

Faculdade de Engenharia da Universidade do Porto

Projeto final

Laboratório de Computadores



Trabalho realizado por:

Grupo 2 – Turma 10

- Diogo Santos(up202108747@fe.up.pt)
- José Santos(up202108673@fe.up.pt)
- Maria Inês Miranda(up202108775@fe.up.pt)
- Miguel Dionísio (up202108788@fe.up.pt)

Índice

Introdução	3
Instruções de Utilização	4
Menu Inicial	4
Como passar os níveis	5
Menu Final	6
Estado do Projeto	7
Tabela de dispositivos	7
Dispositivos	8
Video Graphics	8
Teclado	8
Mouse	9
Timer	9
Organização do Código/Estrutura	10
Video Graphics Module	10
Keyboard Module	10
Mouse Module	10
Timer Module	10
Utils Module	11
Game Levels Module	11
Game Logic Module	11
Game View Module	11
Sprites Module	11
Main	11
Function call Graph	12
Detalhes da Implementação	13
Conclusões	14

Introdução

O jogo que escolhemos para este projeto foi o “Thin ice”, que fazia parte da grande variedade de jogos que havia Club Penguin, e que marcou a nossa infância.

Na verdade, o nosso jogo consiste numa adaptação do jogo original. No jogo o que acontece é que em cada cubo de gelo pelo qual a nossa personagem, conhecida no jogo como ‘puffle’, se movimenta o gelo por onde ele passa derrete e vira água, se ele tocar na água morre e o objetivo é chegar ao cubo vermelho para passar para o próximo nível.

No jogo original para passar cada nível não é exigido ao jogador que tenha derretido o gelo todo do nível anterior, nós adaptamos isso para aumentar o nível de dificuldade para o jogador e tornar o jogo mais interessante, no nosso jogo só é possível passar para o nível seguinte quando o gelo todo estiver derretido, caso chega ao cubo vermelho e isto não se verifique volta à posição inicial. Outra adaptação que fizemos é que não existem cubos de gelo que só derretam à segunda passagem do jogador por ele.

O jogo acaba quando o jogador passar todos os 10 níveis, a dificuldade dos níveis vai sempre aumentando à medida que vão passando.

Instruções de Utilização

Menu Inicial

Inicialmente, ao abrir o jogo a primeira coisa a aparecer é o menu inicial que contém dois botões a partir do qual clicando dentro da sua área com o rato consegue aceder ao primeiro nível ou sair, isto é clicando no botão 'Start' ou 'Quit' respetivamente. Isto também é possível se usar o teclado ao pressionar a tecla 'G'.



