

# Guidelines for programming assignment #1

INFO-0010-2/4

2017-2018

# Context

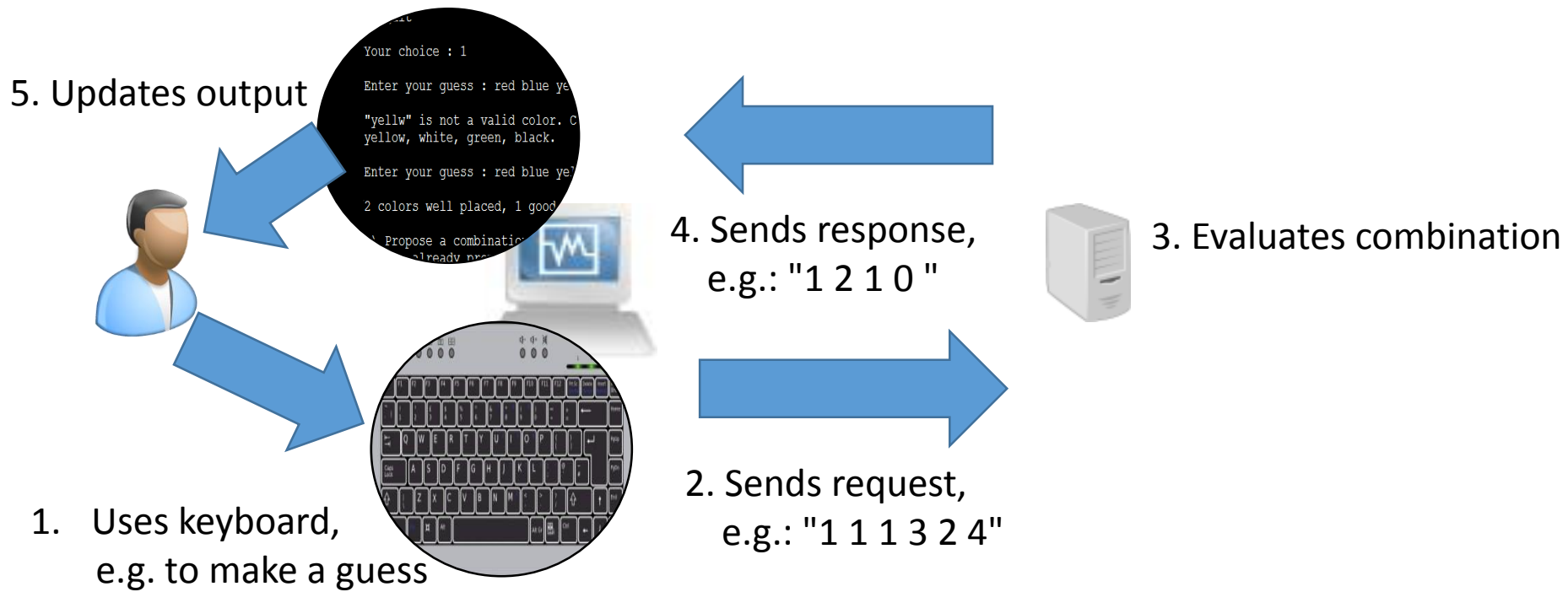
- You will develop a client/server application in order to play Mastermind.
- Java (1.8) Sockets.
- Console input/output (no fancy GUI).
- Imposed protocol.
- To be realized alone.
- **HARD deadline** : March, 25th, 2018



