

LCOM Final Project



T-REX game

LCOM - T4G05

Ana Luísa Ferreira Marques -
up201907565 Miguel Faria Amorim -
up201907756

Tiago Rodrigues - up201906807

December 2020

CONTENTS

Introduction	2
User Instructions	3
Menu	3
Leader board	4
Play	5
Gameplay	6
Username	7
Project status	8
Timer	8
KBD	8
Mouse	9
Video card	9
RTC	9
Code Organization / structure	10
Proj.c	10
While_game.c	10
Sprite.c	10
User.c	10
Graphic.c	11
Keyboard.c	11
Mouse.c	11
RTC.c	11
Timer.c	11
Utils.c	12
File.c	12
Implementation details	14
Structs	14
Frame rate	14
Leader Board	14
Collision	14
Conclusion	15

INTRODUCTION

Within the course of Laboratório de Computadores (LCOM), we decided to do a game as final project, since it is easier to show the knowledge and the skills acquired with the classes.

With that in mind, we prepared the famous game **T-REX GAME**, the game that people play when don't have internet on browser Chrome. Here, the main point is to survive if possible, while there are obstacles that make the user lose if he collides with any of them.

If you lose, the game shows **GAME OVER** and if the user made a new record (better than, at least one of three old best scores), the user inputs his nickname and goes to the leader board.

The game developed in this project is the well-known "T-rex game", now with new features. Still, as easy to handle as the original, as described below.

USER INSTRUCTIONS

MENU

The game starts with the initial menu(fig.1), with allows the user to start the game, pressing the SPACEBAR button of the keyboard, or to see the leader board pressing ENTER of the keyboard. HOWEVER, the user must introduce his nickname before starting the game.



There is no Internet connection
PRESS SPACEBAR TO START

LEADER BOARD

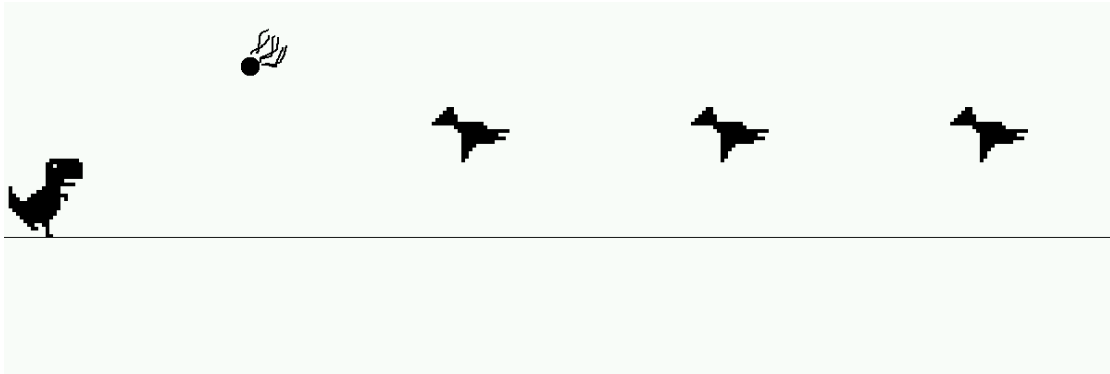
When the user presses ENTER, the option leader board is selected, and the best three scores made yet are shown. To each score, there is a respective nickname and the date (dd mm aaaa) when the user made that score.

To go back to the menu board, the user must press SPACEBAR. The next figure (fig.2) shows how the leader board looks like.

