A picture containing font, graphics, screenshot, black

Description automatically generated

## TypoRacer

## LC 2022/2023

T10G03

Guilherme Coutinho

Rui Carvalho

Xavier Outeiro

Maio 2023Instruções

### **Menu Inicial**

* Quando é iniciado o programa é apresentado o menu principal, onde é possível escolher, utilizando o cursor, uma de duas opções:
  + **Play –** O programa muda para o[***Game State***](#_Estado_de_Jogo_1)***.***
  + **Exit –** Sai do programa.



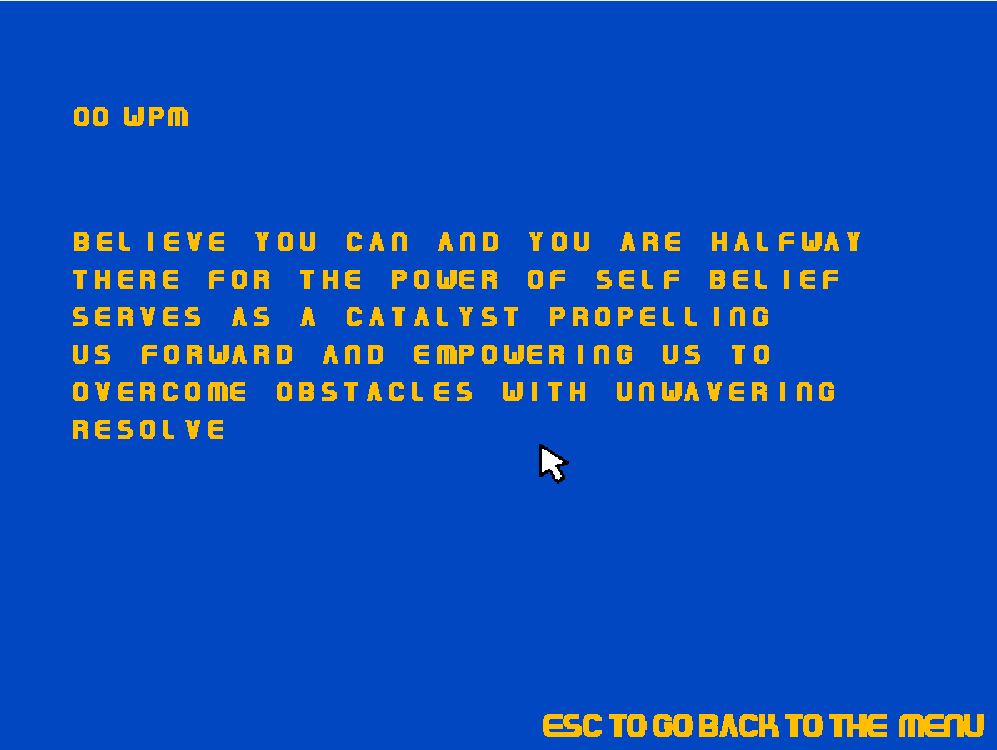
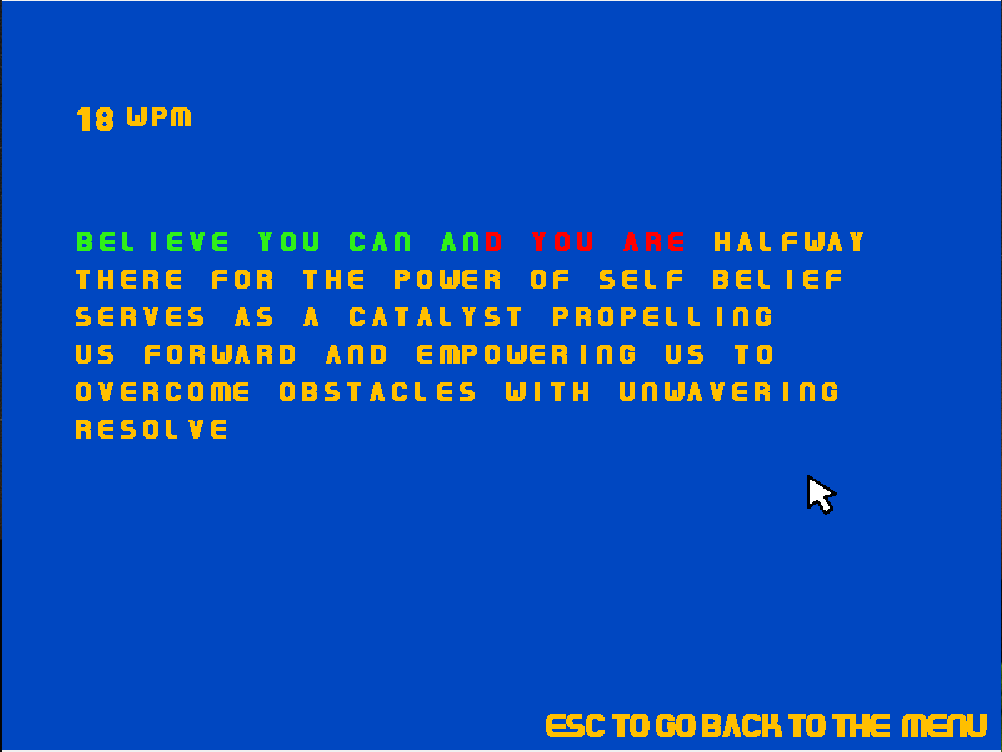
*Figura 1: Menu Inicial - Play*