# Game rules

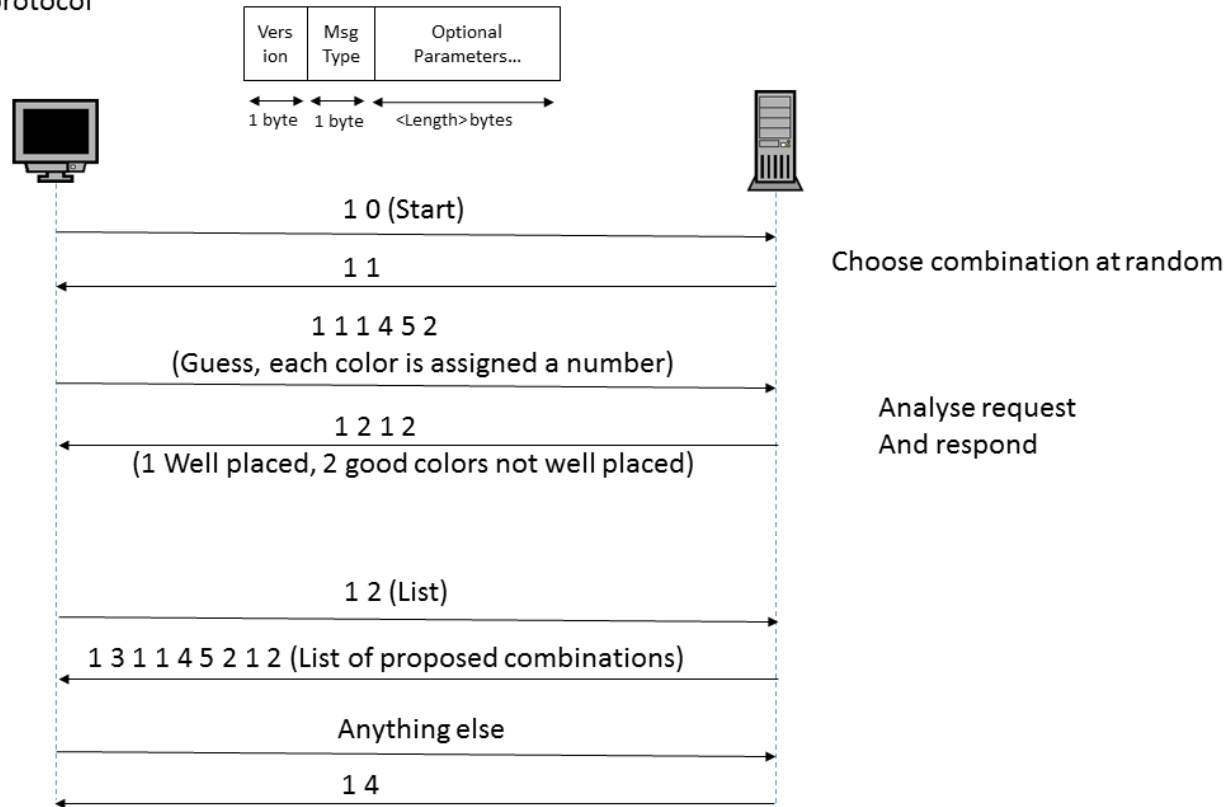
1. A combination of colors is selected at random (for instance, red blue red green).
2. User tries to guess the combination, and provides one to be evaluated
3. The response is the number of correctly placed colors and the number of incorrectly placed good colors
4. If the user guessed the combination correctly or failed 12 times, go to 5, otherwise go to 2.
5. Game over. User wins if the proposed combination is correct, he loses otherwise. User wants a new game? Go to 1, otherwise, quit.

