| 55| 56  /
           \ 57| 58| 59| 60| 61  /

current game situation:
```



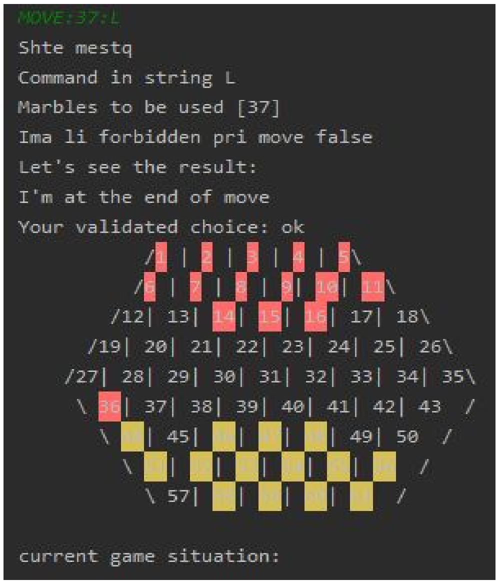
```
MOVE:25:R
Shte mestq
Command in string R
Marbles to be used [25]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
          /6 | 7 | 8 | 9| 10| 11\
        /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
         \ 51| 52| 53| 54| 55| 56  /
           \ 57| 58| 59| 60| 61  /

current game situation:
```



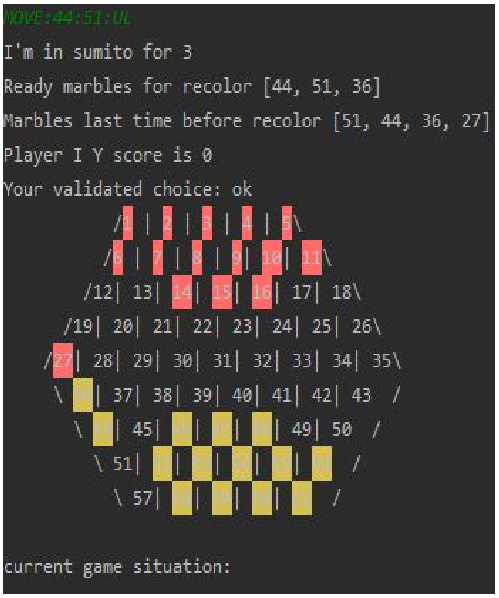
```
MOVE:11:18:DR
I'm in sumito for 3
Ready marbles for recolor [11, 18, 26]
Marbles last time before recolor [11, 18, 26, 35]
Player I R score is 3
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
          /6 | 7 | 8 | 9| 10| 11\
        /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
         \ 51| 52| 53| 54| 55| 56  /
           \ 57| 58| 59| 60| 61  /

current game situation:
```



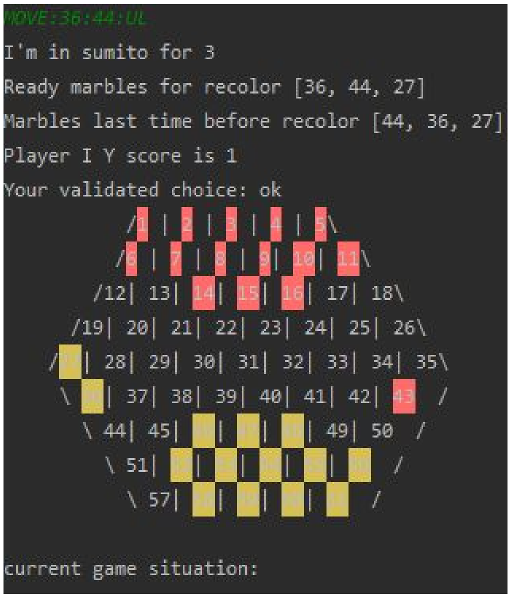
```
MOVE:18:26:DR
I'm in sumito for 3
Ready marbles for recolor [35, 26, 18]
Marbles last time before recolor [18, 26, 35]
Player I R score is 4
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
          /6 | 7 | 8 | 9| 10| 11\
        /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
         \ 51| 52| 53| 54| 55| 56  /
           \ 57| 58| 59| 60| 61  /

current game situation:
```



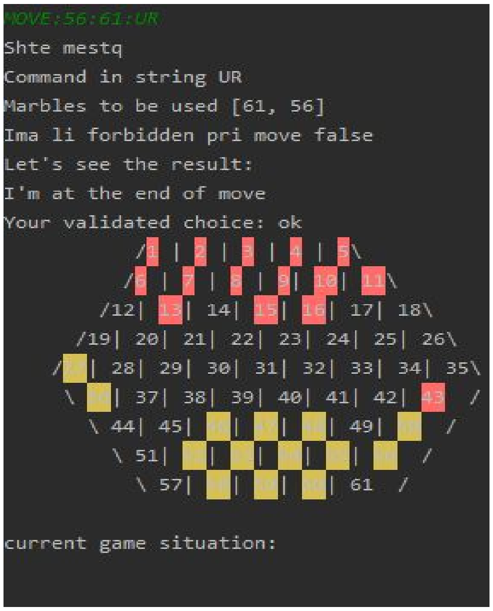
```
MOVE:37:L
Shte mestq
Command in string L
Marbles to be used [37]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
          /6 | 7 | 8 | 9| 10| 11\
        /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
         \ 51| 52| 53| 54| 55| 56  /
           \ 57| 58| 59| 60| 61  /

current game situation:
```



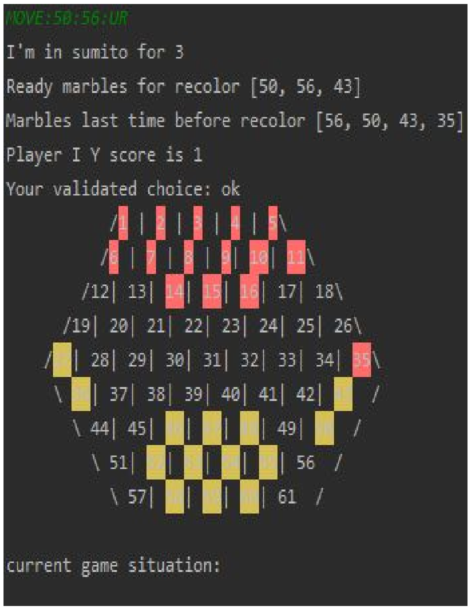
```
MOVE:44:51:UL
I'm in sumito for 3
Ready marbles for recolor [44, 51, 36]
Marbles last time before recolor [51, 44, 36, 27]
Player I Y score is 0
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
          /6 | 7 | 8 | 9| 10| 11\
        /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
         \ 51| 52| 53| 54| 55| 56  /
           \ 57| 58| 59| 60| 61  /

current game situation:
```



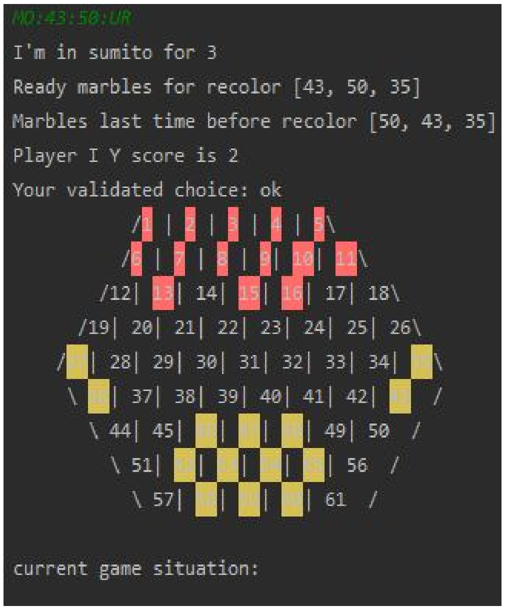
```
MOVE:36:44:UL
I'm in sumito for 3
Ready marbles for recolor [36, 44, 27]
Marbles last time before recolor [44, 36, 27]
Player I Y score is 1
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
          /6 | 7 | 8 | 9| 10| 11\
        /12| 13| 14| 15| 16| 17| 18\
      /19| 20| 21| 22| 23| 24| 25| 26\
    /27| 28| 29| 30| 31| 32| 33| 34| 35\
     \ 36| 37| 38| 39| 40| 41| 42| 43  /
       \ 44| 45| 46| 47| 48| 49| 50  /
         \ 51| 52| 53| 54| 55| 56  /
           \ 57| 58| 59| 60| 61  /

current game situation:
```



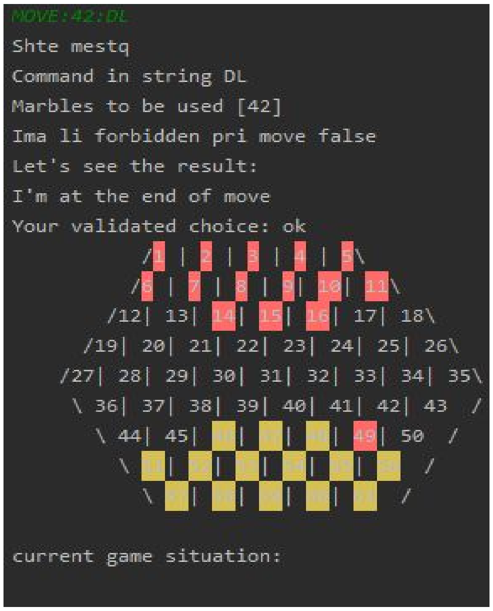
```
MOVE:56:61:UR
Shte mestq
Command in string UR
Marbles to be used [61, 56]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
         /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



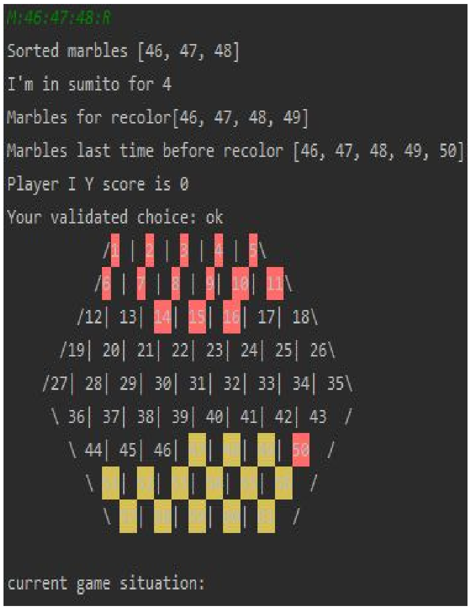
```
MOVE:50:56:UR
I'm in sumito for 3
Ready marbles for recolor [50, 56, 43]
Marbles last time before recolor [56, 50, 43, 35]
Player I Y score is 1
Your validated choice: ok
         /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



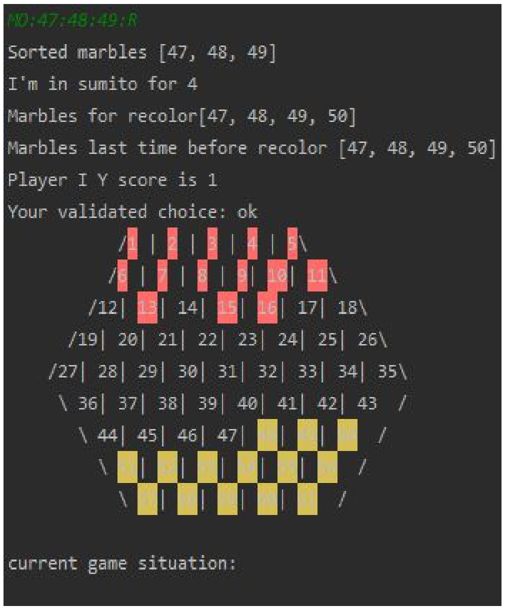
```
MO:43:50:UR
I'm in sumito for 3
Ready marbles for recolor [43, 50, 35]
Marbles last time before recolor [50, 43, 35]
Player I Y score is 2
Your validated choice: ok
         /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