*Figura 2: Menu Inicial - Exit*

### 

### **Game State**

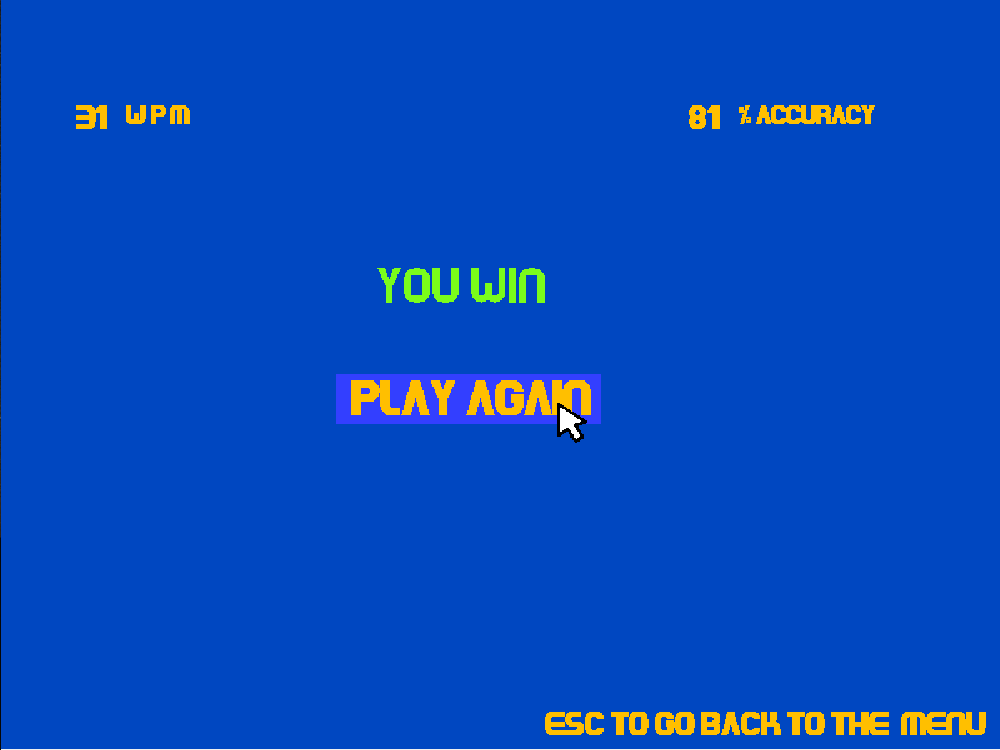
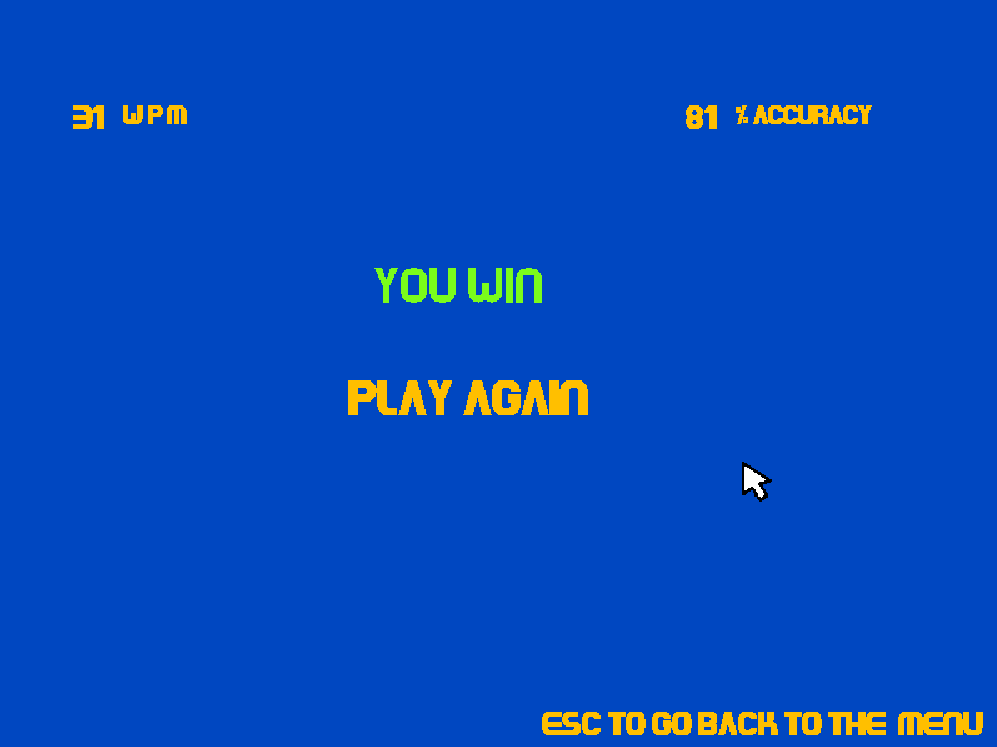
* É neste estado que decorre a funcionalidade principal deste jogo, onde é desenhada no ecrã uma frase aleatória, e onde o objetivo do jogador é escrever essa mesma frase o mais rápido e acertadamente possível. Isto é, a partir do momento que o jogador começa a escrever, e considerando o tamanho de uma palavra como 5 caracteres, é também mostrado no ecrã a velocidade a que o jogador está a escrever em *wpm (Words Per Minute),* com base no tempo passado desde que o jogador começou a escrever e no número de caracteres corretamente escritos.
* São também mostrados a verde e vermelho os caracteres já escritos corretamente e incorretamente, respetivamente, pelo jogador.
* A partir daqui é possivel o jogador ir para outros dois estados:
  + Se pressionar a tecla Esc irá voltar para o [***Menu Inicial***](#_Menu_Inicial)*.*
  + Se escrever toda a frase corretamente irá para o [***Win State***](#_Win_State).