# Architecture

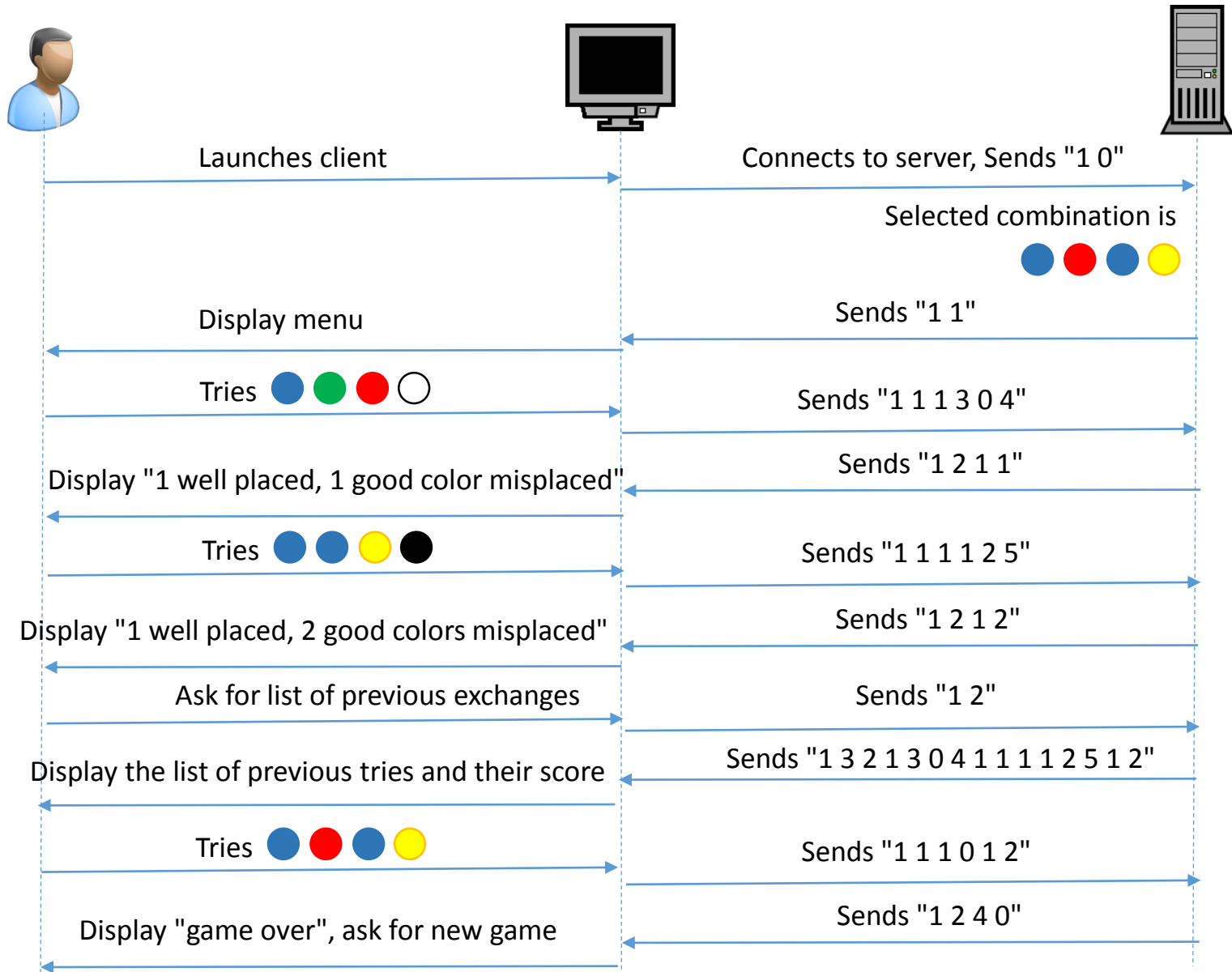


# MastermindProtocol (MP)

Mastermind protocol



# Complete Example



# (Un)intentional malevolence

- What happens if I send "1 3", or "2 0"?
  - Good behaviour : Send "1 4".
  - Bad behaviour : Trigger exception.
  - Never expect, always check!
- What happens if I send "1 1"?
  - Server waits for the rest of the request, that never comes.
  - If single-threaded, cannot handle new connections.
  - One thread for new connections, then one thread per connection.
  - Use Socket time-outs, close connection if too long.

# Extra guidelines

- Port number : 2xxx, where xxx = last 3 digits of ULiege ID
- Class named "MastermindClient" and "MastermindServer".
- The server console should display the secret combination
- No "package" instruction, no shutdown hooks, limited libraries for import, no file manipulation.
- Fully functional on student machines (ms8\*\*.montefiore.ulg.ac.be).
- Short report.
- Send submission to Montefiore Submission Platform ([submit.run.montefiore.ulg.ac.be](http://submit.run.montefiore.ulg.ac.be)) before March, 25th 2018!