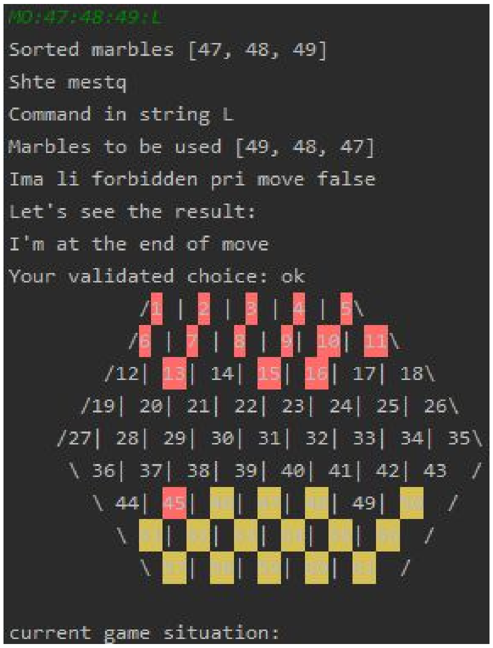
```
MOVE:42:DL
Shte mestq
Command in string DL
Marbles to be used [42]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
         /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



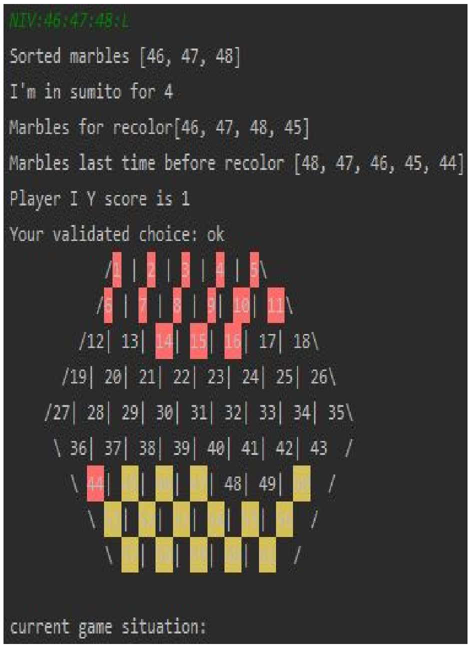
```
M:46:47:48:R
Sorted marbles [46, 47, 48]
I'm in sumito for 4
Marbles for recolor[46, 47, 48, 49]
Marbles last time before recolor [46, 47, 48, 49, 50]
Player I Y score is 0
Your validated choice: ok
         /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



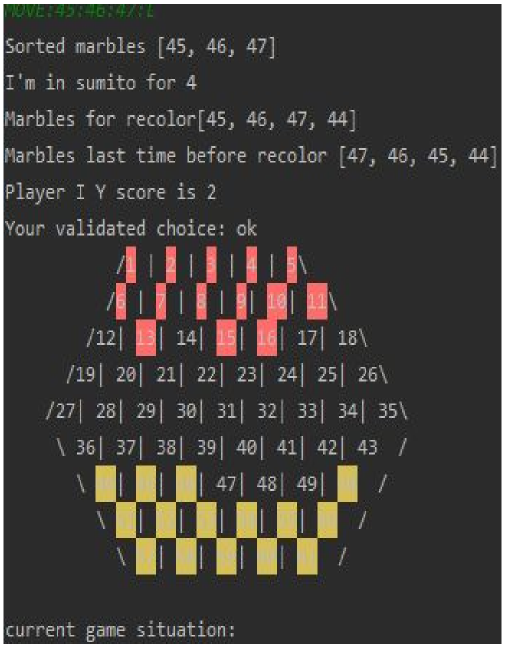
```
MO:47:48:49:R
Sorted marbles [47, 48, 49]
I'm in sumito for 4
Marbles for recolor[47, 48, 49, 50]
Marbles last time before recolor [47, 48, 49, 50]
Player I Y score is 1
Your validated choice: ok
         /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



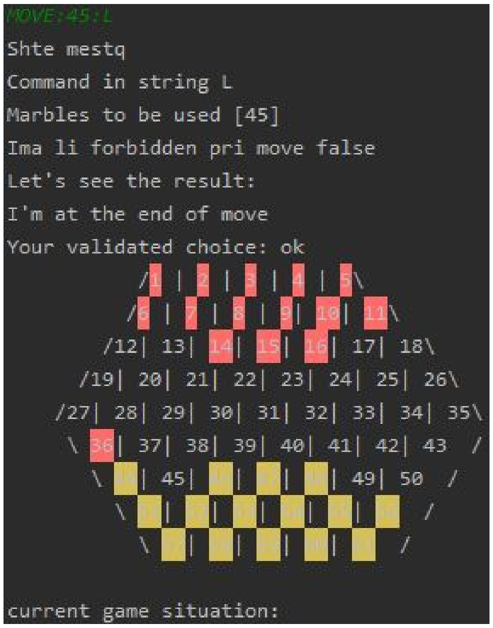
```
MO:47:48:49:L
Sorted marbles [47, 48, 49]
Shte mestq
Command in string L
Marbles to be used [49, 48, 47]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
         /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



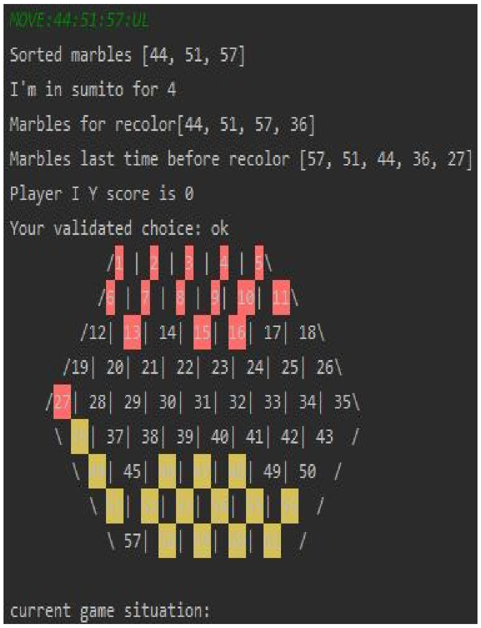
```
MV:46:47:48:L
Sorted marbles [46, 47, 48]
I'm in sumito for 4
Marbles for recolor[46, 47, 48, 45]
Marbles last time before recolor [48, 47, 46, 45, 44]
Player I Y score is 1
Your validated choice: ok
         /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



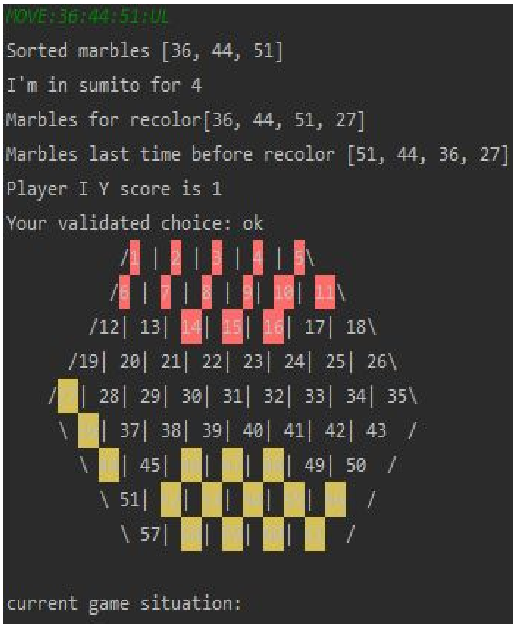
```
MOVE:45:46:47:L
Sorted marbles [45, 46, 47]
I'm in sumito for 4
Marbles for recolor[45, 46, 47, 44]
Marbles last time before recolor [47, 46, 45, 44]
Player I Y score is 2
Your validated choice: ok
         /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



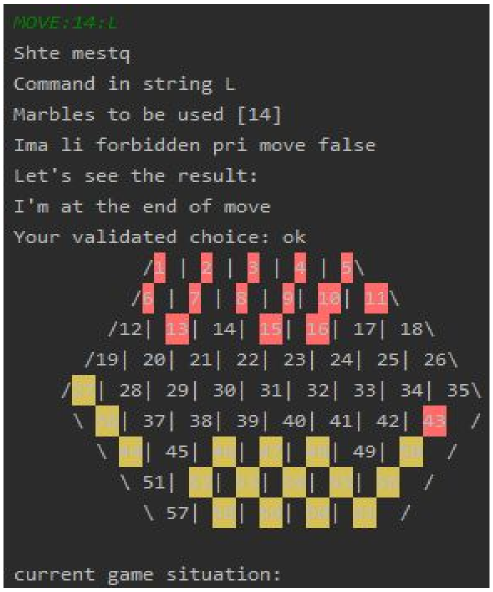
```
MOVE:45:L
Shte mestq
Command in string L
Marbles to be used [45]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
        /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



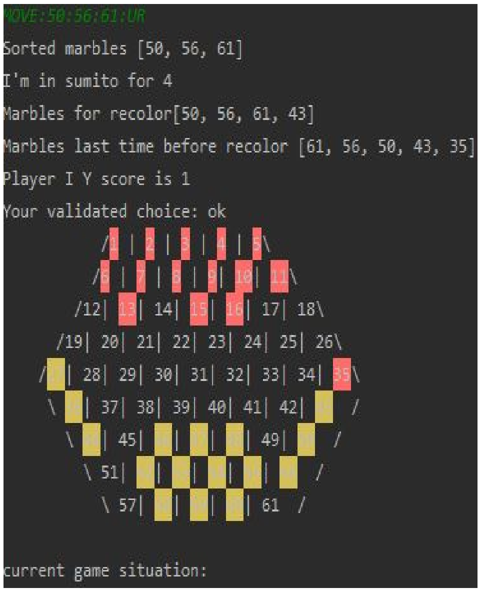
```
MOVE:44:51:57:UL
Sorted marbles [44, 51, 57]
I'm in sumito for 4
Marbles for recolor[44, 51, 57, 36]
Marbles last time before recolor [57, 51, 44, 36, 27]
Player I Y score is 0
Your validated choice: ok
        /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