*Figura 4: Game State - Jogo a decorrer*

*Figura 3: Game State - Inicialmente*

### **Win State**

* Neste estado é apresentada, não só uma mensagem para o jogador (“You Win”), mas também a velocidade em *wpm* a que o jogador escrever a frase e a percentagem de acerto do jogador, com base nos erros e no tamanho da frase apresentada.
* A partir deste estado o jogador pode:
  + Pressionar a tecla Esc, voltando assim para o [***Menu Inicial***](#_Menu_Inicial).
  + ****Selecionar o botão *Play Again,* iniciando, de novo, o [***Game State***](#_Game_State).

*Figura 5: Win State*

*Figura 6: Win State - Play Again*

**Estado do Projeto**

### Dispositivos

|  |  |  |
| --- | --- | --- |
| Dispositivo | Funcionalidades | Int. |
| Timer | Frame Rate | Sim |
| Rato | Seleção de opções nos menus | Sim |
| Teclado | Escrita da frase mostrada no ecrã e seleção de opções nos menus | Sim |
| Placa Gráfica | Mostrar toda a UI do jogo | Não |
| RTC | Contar o tempo a partir do momento em que o utilizador começa a escrever frase, esse valor é usado no cálculo da velocidade de escrita | Sim |

#### Timer

A cada interrupção do timer, neste caso o timer 1, é chamada a função ***timer\_update\_state***, e o ecrã é atualizado (60 vezes por segundo), é atualizado também o desenho do cursor de acordo com a posição do rato na função ***design\_draw\_mouse*** e é efetuada a troca de buffers utilizando a função ***design\_swap\_buffers***. A frequência do timer é definida na função ***timer\_set\_frequency.***