LEADERBOARD			
SCORE	DATE		
2548	04	01	2021
1847	04	01	2021
1435	04	01	2021

PLAY

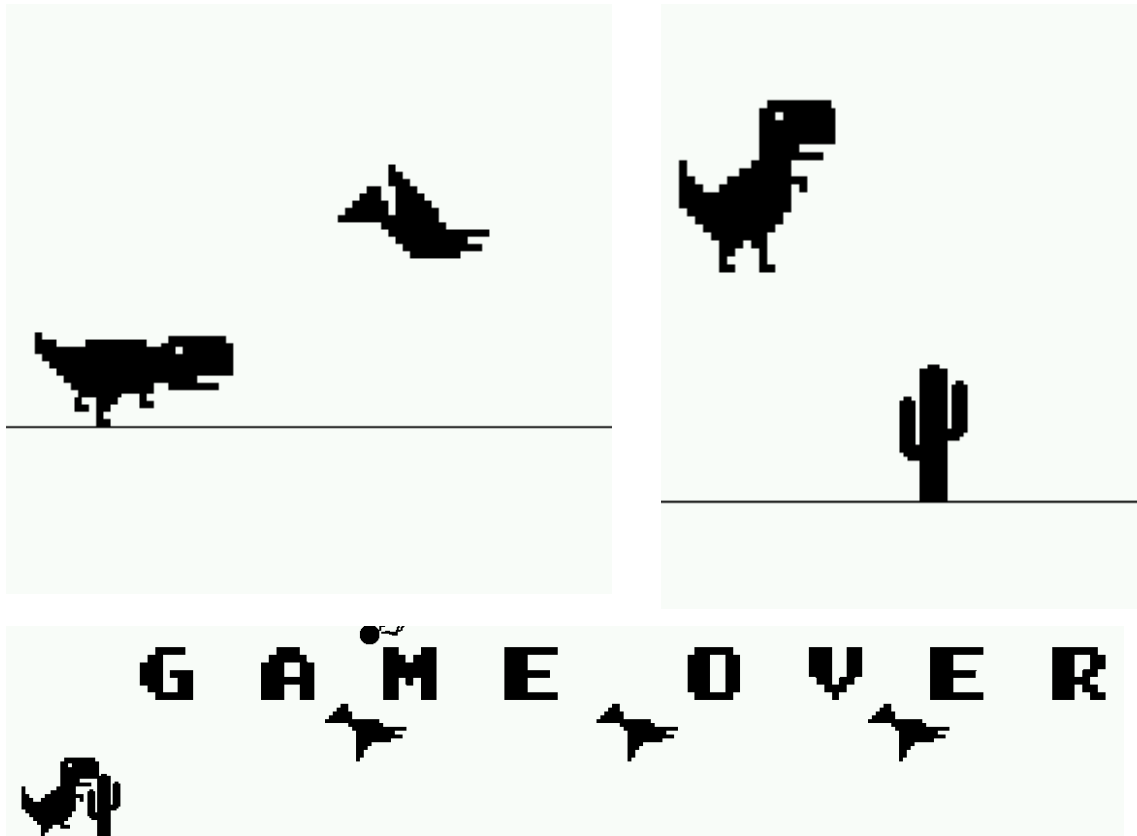
When the user presses SPACEBAR, the game starts, showing the game avatar, the T-REX.



GAMEPLAY

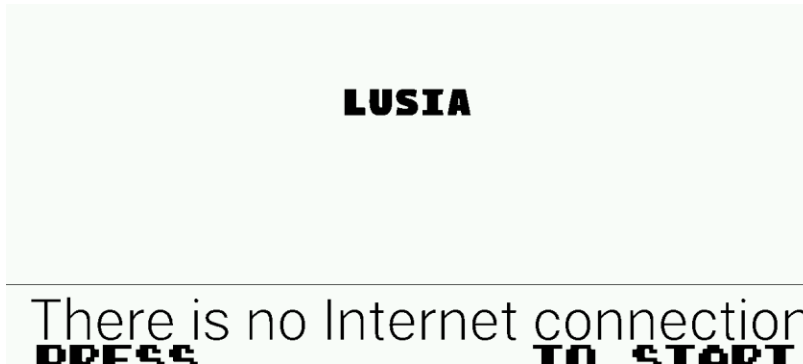
After pressing SPACEBAR in Menu section, the game starts and the user must dodge the obstacles, that are going to spawn. The user must avoid the cactus by jumping pressing UP ARROW, avoid the birds by stooping pressing DOWN ARROW and avoid the meteorites using mouse to destroy them.

While the user doesn't die, the obstacles continue spawning and moving faster than before to make the game more difficult to play. When the user dies, the game shows GAME OVER and if it's a new record the game appear NEW RECORD and the score.



USERNAME

In the beginning of the game the user must introduce his nickname and the game will show his nick while in the game.



PROJECT STATUS

All the features mentioned in the previous section were implemented. However, the serial port belonging to the initial objectives has not been implemented.

Device	What for	Interrupt mode
Timer	Controlling frame rates and scores	Interrupt
KBD	Menu navigation, control the T-Rex and input the username	Interrupt
Mouse	Destroy meteorites	Interrupt
Video Card	Application menus and screens display Page flipping	N/A
RTC	Obtain current data to score record	N/A

TIMER

The timer is used to control the flow of the game (each sprite print):

- to make the dinosaur change its legs, conveying the idea of movement (switch_leg)
- to make the bird change its wings, for the same reason (obstacles_spawn),
- so that the obstacles appear on the screen in time sufficiently spaced for the dinosaur to be able to overcome them, namely:
 - the cactus (obstacles_spawn),
 - the birds (obstacles_spawn)
 - the meteorites (obstacles_spawn)
- to increase the score (score)

The timer works with interrupts, so we used the functions developed on lab2.