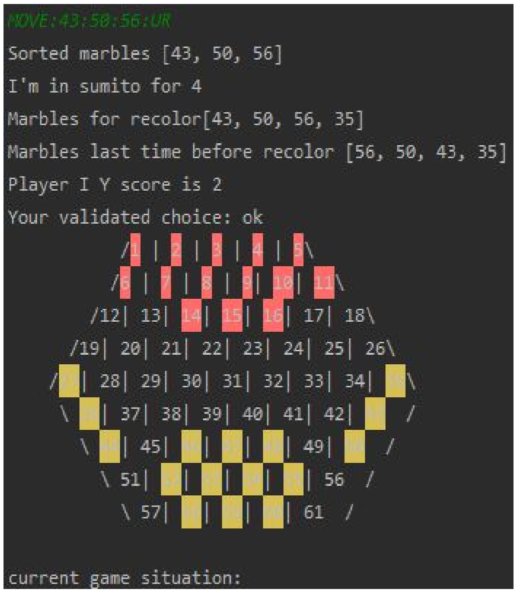
```
MOVE:36:44:51:UL
Sorted marbles [36, 44, 51]
I'm in sumito for 4
Marbles for recolor[36, 44, 51, 27]
Marbles last time before recolor [51, 44, 36, 27]
Player I Y score is 1
Your validated choice: ok
        /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



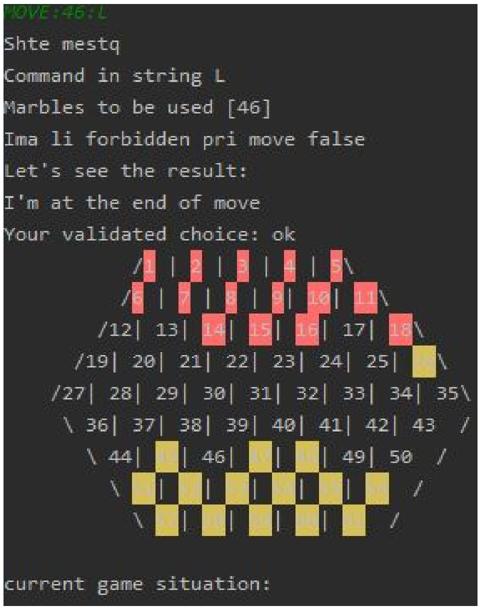
```
MOVE:14:L
Shte mestq
Command in string L
Marbles to be used [14]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
        /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



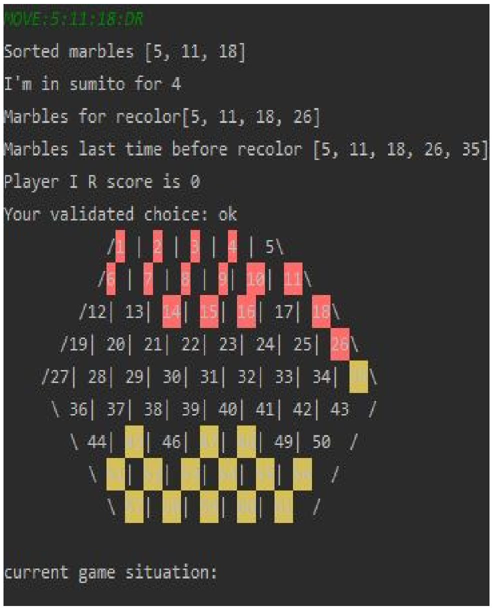
```
MOVE:50:56:61:UR
Sorted marbles [50, 56, 61]
I'm in sumito for 4
Marbles for recolor[50, 56, 61, 43]
Marbles last time before recolor [61, 56, 50, 43, 35]
Player I Y score is 1
Your validated choice: ok
        /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



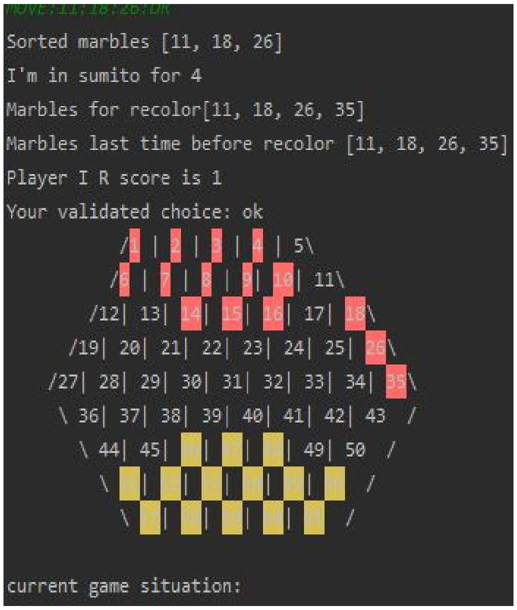
```
MOVE:43:50:56:UR
Sorted marbles [43, 50, 56]
I'm in sumito for 4
Marbles for recolor[43, 50, 56, 35]
Marbles last time before recolor [56, 50, 43, 35]
Player I Y score is 2
Your validated choice: ok
        /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



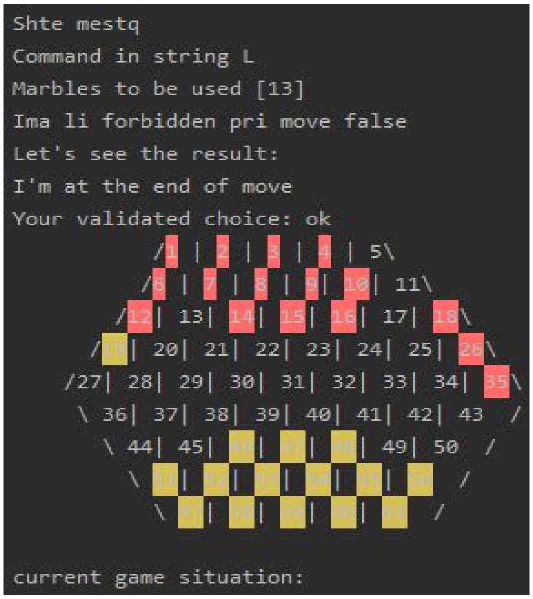
```
MOVE:46:L
Shte mestq
Command in string L
Marbles to be used [46]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
        /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



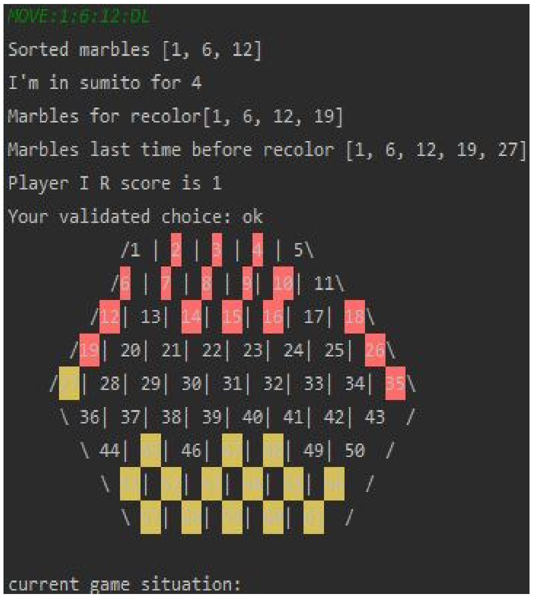
```
MOVE:5:11:18:DR
Sorted marbles [5, 11, 18]
I'm in sumito for 4
Marbles for recolor[5, 11, 18, 26]
Marbles last time before recolor [5, 11, 18, 26, 35]
Player I R score is 0
Your validated choice: ok
        /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



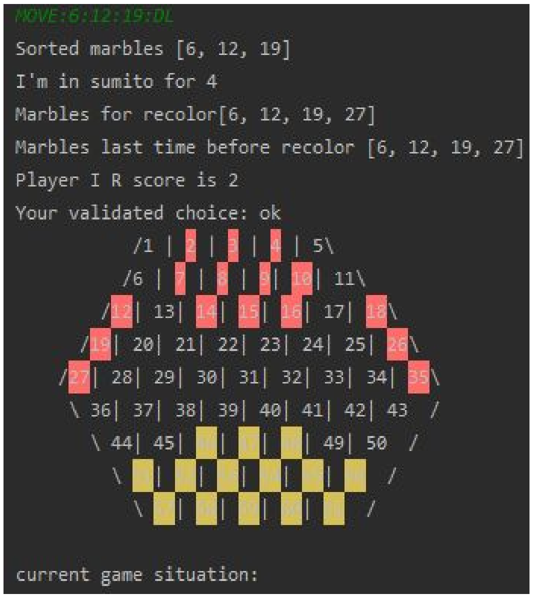
```
MOVE:11:18:26:DR
Sorted marbles [11, 18, 26]
I'm in sumito for 4
Marbles for recolor[11, 18, 26, 35]
Marbles last time before recolor [11, 18, 26, 35]
Player I R score is 1
Your validated choice: ok
        /1 | 2 | 3 | 4 | 5\
       /6 | 7 | 8 | 9| 10| 11\
     /12| 13| 14| 15| 16| 17| 18\
   /19| 20| 21| 22| 23| 24| 25| 26\
 /27| 28| 29| 30| 31| 32| 33| 34| 35\
  \ 36| 37| 38| 39| 40| 41| 42| 43  /
    \ 44| 45| 46| 47| 48| 49| 50  /
      \ 51| 52| 53| 54| 55| 56  /
        \ 57| 58| 59| 60| 61  /

current game situation:
```



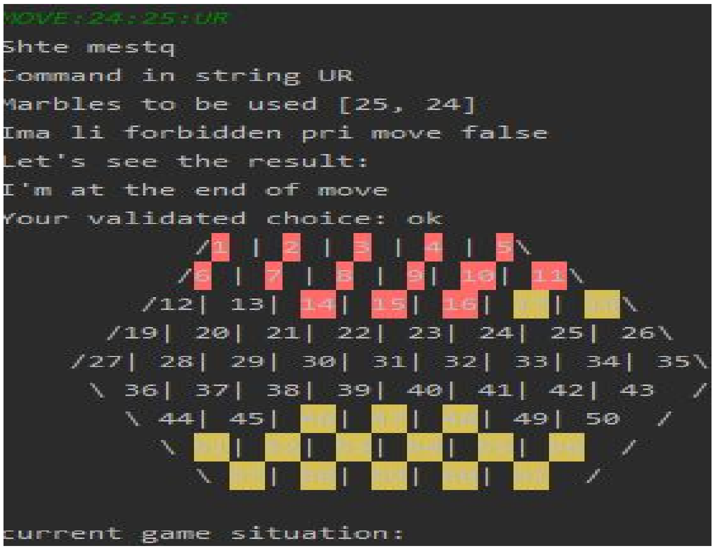
```
Shte mestq
Command in string L
Marbles to be used [13]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
     /19| 20| 21| 22| 23| 24| 25| 26\
   /27| 28| 29| 30| 31| 32| 33| 34| 35\
    \ 36| 37| 38| 39| 40| 41| 42| 43  /
      \ 44| 45| 46| 47| 48| 49| 50  /
        \ 51| 52| 53| 54| 55| 56  /
          \ 57| 58| 59| 60| 61  /

current game situation:
```