Como passar os níveis

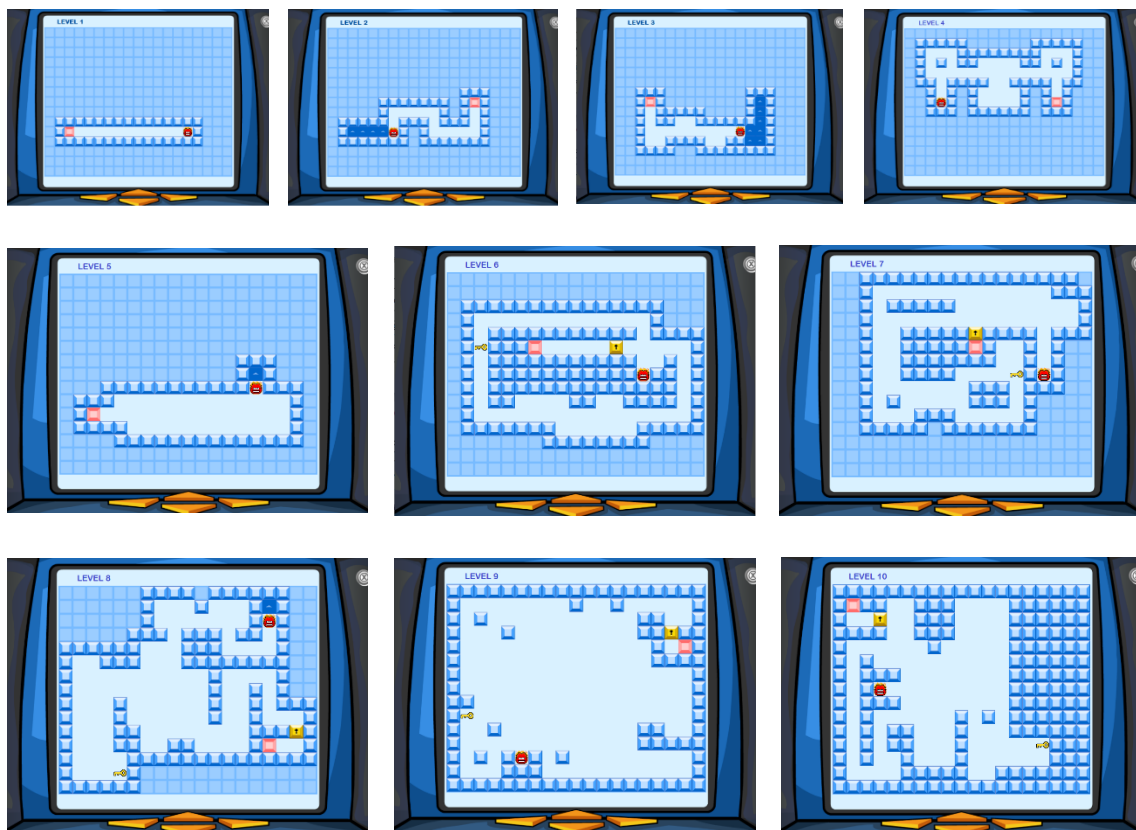
O objetivo do jogo é conseguir que o puffle chegue ao cubo vermelho derretendo todos os cubos de gelo claros pelo caminho. O nível não fica completo até que derreta realmente todos os cubos. Sempre que tocar em água volta à posição inicial e reinicia o nível, o mesmo acontece se chegar ao cubo vermelho sem derreter todos os cubos pelo caminho. Não é possível passar pelos cubos de gelo mais escuros.

Nos níveis mais avançados irá aparecer-lhe um cadeado e umas chaves. Quando isto se verificar o jogador tem que passar pela chave para que o cadeado desapareça e assim seja possível acabar o nível.

Para movimentar o puffle o jogador tem que pressionar no teclado as seguintes teclas:

