Students identification

# Introduction

Small introduction of the project

# Implemented functionalities

## Part A

Table 1: Implemented functionalities (PART A)

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Not implemented** | **With faults** | **Totally Correct** |
| **game-server.c** | | | |
| Suitable sockets |  |  | x |
| Assignment of Astronaut letters |  |  | x |
| Storage of clients Astronauts and Aliens |  |  | x |
| Management of astronauts |  |  | x |
| **astronaut-client.c** | | | |
| Connect |  |  | x |
| Movement |  |  | x |
| Zap |  |  | x |
| Disconnect |  |  | x |
| Display score |  |  | x |
| **outer-space-display.c + display on game-server.c** | | | |
| Correct update of screen |  |  | x |
| Zapps |  |  | x |
| Destruction of aliens |  |  | x |
| **Rules** | | | |
| Movement of Aliens |  |  | x |
| Astronaut zapping |  |  | x |
| Astronaut zapping (fire rate) |  |  | x |
| Astronaut zapping (0.5 second ray) |  |  | x |
| Aliens are destruction (points) |  |  | x |
| Astronauts stunning |  |  | x |
| **Misc** | | | |
| Messages validation |  |  | x |
| Optimization of communication |  |  |  |
| Code organization |  |  | x |
| Functions return values validation |  |  |  |

## Part B

Table 2: Implemented functionalities (PART B)

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Not implemented** | **With faults** | **Totally**   * + 1. **Correct** |
| **game-server.c** | | | |
| 1. Suitable sockets |  |  | x |
| 1. Disconnect of clients |  |  | x |
| 1. Communication thread | x |  |  |
| 1. Aliens thread |  |  | x |
| 1. Synchronization |  |  | x |
| **astronaut-display-client.c** | | | |
| Connect |  |  |  |
| Movement |  |  |  |
| Zap |  |  |  |
| Disconnect |  |  |  |
| Update of display |  |  |  |
| Display score |  |  |  |
| Threads |  |  |  |
| **space-high-scores** | | | |
| Client on other language |  |  | x |
| Protocolbuffer messages |  |  | x |
| **Rules** | | | |
| Movement of Aliens |  |  | x |
| Astronaut zapping |  |  | x |
| Astronaut zapping (fire rate) |  |  | x |
| Astronaut zapping (0.5 second ray) |  |  | x |
| Aliens destruction and points |  |  | x |
| Astronauts stunning |  |  | x |
| Aliens population recovery |  |  |  |
| **Misc** | | | |
| Messages validation |  |  |  |
| Optimization of communication |  |  |  |
| Code organization |  |  |  |
| Functions return values validation |  |  |  |

## Description of faulty functionalities

In this section students should briefly describe (2/3 lines) every functionality that was not completely implemented, and why.

# PART A Description of code

## Data types

In this section students should present a list and describe the various datatype (structs) and lists/array used to store the Astronauts, aliens and clients

## Functions List

In this section students should present a list with every implemented function, divided by the various components implemented:

* game-server.c
* astronaut-client.c
* outer-space-display.c.

For each function students should describe the objective of each function.

## Implementation details

Description of the implementation of the astronaut zapping fire rate.

Description of the implementation of the astronaut zapping 0.5 second ray display

Description of the implementation of the astronauts stunning (immobility)

# PART B Description of code

## Data types

In this section students should present a list and describe the various datatype (structs) and lists/array that were added or modified in server of the Part B of the project.

## Functions List

In this section students should present a list with new of modified functions from Part A to Part B:

* game-server.c
* astronaut-display-client.c
* space-high-scores.c

For each function students should describe the objective of each function.

## Implementation details

Description of the implementation of the astronaut zapping fire rate.

Description of the implementation of the astronaut zapping 0.5 second ray display.

Description of the implementation of the astronauts stunning (immobility).

Description of the implementation of the aliens destruction and points

Description of the implementation of the aliens population recovery

## Threads

In the section students should present a list of every thread implemented, divided by components:

* astronaut-display-client.c
* game-server.c

Students should describe its overall functioning and objectives of each thread.

## Shared variables

In this section students should present all shared variables that are accessed by multiple threads, divided by components.  
Students should describe the objective of each variable and what threads access it.

## Synchronization

In this section students should present all synchronization variables used: mutextes.

Students should describe what type of guarding of synchronization they implement, what variables are related and on what functions they are accessed.

# PART A Communication

## Sockets

In this section student should decribe the sockets used in Part A.

## Transferred data

In this section students should present the exchanges data structures/messages and relate them to the sockets described earlier.

Students can present the typedef used and relate them to the messages presented in the assignments.

## Error treatment

In this section students should present what type of error treatments were implemented related to communication: validation of read data, verification of read/write return errors, ...

# PART B Communication

## Sockets

In this section student should decribe any new socket in Part B.

## Transferred data

In this section students should present the modified messages from part A to Part B.

## Error treatment

In this section students should present what type of error treatments were implemented related to communication: validation of read data, verification of read/write return errors, …

# Conclusion

Final remarks about the project, what was learned, accomplishments, major difficulties and adversities.