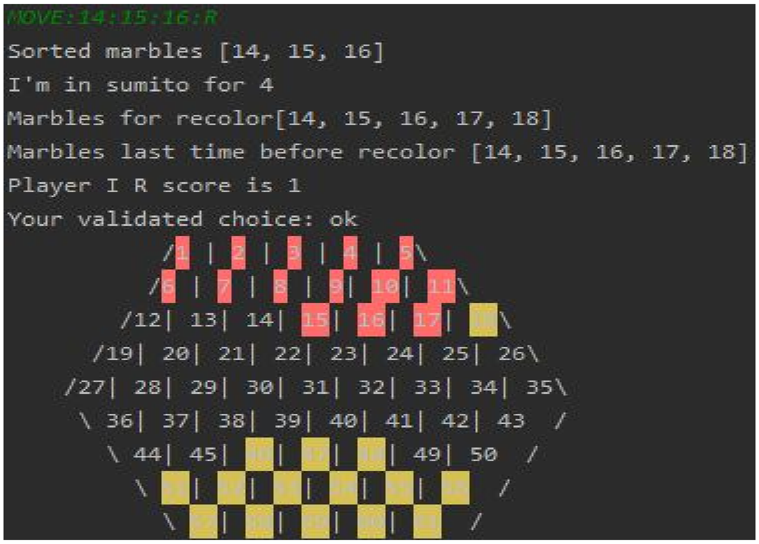
```
MOVE:1:6:12:DL
Sorted marbles [1, 6, 12]
I'm in sumito for 4
Marbles for recolor[1, 6, 12, 19]
Marbles last time before recolor [1, 6, 12, 19, 27]
Player I R score is 1
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
     /19| 20| 21| 22| 23| 24| 25| 26\
   /27| 28| 29| 30| 31| 32| 33| 34| 35\
    \ 36| 37| 38| 39| 40| 41| 42| 43  /
      \ 44| 45| 46| 47| 48| 49| 50  /
        \ 51| 52| 53| 54| 55| 56  /
          \ 57| 58| 59| 60| 61  /

current game situation:
```



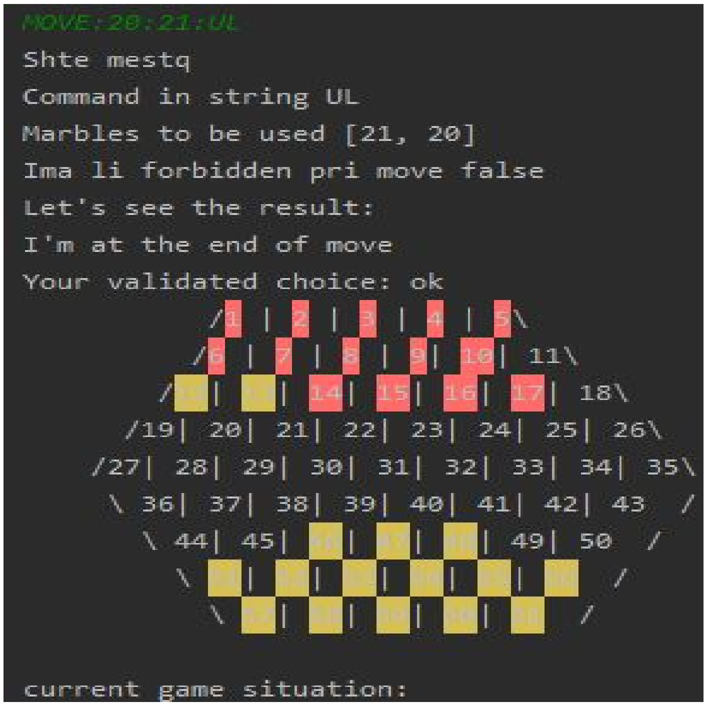
```
MOVE:6:12:19:DL
Sorted marbles [6, 12, 19]
I'm in sumito for 4
Marbles for recolor[6, 12, 19, 27]
Marbles last time before recolor [6, 12, 19, 27]
Player I R score is 2
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
     /19| 20| 21| 22| 23| 24| 25| 26\
   /27| 28| 29| 30| 31| 32| 33| 34| 35\
    \ 36| 37| 38| 39| 40| 41| 42| 43  /
      \ 44| 45| 46| 47| 48| 49| 50  /
        \ 51| 52| 53| 54| 55| 56  /
          \ 57| 58| 59| 60| 61  /

current game situation:
```



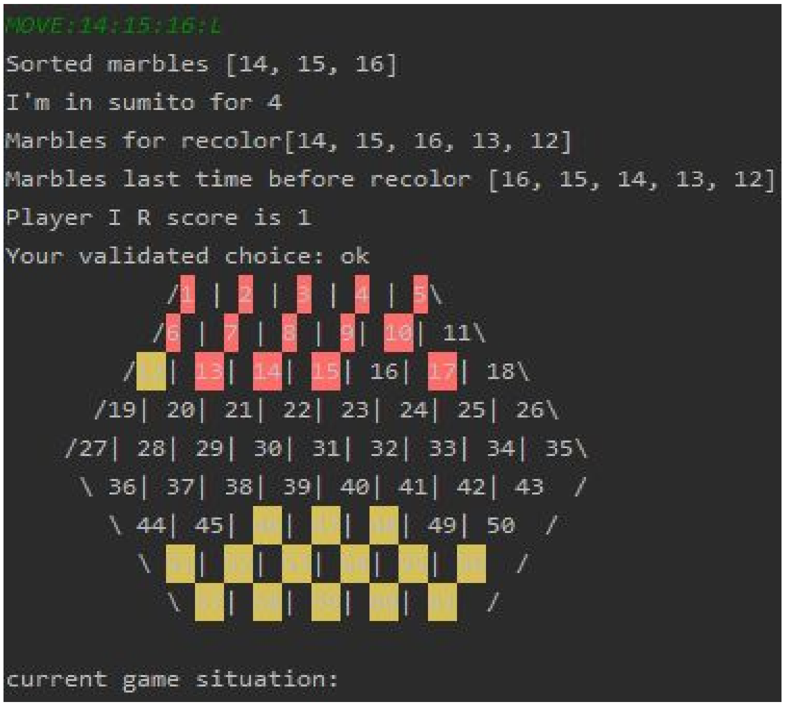
```
MOVE:24:25:UR
Shte mestq
Command in string UR
Marbles to be used [25, 24]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
     /19| 20| 21| 22| 23| 24| 25| 26\
   /27| 28| 29| 30| 31| 32| 33| 34| 35\
    \ 36| 37| 38| 39| 40| 41| 42| 43  /
      \ 44| 45| 46| 47| 48| 49| 50  /
        \ 51| 52| 53| 54| 55| 56  /
          \ 57| 58| 59| 60| 61  /

current game situation:
```



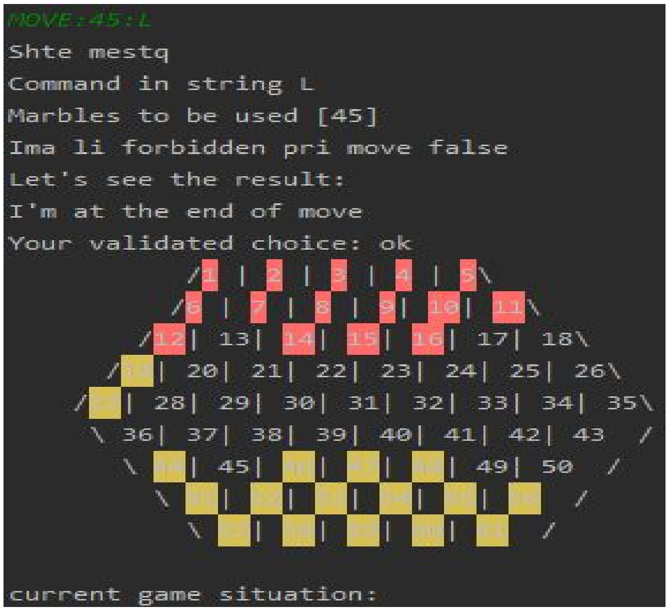
```
MOVE:14:15:16:R
Sorted marbles [14, 15, 16]
I'm in sumito for 4
Marbles for recolor[14, 15, 16, 17, 18]
Marbles last time before recolor [14, 15, 16, 17, 18]
Player I R score is 1
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
     /19| 20| 21| 22| 23| 24| 25| 26\
   /27| 28| 29| 30| 31| 32| 33| 34| 35\
    \ 36| 37| 38| 39| 40| 41| 42| 43  /
      \ 44| 45| 46| 47| 48| 49| 50  /
        \ 51| 52| 53| 54| 55| 56  /
          \ 57| 58| 59| 60| 61  /
```



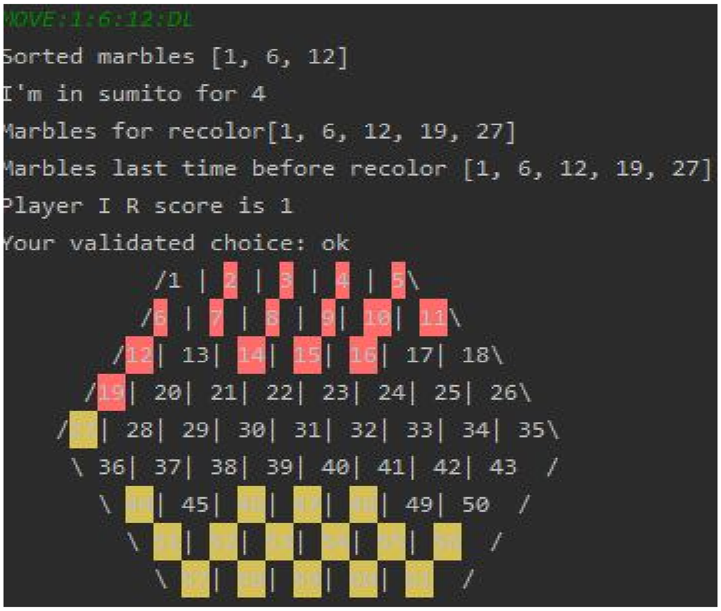
```
MOVE:20:21:UL
Shte mestq
Command in string UL
Marbles to be used [21, 20]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
     /19| 20| 21| 22| 23| 24| 25| 26\
   /27| 28| 29| 30| 31| 32| 33| 34| 35\
    \ 36| 37| 38| 39| 40| 41| 42| 43  /
      \ 44| 45| 46| 47| 48| 49| 50  /
        \ 51| 52| 53| 54| 55| 56  /
          \ 57| 58| 59| 60| 61  /

current game situation:
```



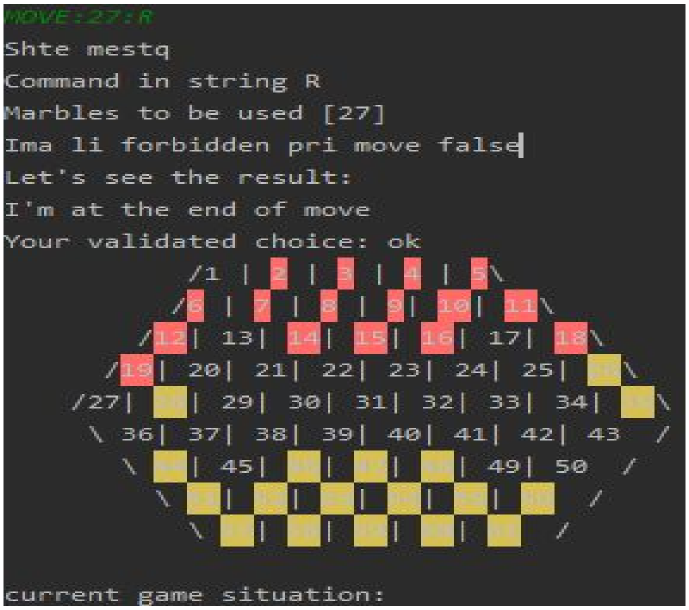
```
MOVE:14:15:16:L
Sorted marbles [14, 15, 16]
I'm in sumito for 4
Marbles for recolor[14, 15, 16, 13, 12]
Marbles last time before recolor [16, 15, 14, 13, 12]
Player I R score is 1
Your validated choice: ok
          /1 | 2 | 3 | 4 | 5\
         /6 | 7 | 8 | 9| 10| 11\
       /12| 13| 14| 15| 16| 17| 18\
     /19| 20| 21| 22| 23| 24| 25| 26\
   /27| 28| 29| 30| 31| 32| 33| 34| 35\
    \ 36| 37| 38| 39| 40| 41| 42| 43  /
      \ 44| 45| 46| 47| 48| 49| 50  /
        \ 51| 52| 53| 54| 55| 56  /
          \ 57| 58| 59| 60| 61  /

current game situation:
```