#### Rato

A cada interrupção do rato, é chamada a função ***mouse\_update\_state***, atualizando a sua posição em coordenadas cartesianas e verificando se tanto o botão direito como o esquerdo são carregados. Se a posição do rato e o intervalo das coordenadas dos botões dos menus coincidirem, a cor de fundo do botão do menu muda, e se o botão do lado esquerdo do rato for pressionado é efetuada uma mudança de estado do jogo. Isto é feito pelas funções ***menu\_buttons\_update***e ***end\_menu\_buttons\_update.***

#### Teclado

A cada interrupção do teclado é chamada a função ***keyboard\_update\_state***, que atualiza a variável *scancode*, que corresponde à tecla pressionada, e depois, se estiver no Game State, é convertida num caracter utilizando ***get\_letter\_from\_keyboard*** já no interior da função ***game\_controls***, onde esse caracter é comparado ao caracter da posição onde o jogador se encontra e de acordo com essa comparação realizar as ações do jogo.

O teclado é também usado para a navegação entre menus, nomeadamente voltar ao menu principal utilizando a tecla *Esc.*

#### Placa Gráfica

A Placa gráfica é utilizada para desenhar no ecrã todos os elementos do jogo, como o menu inicial, o menu do jogo onde é desenhada a frase e o menu final. O modo de vídeo utilizado é 0x115 que proporciona uma resolução de 800x600, em modo de cor direto o que dá a possibilidade de utilizar 224 cores.

É utilizado o Double buffering via copy, a cada interrupção do timer, utilizando a função ***design\_swap\_buffers.***

Todos as letras e números do jogo foram implementados utilizando *XPM’s*.

A frase, um dos elementos principais do jogo, é desenhada no ecrã utilizando a função ***phrase\_writer***, que inicialmente desenha a frase toda da mesma cor, mas de acordo com o desenrolar do jogo, muda as cores das letras.

#### RTC

O RTC é utilizado a partir do momento em que o jogador começa a escrever a frase, no Game State, com a função ***rtc\_counter\_start***, e sendo que é sempre mostrado no ecrã a velocidade a que se está a escrever é usada a função ***get\_elapsed\_time***, calculando-se assim o tempo passado desde o início e a velocidade de escrita.

## Estrutura do código

### Controller

### Keyboard\_Mouse

### Contém as funções que lidam com as interrupções tanto do teclado como do rato.

### RTC

### Contém as funções que lidam com as interrupções RTC, a função que começa o contador e a função que retorna o tempo passado.

### Timer

### Contém as funções que lidam com as interrupções do timer e a função que define a sua frequência.

### Video

### Contém as funções que lidam com o buffering do ecrã e também funções que desenham pixéis, linhas e retângulos.

### XPM

### Contém os ficheiros .xpm de todos os elementos que são desenhados no ecrã.

### Sprite

### Contém as funções que permitem criar o texto que é desenhado no ecrã.

### Game

### Game

### Contém a struct que define o objeto game, a função que o cria, a que lida com os inputs do jogador e a função que gera a frase aleatória.

### Proj

### Contém o loop principal do projeto onde se dá setup, se lida com as interrupções dos dispositivos, e se dá cleanup de todo o projeto.

### States\_Design

### States

### Contém as funções que alteram o estado do jogo de acordo com as interrupções geradas.

### Design

### Contém as funções desenham todos os elementos do jogo de acordo com o estado em que ele se encontra.

## A screenshot of a computer Description automatically generated with low confidenceGráfico de chamada de funções

## Detalhes de implementação

### Máquinas de estados

Foi usada uma máquina de estados no projeto, que desenha uma *frame* diferente de acordo com o *menuState,* e foi implementada utilizando a função ***design\_draw\_new\_frame.***

## Conclusões

Apesar de algumas dificuldades iniciais em relação à matéria, com alguma ajuda dos professores e algum esforço extra, foi possível avançar na realização do projeto.

Mesmo assim ficaram ainda algumas funcionalidades que gostaríamos de ter implementado e não foi possível devido a falta de tempo e de contribuição de alguns elementos.

Se o projeto fosse continuado no futuro as funcionalidades que gostaríamos de implementar seriam um modo de jogo em que houvesse tempo limite para escrever a frase e um modo de multiplayer utilizando Serial Port.