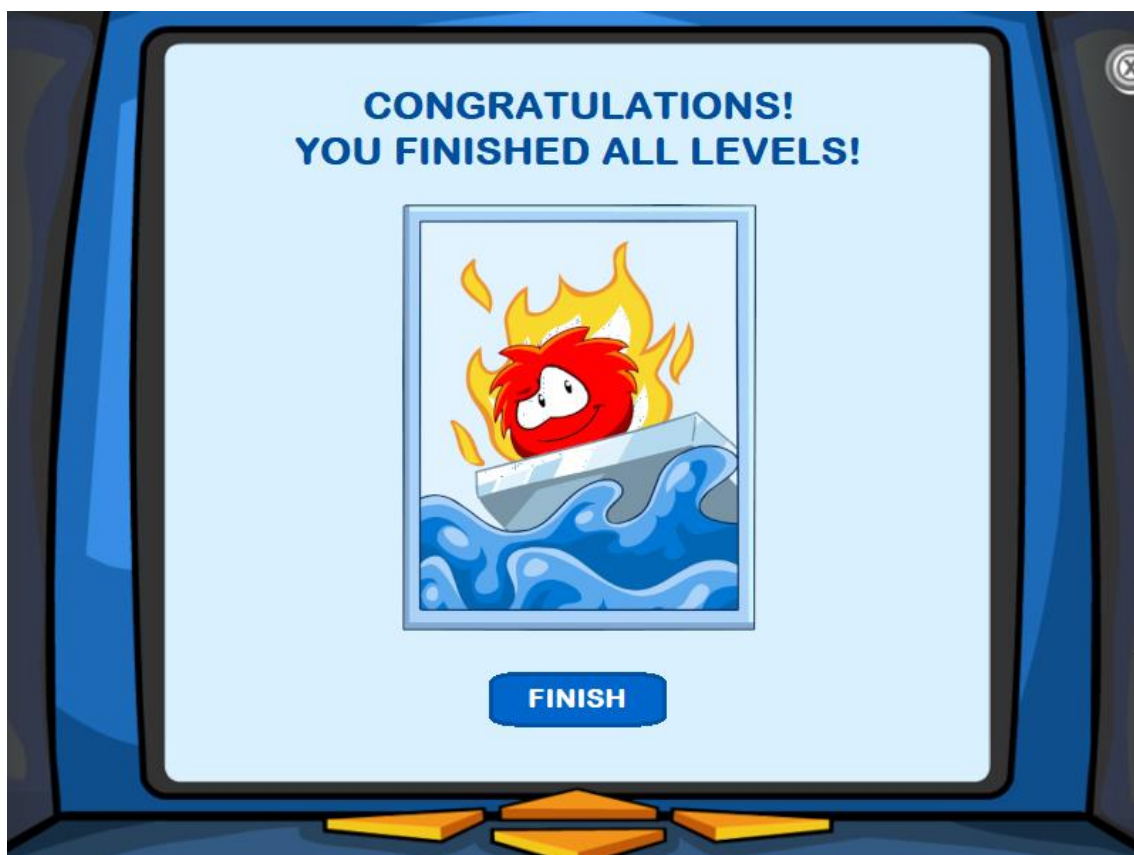
- 'W' para andar para cima;
- 'S' para andar para baixo;
- 'A' para andar para a esquerda;
- 'D' para andar para a direita;

Só é possível passar para o nível seguinte depois de completar o nível anterior. Em qualquer momento do jogo se quiser desistir pode pressionar a tecla 'E' e aparecerá o menu final.



Menu Final

O menu final aparece ao terminar todos os níveis ou ao pressionar a tecla 'E', nele tem um botão, ao utilizar o rato para clicar na sua área ou ao pressionar a tecla 'H' irá aparecer-lhe novamente o menu inicial.



Estado do Projeto

Das funcionalidades antes mencionadas conseguimos implementar todas as que queríamos, de maneira a deixar o jogo o mais parecido com o original, com exceção do cubo gelo que só derrete à segunda passagem do puffle por ele.

Tabela de Funcionalidades

Dispositivo	Funcionalidade	Interrupções
Timer	Define o frame rate da placa gráfica para 60 frames por segundo	Sim
Teclado	Mover o puffle nos vários níveis, trocar para os menus ou para o primeiro nível, sair do jogo	Sim
Mouse	Selecionar opções tanto no menu inicial como no final	Sim
Video Graphics	Mostrar todo o jogo: menu inicial, 10 níveis e menu final	Sim

Dispositivos

Video Graphics

Neste jogo o video graphics é utilizado para dar display dos menus, tanto inicial com final, e dos níveis.

Todo o jogo acontece no modo de vídeo sendo por isso o display dos caracteres realizado através de XPMs. Todos os XPMs usados neste projeto geramos através do GIMP. Usamos uma resolução de 800×600 pixéis, modo VBE 0x115, com cores de 3 bytes em modo direto, ou seja, permitindo 2^{24} cores diferentes.

Neste projeto usamos sprites para criar vários elementos do nosso programa como os botões e os backgrounds dos menus, todos os diferentes cubos, as chaves, o cadeado, as tags e a moldura dos níveis. Para esse efeito usamos a função `Sprite_t *sprite_create(xpm_map_t sprite)` e os vários XPMs que temos para cada elemento antes referido.