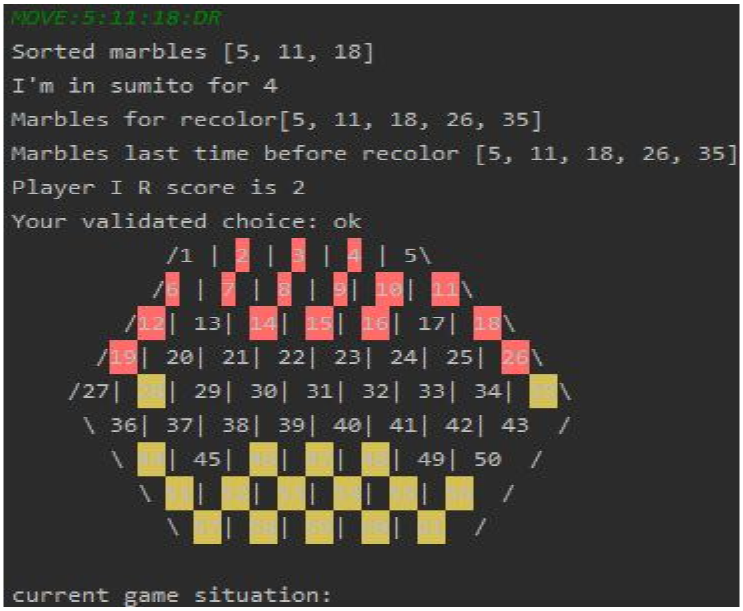
```
MOVE:45:L
Shte mestq
Command in string L
Marbles to be used [45]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
           /6 | 7 | 8 | 9| 10| 11\
         /12| 13| 14| 15| 16| 17| 18\
       /19| 20| 21| 22| 23| 24| 25| 26\
     /27| 28| 29| 30| 31| 32| 33| 34| 35\
      \ 36| 37| 38| 39| 40| 41| 42| 43  /
        \ 44| 45| 46| 47| 48| 49| 50  /
          \ 51| 52| 53| 54| 55| 56  /
            \ 57| 58| 59| 60| 61  /

current game situation:
```



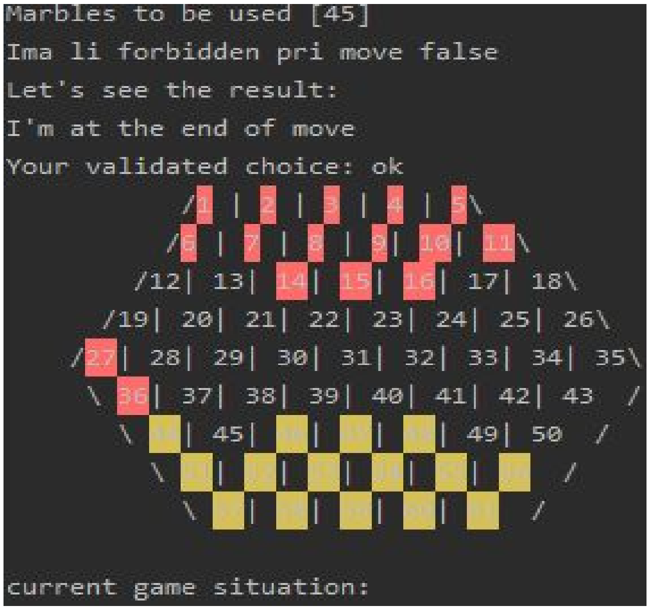
```
MOVE:1:6:12:DL
Sorted marbles [1, 6, 12]
I'm in sumito for 4
Marbles for recolor[1, 6, 12, 19, 27]
Marbles last time before recolor [1, 6, 12, 19, 27]
Player I R score is 1
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
           /6 | 7 | 8 | 9| 10| 11\
         /12| 13| 14| 15| 16| 17| 18\
       /19| 20| 21| 22| 23| 24| 25| 26\
     /27| 28| 29| 30| 31| 32| 33| 34| 35\
      \ 36| 37| 38| 39| 40| 41| 42| 43  /
        \ 44| 45| 46| 47| 48| 49| 50  /
          \ 51| 52| 53| 54| 55| 56  /
            \ 57| 58| 59| 60| 61  /
```



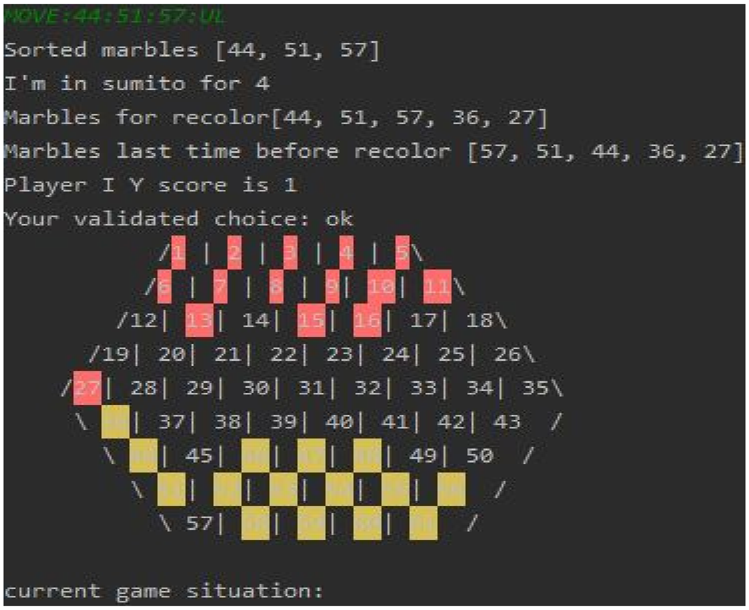
```
MOVE:27:R
Shte mestq
Command in string R
Marbles to be used [27]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
           /6 | 7 | 8 | 9| 10| 11\
         /12| 13| 14| 15| 16| 17| 18\
       /19| 20| 21| 22| 23| 24| 25| 26\
     /27| 28| 29| 30| 31| 32| 33| 34| 35\
      \ 36| 37| 38| 39| 40| 41| 42| 43  /
        \ 44| 45| 46| 47| 48| 49| 50  /
          \ 51| 52| 53| 54| 55| 56  /
            \ 57| 58| 59| 60| 61  /

current game situation:
```



```
MOVE:5:11:18:DR
Sorted marbles [5, 11, 18]
I'm in sumito for 4
Marbles for recolor[5, 11, 18, 26, 35]
Marbles last time before recolor [5, 11, 18, 26, 35]
Player I R score is 2
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
           /6 | 7 | 8 | 9| 10| 11\
         /12| 13| 14| 15| 16| 17| 18\
       /19| 20| 21| 22| 23| 24| 25| 26\
     /27| 28| 29| 30| 31| 32| 33| 34| 35\
      \ 36| 37| 38| 39| 40| 41| 42| 43  /
        \ 44| 45| 46| 47| 48| 49| 50  /
          \ 51| 52| 53| 54| 55| 56  /
            \ 57| 58| 59| 60| 61  /

current game situation:
```

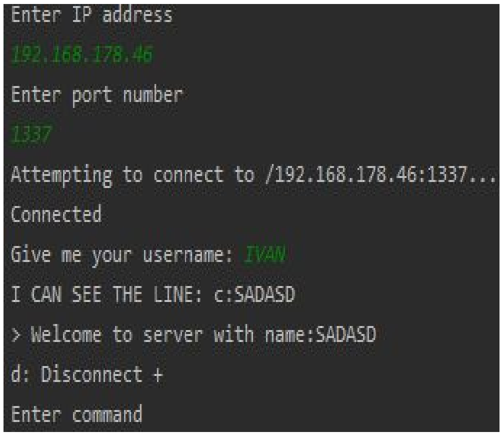


```
Marbles to be used [45]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
           /6 | 7 | 8 | 9| 10| 11\
         /12| 13| 14| 15| 16| 17| 18\
       /19| 20| 21| 22| 23| 24| 25| 26\
     /27| 28| 29| 30| 31| 32| 33| 34| 35\
      \ 36| 37| 38| 39| 40| 41| 42| 43  /
        \ 44| 45| 46| 47| 48| 49| 50  /
          \ 51| 52| 53| 54| 55| 56  /
            \ 57| 58| 59| 60| 61  /

current game situation:
```

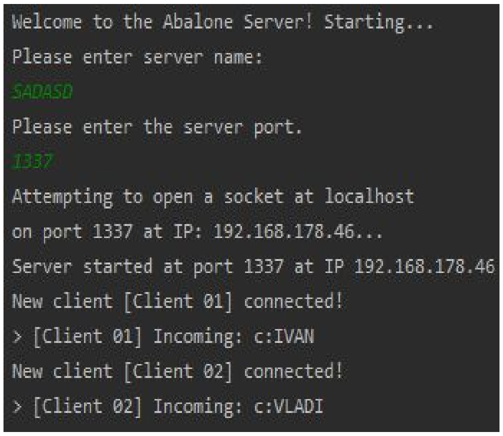


```
MOVE:44:51:57:UL
Sorted marbles [44, 51, 57]
I'm in sumito for 4
Marbles for recolor[44, 51, 57, 36, 27]
Marbles last time before recolor [57, 51, 44, 36, 27]
Player I Y score is 1
Your validated choice: ok
            /1 | 2 | 3 | 4 | 5\
           /6 | 7 | 8 | 9| 10| 11\
         /12| 13| 14| 15| 16| 17| 18\
       /19| 20| 21| 22| 23| 24| 25| 26\
     /27| 28| 29| 30| 31| 32| 33| 34| 35\
      \ 36| 37| 38| 39| 40| 41| 42| 43  /
        \ 44| 45| 46| 47| 48| 49| 50  /
          \ 51| 52| 53| 54| 55| 56  /
            \ 57| 58| 59| 60| 61  /

current game situation:
```

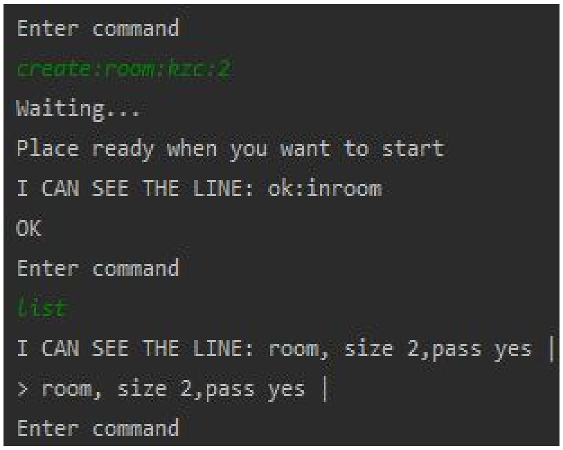
We were also told to do a visual representation of the tests for the servers because we could not write JUNIT test for them.So here are some of them :



