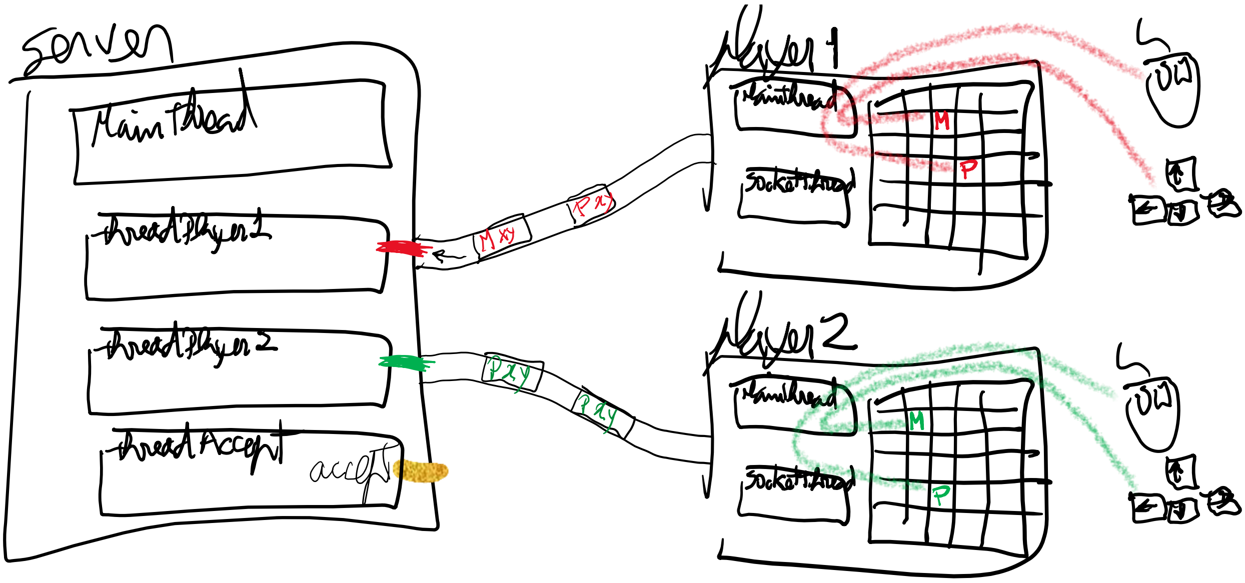
System Programming
9th Laboratory (21th .. 24th April 2020)

In this laboratory students will continue working in the foundations and libraries necessary for implementing the project.
Student should use the code developed in the last laboratory

Exercise I

Implement a simple version of the multiuser packman game.
The system will be composed of a server and multiple clients.
Each player/client will have two characters (a pacman and a monster) that are controlled by the mouse and the keyboard.
Each client will send the positions of the pacman and monster to the server that will forward this information to every other client.
The architecture of the system is as follows:

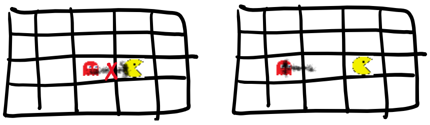


Server
When the server receives from **argv** the size of the board and creates a graphical windows where it will present all the pacmen and monsters on the game.
The server will have a socket to where the clients will connect.
The server should have the following threads:

- The **mainThread** should be responsible for updating the graphical window.
- The **threadAccept** will wait for connection from the clients and every time a new client connects to the server a new thread specific to such client should be created.
- Client threads (**threadPlayer1**, **threadPlayer2**, …) receive the updates (pacman and monster) specific to such player/client.

The server should guarantee the following

- A place on the board can only be occupied by one character (either a pacman or a monster).
- If a character tries to enter a place already occupied it will be bounced back.
- If a character goes against a wall it will also be bounced back
- After updating the position of the characters, the server should send the new position of the character to all the clients.



Client

Each client will control two characters (a pacman and a monster).

The pacman is controlled by the mouse, while the monster is controlled by the keyboard arrows.

Both characters will have the same color. The color is retrieved from the argv.

The client will connect to the server. The address of the server is retrieved from the argv.

The client receives from the server the size of the board and creates a new graphical window.

The client should have two threads:

- The mainThread will receive the events from the mouse/keyboard and update the graphical window.
- The socketThread will receive from the server the location of all the pacmen and monsters.