KBD

The KBD is used in the menu, to start the game with spacebar (menu_handler)

It is also used to control the development of the dinosaur in the game, with the down and up arrow, allowing the dinosaur to dodge the obstacles that appear (jumpup, jumpdown, down)

The KBD is also used for the user to type his nickname (nickname)

The KBD works with interrupts, so we used the functions developed on lab3

MOUSE

The position and the buttons of the mouse is used to make meteorites that fall from the sky disappear, by clicking on them. Its offset is used to check if the cursor coordinates are coincident with the meteorite's coordinates (`new_mouse_pos`).

We also use function `new_mouse_pos()` to update the coordinates of the cursor image.

The mouse works with with interrupts, so we used the functions developed on lab4.

VIDEO CARD

The graphic mode used it's the 0x11A mode, a mode with 1280x1024, where the colours are direct colours and there are 16(5:6:5) bits per pixel.

To improve the animations, we used double buffering and page flipping (`PageFlipping`).

All kinds of animations are made by the video card.

The related functions

Video card is used to make sprites appear in the screen:

- to show the dinosaur walking and jumping and crouching
- to show the bird flying, flapping wings
- to make the meteorites fall, from the right to the left
- to make the cactus slide across the floor
- to show the score at the top right corner
- to show the score board
- to check if the collision between the dinosaur and the obstacle, checking if the borders of the dinosaur are in contact with a painted pixel.

RTC

The RTC is used to get and show the current date to score record using functions like `rtc_reg_data()` and the rest of the functions that read some element of date.

CODE ORGANIZATION / STRUCTURE

PROJ.C

On this module, there is the main function, where we call `while_game()` function (game function) and all interruption subscribes.

Responsible to do this file: Luísa Marques.

Percentage of this file to the all project: 2%

WHILE_GAME.C

On this module, all images and sprites are loaded.

Here is where we receive all interrupts from all features(timer/kbd/mouse/rtc).

Responsible to do this file: Luísa Marques.

Percentage of this file to the all project: 20%

SPRITE.C

On this module, all functions related with moving sprites are developed, like moving the T-Rex, spawn the obstacles. It's also here the functions related to write on the screen, important when the user inputs his nickname or to show the current score.

Responsible to do this file: Miguel Amorim e Tiago Rodrigues.

Percentage of this file to the all project: 40%

USER.C

On this module, we create the struct User to save some information, like the score he did, his name and the date of the score.

Responsible to do this file: Luísa Marques.

Percentage of this file to the all project: 2%

GRAPHIC.C

On this module, we can see all functions related to the video card feature, all prints, VBE functions and functions developed on lab5.

Responsible to do this file: Miguel Amorim.

Percentage of this file to the all project: 10%

KEYBOARD.C

On this module, we can see all functions related to the keyboard feature developed on lab3.

Responsible to do this file: All the members of the group.

Percentage of this file to the all project: 2%

MOUSE.C

On this module, we can see all functions related to the mouse feature developed on lab4.

Responsible to do this file: Tiago Rodrigues.

Percentage of this file to the all project: 9%

RTC.C

On this module, we can see all functions related to the internal clock feature to read data information, to save on files.

Responsible to do this file: Luísa Marques.

Percentage of this file to the all project: 8%

TIMER.C

On this module, we can see all functions related to the timer feature developed on lab2.

Responsible to do this file: Miguel Amorim.

Percentage of this file to the all project: 3%

UTILS.C

On this module, we can see some functions related to help feature on other functions.

Responsible to do this file: Tiago Rodrigues.

Percentage of this file to the all project: 2%

FILE.C

On this module, we can see all functions related txt files.

Responsible to do this file: Tiago Rodrigues e Luísa Marques.