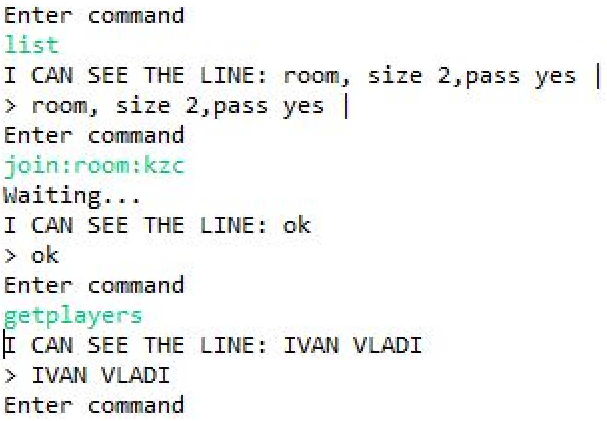
```
Enter IP address
192.168.178.46
Enter port number
1337
Attempting to connect to /192.168.178.46:1337...
Connected
Give me your username: IVAN
I CAN SEE THE LINE: c:SADASD
> Welcome to server with name:SADASD
d: Disconnect +
Enter command
```



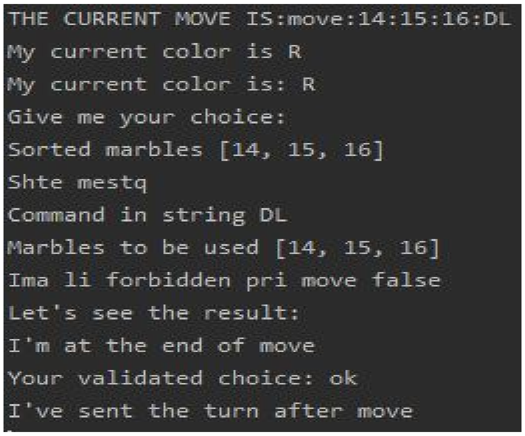
```
Welcome to the Abalone Server! Starting...
Please enter server name:
SADASD
Please enter the server port.
1337
Attempting to open a socket at localhost
on port 1337 at IP: 192.168.178.46...
Server started at port 1337 at IP 192.168.178.46
New client [Client 01] connected!
> [Client 01] Incoming: c:IVAN
New client [Client 02] connected!
> [Client 02] Incoming: c:VLADI
```



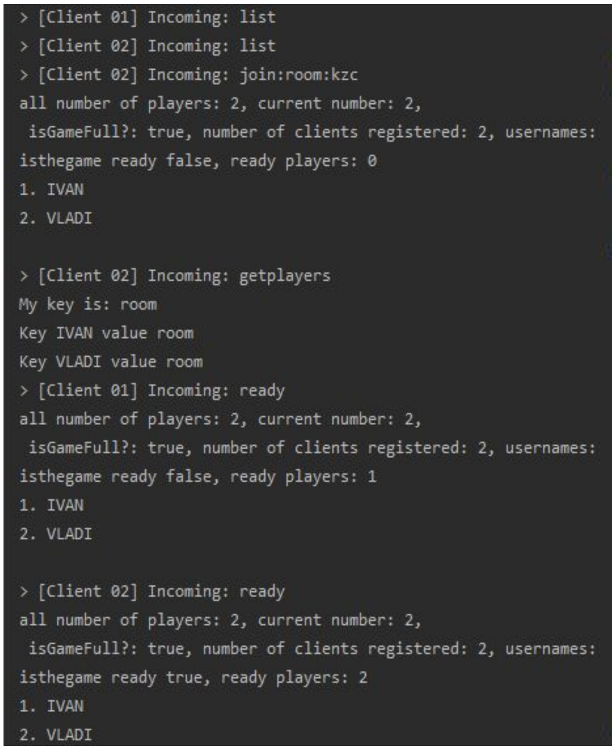
```
Enter command
create:room:kzc:2
Waiting...
Place ready when you want to start
I CAN SEE THE LINE: ok:inroom
OK
Enter command
list
I CAN SEE THE LINE: room, size 2,pass yes |
> room, size 2,pass yes |
Enter command
```



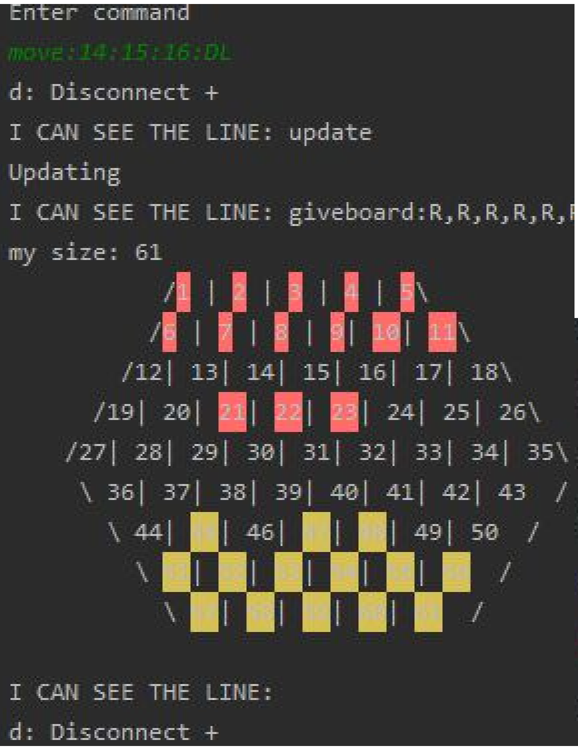
```
Enter command
list
I CAN SEE THE LINE: room, size 2,pass yes |
> room, size 2,pass yes |
Enter command
join:room:kzc
Waiting...
I CAN SEE THE LINE: ok
> ok
Enter command
getplayers
I CAN SEE THE LINE: IVAN VLADI
> IVAN VLADI
Enter command
```



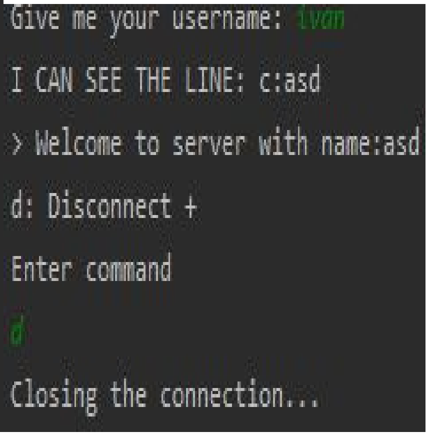
```
THE CURRENT MOVE IS:move:14:15:16:DL
My current color is R
My current color is: R
Give me your choice:
Sorted marbles [14, 15, 16]
Shte mestq
Command in string DL
Marbles to be used [14, 15, 16]
Ima li forbidden pri move false
Let's see the result:
I'm at the end of move
Your validated choice: ok
I've sent the turn after move
```



```
> [Client 01] Incoming: list
> [Client 02] Incoming: list
> [Client 02] Incoming: join:room:kzc
all number of players: 2, current number: 2,
 isGameFull?: true, number of clients registered: 2, usernames:
isthegame ready false, ready players: 0
1. IVAN
2. VLADI

> [Client 02] Incoming: getplayers
My key is: room
Key IVAN value room
Key VLADI value room
> [Client 01] Incoming: ready
all number of players: 2, current number: 2,
 isGameFull?: true, number of clients registered: 2, usernames:
isthegame ready false, ready players: 1
1. IVAN
2. VLADI

> [Client 02] Incoming: ready
all number of players: 2, current number: 2,
 isGameFull?: true, number of clients registered: 2, usernames:
isthegame ready true, ready players: 2
1. IVAN
2. VLADI
```



```
Enter command
move:14:15:16:DL
d: Disconnect +
I CAN SEE THE LINE: update
Updating
I CAN SEE THE LINE: giveboard:R,R,R,R,R,R
my size: 61
              /1 | 2 | 3 | 4 | 5\
             /6 | 7 | 8 | 9| 10| 11\
           /12| 13| 14| 15| 16| 17| 18\
          /19| 20| 21| 22| 23| 24| 25| 26\
        /27| 28| 29| 30| 31| 32| 33| 34| 35\
         \ 36| 37| 38| 39| 40| 41| 42| 43  /
          \ 44| 45| 46| 47| 48| 49| 50  /
           \ 51| 52| 53| 54| 55| 56  /
             \ 57| 58| 59| 60| 61  /

I CAN SEE THE LINE:
d: Disconnect +
```

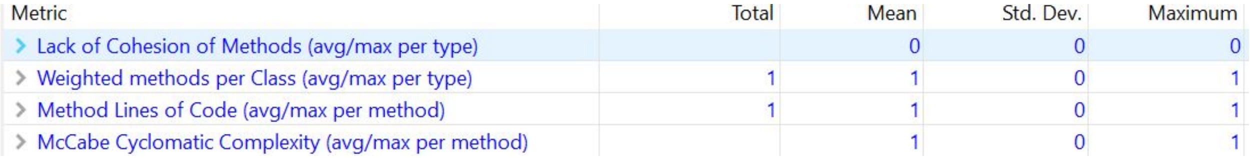


```
Give me your username: ivan
I CAN SEE THE LINE: c:asd
> Welcome to server with name:asd
d: Disconnect +
Enter command
d
Closing the connection...
```

Metrics report

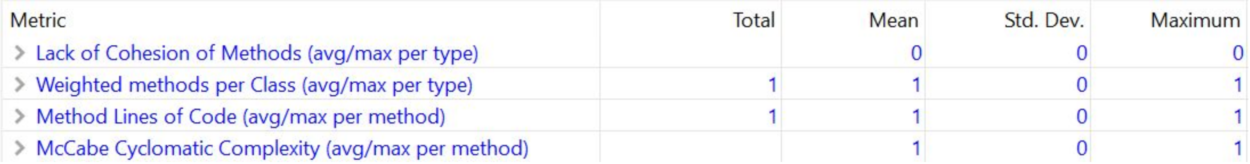
1.Exceptions package - nothing special

Capacity Overflow Exception



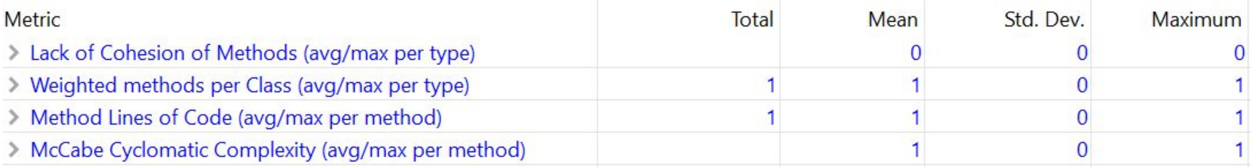
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0	0	0
Weighted methods per Class (avg/max per type)	1	1	0	1
Method Lines of Code (avg/max per method)	1	1	0	1
McCabe Cyclomatic Complexity (avg/max per method)		1	0	1