Implementamos também single buffering, tudo é desenhado no `secondary_fb` que é copiado para o `primary_fb` a cada timer interrupt.

A implementação da configuração e funcionalidades do video graphics encontram-se no ficheiro `graphics.c`.

Teclado

O teclado controla praticamente todo o jogo. Nos menus, tal como o rato, também pode ser usado para concretizar qualquer opção dos menus e nos níveis é responsável por todos os movimentos do puffle. Para além disso independente de estar nos menus ou nos níveis certas teclas têm sempre a mesma funcionalidade:

- Tecla 'E' - ir para o menu final;
- Tecla 'G' - ir para o primeiro nível;
- Tecla 'H' - ir para o menu inicial;
- Tecla 'Q' - sair do jogo;

Nos níveis como já referido para mover o puffle são usadas as seguintes teclas:

- 'W' para andar para cima;
- 'S' para andar para baixo;
- 'A' para andar para a esquerda;
- 'D' para andar para a direita;

Para todas as funções utilizadas para o teclado, por cada tecla pressionada, é feito o processamento do input através do make code gerado.

A implementação das configurações, funcionalidades e interrupções do teclado estão no ficheiro keyboard.c.

Mouse

O mouse é utilizado para seleccionar as opções de ambos os menus. Para desempenhar esta funcionalidade as suas várias posições são convertidas em coordenadas cartesianas para que assim seja possível distinguir os botões que o jogador selecciona.

A implementação das configurações, funcionalidades e interrupções do mouse estão no ficheiro mouse.c.

Timer

O timer é utilizado para controlar o frame rate da placa gráfica. Estabelece que o frame rate é 60 frames por segundo, ou seja, atualiza o ecrã 60 vezes por segundo.

A implementação das configurações, funcionalidades e interrupções do timer estão no ficheiro timer.c.

Organização do Código/ Estrutura

Video Graphics Module (relative weight- 15%)

Neste módulo estão presentes as funções que desenvolvemos no Lab5. As funções utilizadas permitem a configuração do video graphics e desenhar a estrutura da base dos nossos níveis que consiste em desenhar várias e linhas horizontais e verticais.

Keyboard Module (relative weight- 15%)

Neste módulo estão presentes as funções que desenvolvemos no Lab3. Com estas funções configuramos o dispositivo e conseguimos aplicar todas as funcionalidades que queríamos ligadas ao dispositivo, ao conseguirmos invocar as interrupções geradas pelo dispositivo.

Mouse Module (relative weight- 5%)

Neste módulo estão presentes as funções que desenvolvemos no Lab4. Com estas funções configuramos o dispositivo e conseguimos aplicar todas as funcionalidades que queríamos ligadas ao dispositivo, ao conseguirmos invocar as interrupções geradas pelo dispositivo.

Timer Module (relative weight- 5%)

Neste módulo estão presentes as funções que desenvolvemos no Lab2. Com estas funções configuramos o dispositivo, as interrupções geradas por ele e mantemos os vários dispositivos sincronizados.

Utils Module (relative weight- 1%)

Neste módulo estão presentes as funções que desenvolvemos no Lab2. Estas funções são utilizadas para todos os dispositivos, nas aulas práticas foram criadas para o Lab2 e depois reutilizadas para todos os outros e no nosso projeto também.

Game Levels Module (relative weight- 13%)

Neste módulo estão presentes as funções que permitem desenhar cada nível.

Game Logic Module (relative weight- 13%)

Neste módulo estão presentes as funções que permitem controlar os vários estados do jogo através dos dispositivos implementados.

Game View Module (relative weight- 13%)

Neste módulo estão presentes as funções que permitem desenhar os menus e o rato, derreter o caminho já percorrido pelo jogador e controlar se o jogador toca em água ou completa o nível derretendo todo o caminho.