Percentage of this file to the all project: 4%

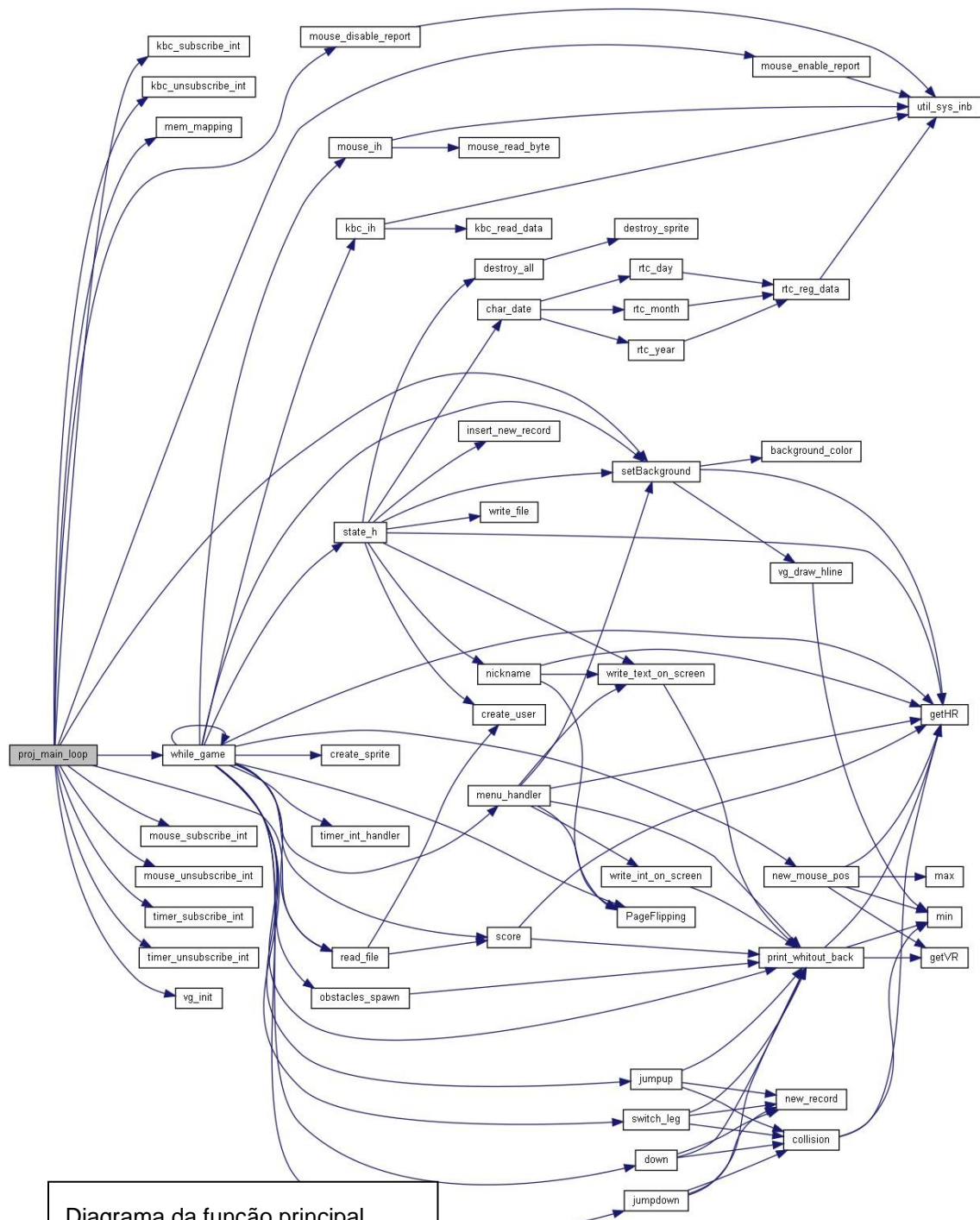


Diagrama da função principal
do projeto

IMPLEMENTATION DETAILS

STRUCTS

It was useful to use structs, like `Sprite`, which saves the information about one image and `User`, which saves information like score, nickname, and date.

FRAME RATE

The timer is used to control the frame rate and the generation of frames.

While on the game:

- Every 100 interruptions, a new cactus spawn.

- Every 600 interruptions, a new meteor spawn.

- Every interruption, all sprints are updated.

- Every interruption, the dinosaur switches leg.

- Every interruption, the user accumulates one point.

LEADER BOARD

As said before, every interruption, the user accumulates one point, so we decided to make a leader board

We are saving the three best scores, the order is decreasing and if a new record is made, the program changes with the new record:

COLLISION

As said before, every interruption, the user accumulates one point, so we decided to make a leader board

We are saving the three best scores, the order! on the `files_scoore.t` is decreasing and if a new record is made, the program changes with the new record.

CONCLUSION

All of us, will remember this course as hard-working but interesting.

On one hand, it was difficult to finish some labs and the material was difficult to find, since it is so dispersed through the handouts.

On the other hand, it is our first project out of the console. The first project in which we had to be more organized, and the fact that all material was on doxygen files was helpful.

So, despite all the hard work this project took, it was exciting, we acquired a lot of experience and we realize how important it was to learn about the I/O devices.