Exit program exception



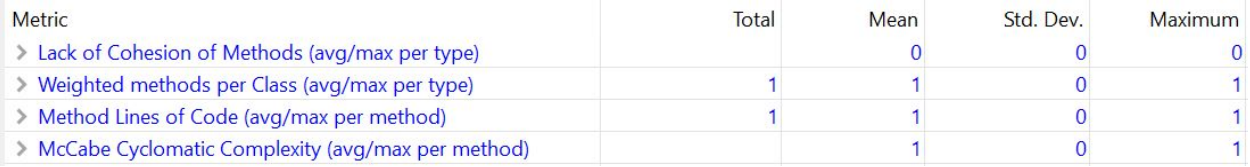
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0	0	0
Weighted methods per Class (avg/max per type)	1	1	0	1
Method Lines of Code (avg/max per method)	1	1	0	1
McCabe Cyclomatic Complexity (avg/max per method)		1	0	1

ProtocolException



Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0	0	0
Weighted methods per Class (avg/max per type)	1	1	0	1
Method Lines of Code (avg/max per method)	1	1	0	1
McCabe Cyclomatic Complexity (avg/max per method)		1	0	1

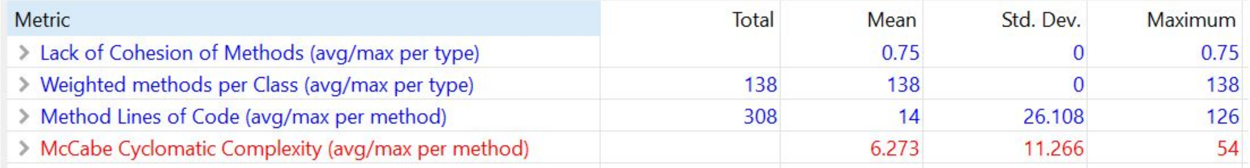
ServerUnavailableException



Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0	0	0
Weighted methods per Class (avg/max per type)	1	1	0	1
Method Lines of Code (avg/max per method)	1	1	0	1
McCabe Cyclomatic Complexity (avg/max per method)		1	0	1

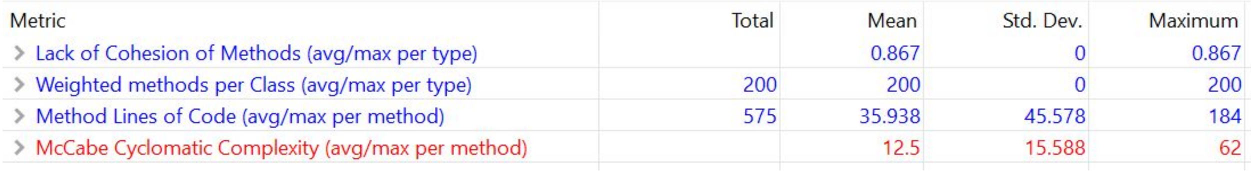
2. Board package

Board class - this is one of our most complex classes, because it has everything in order to manipulate, create, erase data on the board, and the complexity is up high because of our use of toString, but we couldn't figure an easier way.



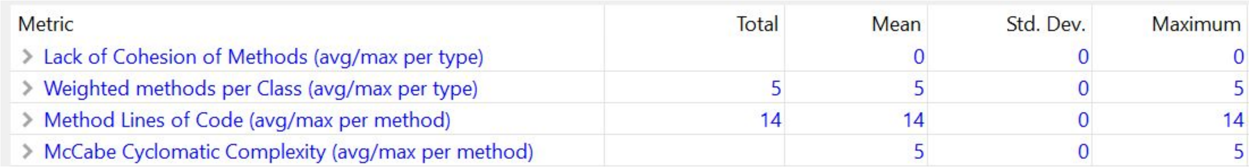
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0.75	0	0.75
Weighted methods per Class (avg/max per type)	138	138	0	138
Method Lines of Code (avg/max per method)	308	14	26.108	126
McCabe Cyclomatic Complexity (avg/max per method)		6.273	11.266	54

HumanPlayer class - our most complex class, because it is used to pass a String for the user's move and do a full validation of it, a full check if it's a sumito or normal move and execute the move on the board then return ok if it's done and "" if it was not successfull.



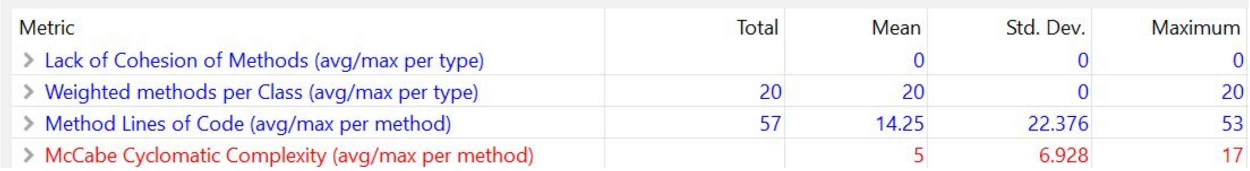
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0.867	0	0.867
Weighted methods per Class (avg/max per type)	200	200	0	200
Method Lines of Code (avg/max per method)	575	35.938	45.578	184
McCabe Cyclomatic Complexity (avg/max per method)		12.5	15.588	62

Mark enumerator - the enumerator we use to see the colors and their partner's



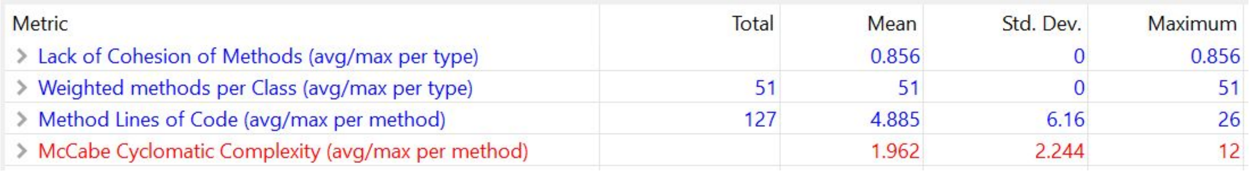
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0	0	0
Weighted methods per Class (avg/max per type)	5	5	0	5
Method Lines of Code (avg/max per method)	14	14	0	14
McCabe Cyclomatic Complexity (avg/max per method)		5	0	5

NaiveStrategy class - deprecated, because we didn't see fit in the final version, it's too easy



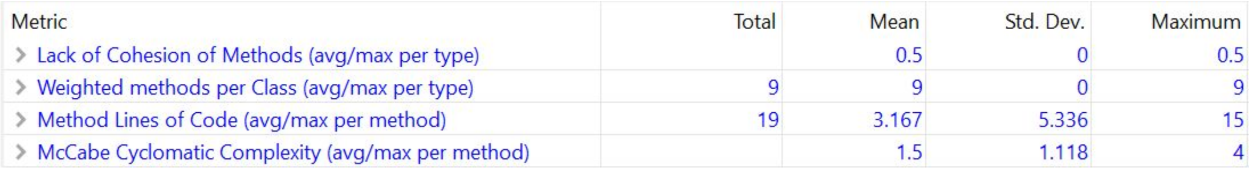
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0	0	0
Weighted methods per Class (avg/max per type)	20	20	0	20
Method Lines of Code (avg/max per method)	57	14.25	22.376	53
McCabe Cyclomatic Complexity (avg/max per method)		5	6.928	17

NetGame class - the class we use in order to connect the server game with the clients move, it has a special modified constructor to fit the need of assigning colors to the players as they log in



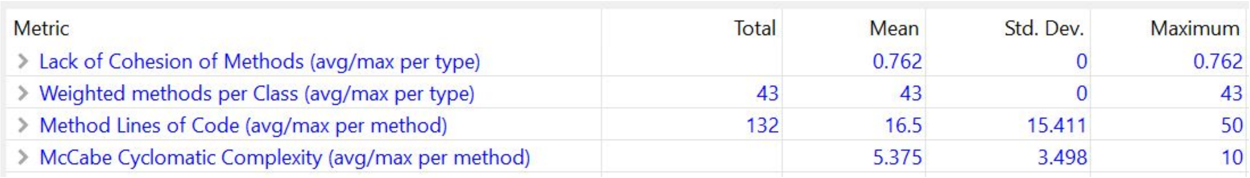
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0.856	0	0.856
Weighted methods per Class (avg/max per type)	51	51	0	51
Method Lines of Code (avg/max per method)	127	4.885	6.16	26
McCabe Cyclomatic Complexity (avg/max per method)		1.962	2.244	12

Player - abstract class we use as base for the human player



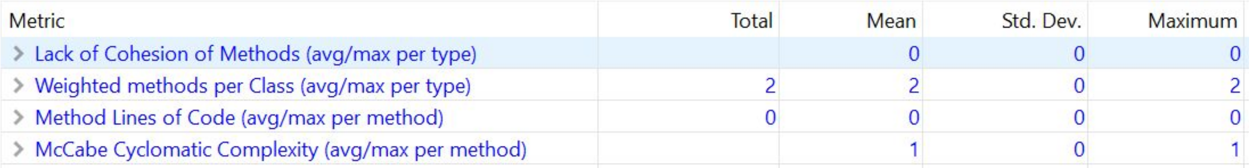
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0.5	0	0.5
Weighted methods per Class (avg/max per type)	9	9	0	9
Method Lines of Code (avg/max per method)	19	3.167	5.336	15
McCabe Cyclomatic Complexity (avg/max per method)		1.5	1.118	4

SmartStrategy - we use it to determine a move for the ComputerPlayer



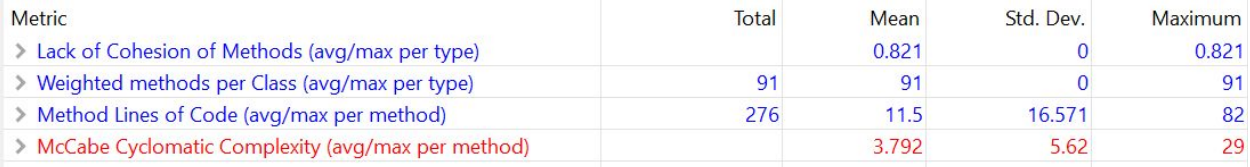
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0.762	0	0.762
Weighted methods per Class (avg/max per type)	43	43	0	43
Method Lines of Code (avg/max per method)	132	16.5	15.411	50
McCabe Cyclomatic Complexity (avg/max per method)		5.375	3.498	10

Interface Strategy



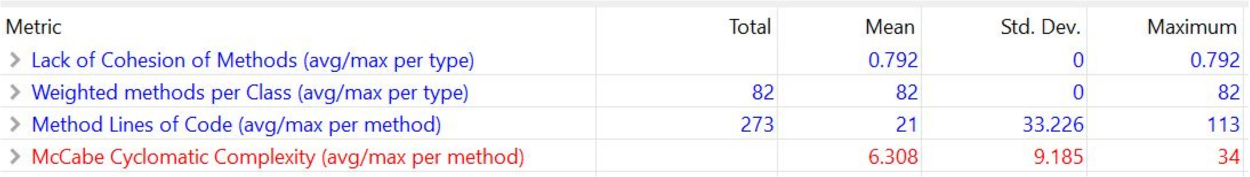
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0	0	0
Weighted methods per Class (avg/max per type)	2	2	0	2
Method Lines of Code (avg/max per method)	0	0	0	0
McCabe Cyclomatic Complexity (avg/max per method)		1	0	1