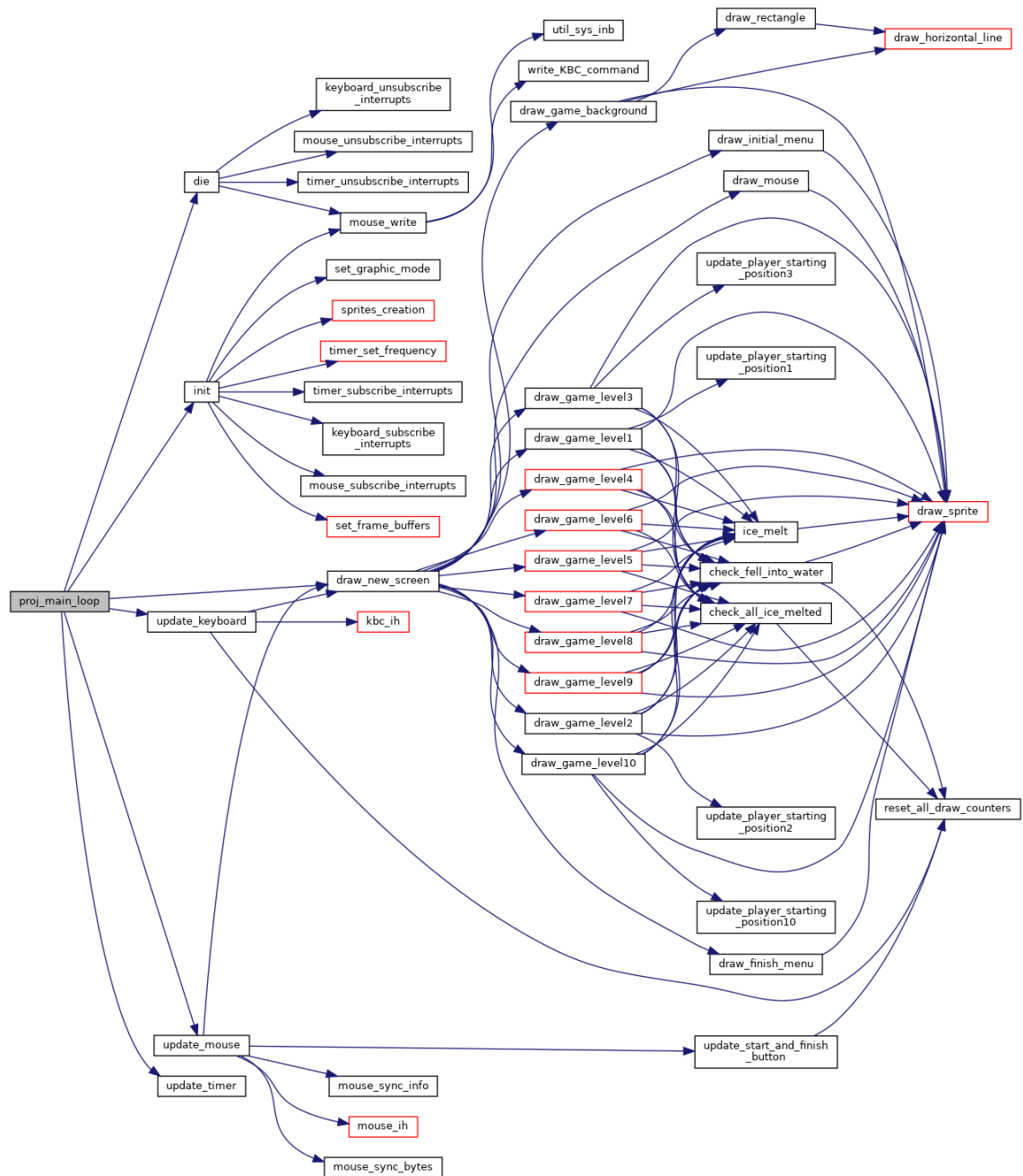
Sprites Module (relative weight- 8%)

Neste módulo estão presentes as funções que permitem criar os elementos usados nos menus e nos níveis.

