$\begin{array}{c} {\bf CAB202 \text{ - Microprocessors and Digital}} \\ {\bf Systems} \end{array}$

Assignment 1

Pedro Alves (n9424342)

April 19, 2018

Contents

Executive Summary	3
Program Overview	4
Splash Screen	5
Border	6
Dashboard	8
Race Car	9
Horizontal Movement	10
Acceleration and Speed	11
Scenery and Obstacles	12
Fuel Depot	13
Fuel	14
Distance Travelled	15
Collision	16
Game Over Dialogue	17
Part B - Highscore Screen	18
References	10

Executive Summary

Program Overview

Things to talk about change_state

Splash Screen

The splash screen is the first screen the player sees when they start the game. It provides basic information about the game and will change to the main game screen when the player presses any key.

Functions

```
// main.c
void update_start_screen();
```

Called every tick of the main game loop. Will change the game's state to *GAME_SCREEN* if there is any key in the input buffer. Since the *change_state()* function already purges the input buffer, we do not have to worry about the game skipping straight to the *GAME_SCREEN*.

```
// main.c
void draw_start_screen();
```

Calculates the x and y coordinates of each string to be shown based on the dimensions of the screen. Will then call $draw_string()$ and $draw_center_text()$ multiple times to add the strings to the desired location.

```
// main.c
void draw_center_text(char * text, int y);
```

Calculates what x coordinate is required in order to have the text appear at the middle of the screen. Then calls $draw_string()$ to print the text.

Testing

Testing that the splash screen shows up when the game is started.

```
Race to Zombie Mountain

INSTRUCTIONS

Reach the finish line

Collisions reduce car condition

Game over if car condition is 0,

collides with fuel station or

runs out of fuel

Drive with low speed next to fuel station to refuel

Press any key to play...

Pedro Alves - n9424342
```

Figure 1: The splash screen when the player starts the game

Border

The border is simply a rectangle that is drawn on the edge of the terminal. It supports every terminal size. The $draw_borders()$ function is the last one called before $show_screen()$ in the draw step of the game loop. This ensures that no other graphics ever block the border.

Globals

```
// zombiemountain.h
#define BORDER.CHAR 46
```

The character that will be used to represent the border. The number 46 represents the ASCII character "." (full stop).

Functions

```
// main.c
void draw_borders();
```

Draws 4 lines that form a rectangle on the edge of the screen. The length of these lines are calculated by using the screen width and height in order to make the borders work on every screen size.

Testing

The game is started in different sized terminals and the borders are verified to have been drawn correctly.

Screen: 80x24

```
Screen Width: 80
Screen Height: 24
Race to Zombie Mountain

INSTRUCTIONS
CONTROLS
Reach the finish line
Collisions reduce car condition
Game over if car condition is 0,
collides with fuel station or
runs out of fuel
Drive with low speed next to fuel station to refuel

Press any key to play...

Pedro Alves - n9424342
```

Figure 2: The border with screen dimensions of 80x24

Screen: 126x31

```
Screen Width: 126
Screen Height: 31
Race to Zombie Mountain

INSTRUCTIONS
Reach the finish line
Collisions reduce car condition
Game over if car condition is 0,
collides with fuel station or
runs out of fuel
Drive with low speed next to fuel station to refuel

Press any key to play...

Pedro Alves - n9424342
```

Figure 3: The border with screen dimensions of 126x31

Screen: 190x50

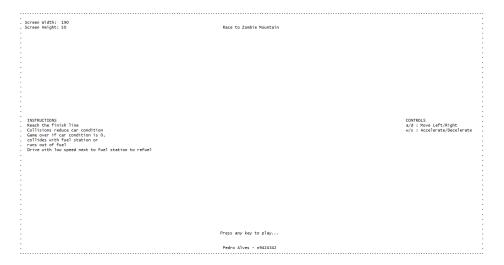


Figure 4: The border with screen dimensions of 190x50

Dashboard

Race Car

Horizontal Movement

Acceleration and Speed

Scenery and Obstacles

Fuel Depot

Fuel

Distance Travelled

Collision

Game Over Dialogue

Part B - Highscore Screen

References