AbaloneClient - a client we use to establish a connection between a player and the server, also executing commands as sent by the TUI class



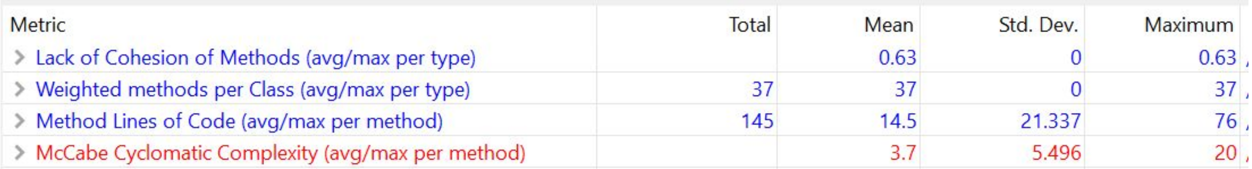
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0.821	0	0.821
Weighted methods per Class (avg/max per type)	91	91	0	91
Method Lines of Code (avg/max per method)	276	11.5	16.571	82
McCabe Cyclomatic Complexity (avg/max per method)		3.792	5.62	29

AbaloneClientTUI - a TUI we use to show menus and handle as given by the user or the server, validate commands given by the user and execute the appropriate commands in an appropriate way



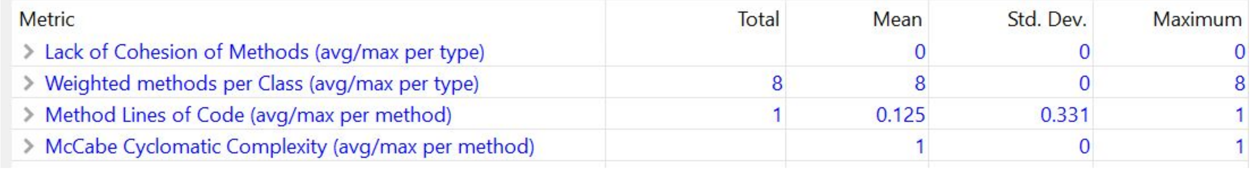
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0.792	0	0.792
Weighted methods per Class (avg/max per type)	82	82	0	82
Method Lines of Code (avg/max per method)	273	21	33.226	113
McCabe Cyclomatic Complexity (avg/max per method)		6.308	9.185	34

AbaloneClientHandler - the class used to handle communication between the server and a client, a thread is created for every client connected



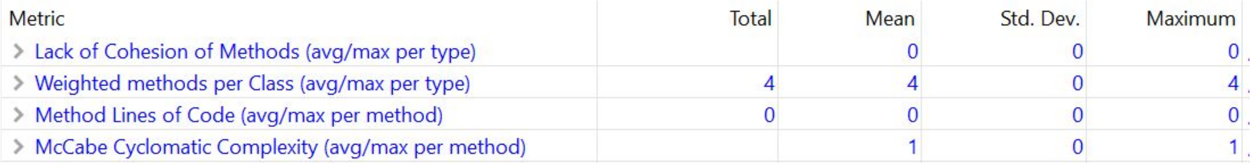
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0.63	0	0.63
Weighted methods per Class (avg/max per type)	37	37	0	37
Method Lines of Code (avg/max per method)	145	14.5	21.337	76
McCabe Cyclomatic Complexity (avg/max per method)		3.7	5.496	20

AbaloneClientView - interface for AbaloneClient



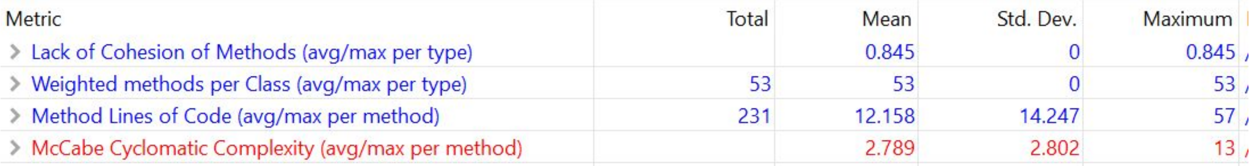
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0	0	0
Weighted methods per Class (avg/max per type)	8	8	0	8
Method Lines of Code (avg/max per method)	1	0.125	0.331	1
McCabe Cyclomatic Complexity (avg/max per method)		1	0	1

AbaloneServerView - interface for the AbaloneServer



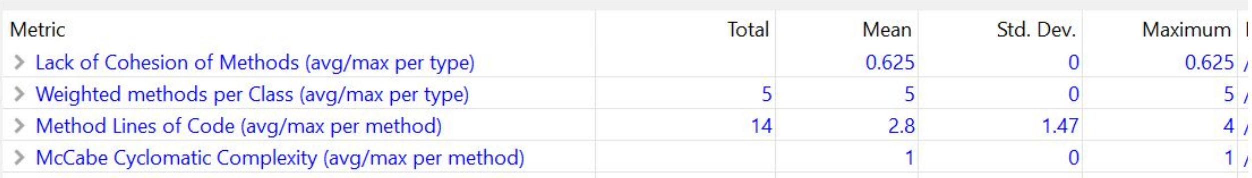
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0	0	0
Weighted methods per Class (avg/max per type)	4	4	0	4
Method Lines of Code (avg/max per method)	0	0	0	0
McCabe Cyclomatic Complexity (avg/max per method)		1	0	1

AbaloneServer - a class for the creation of the server and listener in order to connect user and create a thread(AbaloneClientHandler) to handle commands. Also the synchronized methods execution is done on server



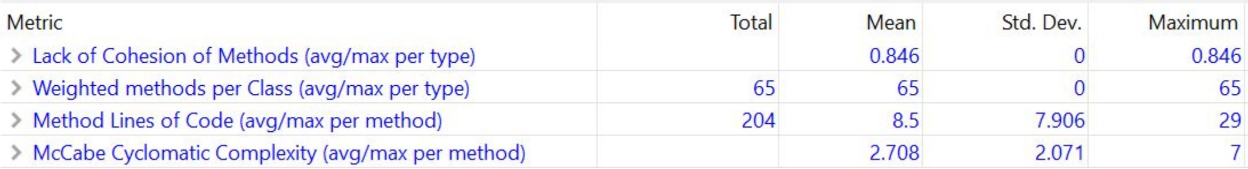
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0.845	0	0.845
Weighted methods per Class (avg/max per type)	53	53	0	53
Method Lines of Code (avg/max per method)	231	12.158	14.247	57
McCabe Cyclomatic Complexity (avg/max per method)		2.789	2.802	13

AbaloneServerTUI - simple TUI class to complement the Server



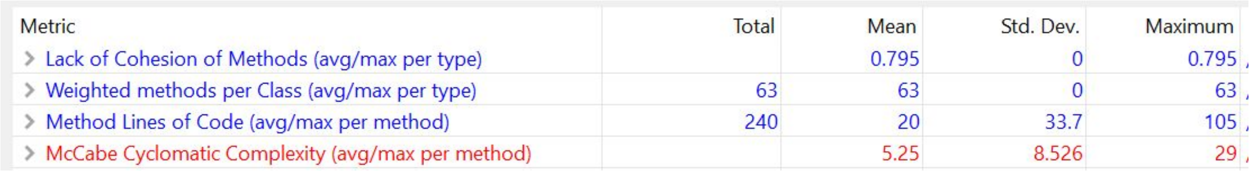
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0.625	0	0.625
Weighted methods per Class (avg/max per type)	5	5	0	5
Method Lines of Code (avg/max per method)	14	2.8	1.47	4
McCabe Cyclomatic Complexity (avg/max per method)		1	0	1

ComputerAbalonePlayer - class to create communication between a ComputerPlayer and server(identical to AbaloneClient)



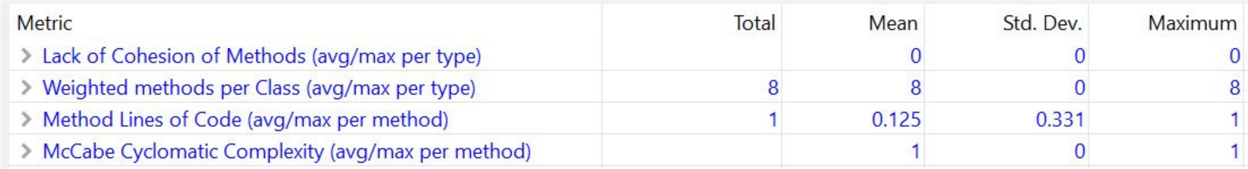
Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0.846	0	0.846
Weighted methods per Class (avg/max per type)	65	65	0	65
Method Lines of Code (avg/max per method)	204	8.5	7.906	29
McCabe Cyclomatic Complexity (avg/max per method)		2.708	2.071	7

ComputerAbaloneClientTUI - class to handle special way of communication between the computer player and the server, always sends a move



Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0.795	0	0.795
Weighted methods per Class (avg/max per type)	63	63	0	63
Method Lines of Code (avg/max per method)	240	20	33.7	105
McCabe Cyclomatic Complexity (avg/max per method)		5.25	8.526	29

ComputerPlayerAbalone - simple interface



Metric	Total	Mean	Std. Dev.	Maximum
Lack of Cohesion of Methods (avg/max per type)		0	0	0
Weighted methods per Class (avg/max per type)	8	8	0	8
Method Lines of Code (avg/max per method)	1	0.125	0.331	1
McCabe Cyclomatic Complexity (avg/max per method)		1	0	1

ProtocolMessages1 - the messages we used

Reflection on planning:
1- So in the Design project we were 4 people and we would every day work for 4/5 hours and we wanted to finish it as early as possible which we did. We wanted to do the same for the programming project work until we want to without burning out on it.
2 - So in the beginning the planning was going as planned until some people from our group decided to make changes on the protocol that we agreed on.They would make changes absolutely every day without asking anyone else and making it more suitable for their own code. We had to rechange our code every single time according to their changes which messed our planning a lot. We were planning to finish by tuesday 28.01 and do all the documentation on wednesday but with all the changes that were happening to the protocol we were not able to do so.
3 - The countermeasure that we took is that every day for the past 4 days(26/27/28/29) we have been staying up until 5am so that we could cope with all the changes and implement all the stuff that some people from our group decided to add to the protocol without asking anyone.
4 - The thing that we learned from this experience is that we should be ready to work as much as possible and next time to not let people who do not know what is going on with the protocol to touch it because they end up making some stuff up that is impossible to be implemented or the time which is given for the project is not enough to implement everything.
5 - First to do is to talk out the protocol on the first day and no more changes but if there is a change the change should be talked with absolutely everyone and voted and if a change is made is should be 1 and a half weeks before the deadline if it is after that time no changes should be made.The second to do is make everything as simple as possible and work together with your group mate on everything. The two don'ts are to not divide the work but better work on everything together and the second not to do is not to think that things are easy because sometimes there could be complications.

https://drive.google.com/open?id=1z8VNQ9-zaVRAVhZGP2q3V6N4dTtjI1MA - link to screenshots of our tutorial group members conversations