Main (relative weight- 12%)

Neste módulo está presente o loop principal do projeto que recebe cada interrupção e trata dela.

Function call Graph



Detalhes da Implementação

Funções, não lecionadas, específicas do nosso jogo:

- void ice_melt();
- void check_fell_into_water();
- bool check_all_ice_melted();

O objetivo do jogador em cada nível é chegar ao cubo vermelho, a meta, e ao mesmo tempo conseguir derreter o gelo todo que há pelo caminho.

A função ice_melt() torna possível que, à medida que o jogador se vai mexendo (teclas AWSD), é printado um xpm de bloco de água no bloco de caminho em que ele esteve antes do seu movimento. Temos 2 variáveis globais que ajudam neste processo: playerPosition e copy, playerPosition corresponde sempre à posição mais recente do jogador, e copy irá sempre corresponder à sua posição antes do movimento mais recente.

A função check_fell_into_water() é o complemento lógico para a função ice_melt(), nesta função é verificado se o jogador se desloca para um bloco de água. Para tal, temos uma variável global water_path[250], um array resetado no início de cada nível novo e que pode guardar até 250 posições de blocos de água, quando a função é chamada ela vai iterar por todos os blocos de água no array e caso a posição do mesmo seja igual à posição do jogador, ele cairá à água e o nível começará de novo.

A função check_all_ice_melted() é chamada quando o jogador chega à meta do nível, há 2 variáveis globais que utilizamos para auxiliar esta função: water_counter e path_counter, sempre que um novo frame do jogo é desenhado os valores destas variáveis são atualizados, quando o jogador estiver no cubo vermelho esta função vai verificar se ele derreteu todo o gelo (se o número de blocos de água é igual ao número de blocos de caminho que haviam no início), caso seja igual a função retorna true e o jogador avança para o próximo nível, caso contrário a função retorna false e o jogador começa o nível de novo.

Conclusões

Em suma, o projeto foi uma oportunidade valiosa de aprendizagem, permitindo que o grupo adquirisse conhecimentos práticos na implementação e integração dos dispositivos lecionados, como o teclado, mouse, timer e placa de vídeo. Durante o desenvolvimento, o grupo teve a chance de aprimorar as suas habilidades de programação e compreender a importância da gestão eficiente do tempo e dos recursos disponíveis.

A implementação bem sucedida desses dispositivos essenciais, como o teclado, possibilitou o controle completo do jogo, desde a movimentação do puffle nos níveis até a interação nos menus. O mouse, por sua vez, proporcionou aos jogadores a capacidade de selecionar opções nos menus inicial e final, melhorando a experiência de usuário. Além disso, o timer desempenhou um papel crucial ao controlar o frame rate da placa de vídeo, garantindo um jogo fluído e responsivo, com uma taxa de atualização de 60 frames por segundo.

Além das funcionalidades implementadas com sucesso, também foram identificados alguns "likes to have" que poderiam ser adicionados ao programa em versões futuras. Embora não tenham sido definidos inicialmente, esses recursos adicionais poderiam enriquecer ainda mais a experiência de jogo. Entre eles, destacam-se a implementação do Real-Time Clock (RTC) para fornecer recursos de tempo real, e a possibilidade de fazer o cubo de gelo derreter após a segunda passagem do puffle, o que adicionaria um elemento estratégico ao jogo. Essas melhorias sugeridas apresentam potencial para aumentar a complexidade e o desafio do jogo, proporcionando aos jogadores novas opções e tornando a experiência ainda mais envolvente.

Em resumo, o projeto permitiu que o grupo aplicasse os conhecimentos adquiridos durante as aulas práticas sobre dispositivos como teclado, mouse, timer e placa de vídeo. Essa experiência prática contribuiu para o aprimoramento das habilidades de programação e ressaltou a importância da gestão eficiente do tempo e dos recursos disponíveis. A implementação bem-sucedida desses dispositivos foi fundamental para criar um jogo funcional e oferecer aos jogadores uma experiência satisfatória.