

Trabalho Prático - Dancing Plates

Pedro O.S. Vaz de Melo

May 26, 2023

1 Descrição do Problema

O objetivo deste trabalho é fazer com que o aluno utilize as técnicas de programação aprendidas na disciplina para desenvolver um jogo eletrônico gráfico semelhante ao jogo *Dancing Plates*. *Dancing Plates* foi inspirado pela tradição circense de manter pratos giratórios suspensos em postes. O jogador controla uma garota tentando evitar que um grupo de vários pratos giratórios equilibrados em postes caia. O jogador pode estabilizar pratos se situando imediatamente abaixo do poste correspondente e pressionando um botão específico para estabilizar o prato (exemplo: a tecla espaço). O jogador perde se um prato ficar muito tempo sem ser estabilizado e, com isso, cair no chão.

2 Critérios de Avaliação

Este jogo pode ser tão complexo quanto você deseja, mas há uma versão básica que lhe garante os 20 pontos do trabalho prático. Abaixo as funcionalidades que devem ser implementadas na versão básica do jogo:

- **Controle preciso dos movimentos do jogador.** O jogador deve ter acesso a teclas que controlam a movimentação para esquerda e direita. Além disso, o jogador não deve poder ultrapassar os limites da tela e nem realizar saltos (teletransportes) em qualquer direção. Por fim, enquanto a tecla de movimento é pressionada, o jogador deve se movimentar, somente parando quando a tecla é solta. *(2 pontos)*
- **Pratos.** O seu jogo deve ter pelo menos 8 pratos, que devem estar dispostos de forma regular ao longo da tela, isto é, a distância entre eles deve ser igual. À medida que o tempo passa, o prato perde equilíbrio e você deve indicar isso para o jogador. A maneira mais simples de implementar isso é gradualmente alterar a cor do prato de branco para vermelho. *(2 pontos)*
- **Fim de jogo.** Quando o prato atingir o seu desequilíbrio máximo, você deve fazer com que ele caia do poste. Além disso, o jogo deve terminar. *(2 pontos)*
- **Aparecimento dos pratos.** O jogo deve começar com uma quantidade pequena de pratos (no máximo 3), que são aqueles localizados mais na parte central da tela. À medida que o tempo passa, novos pratos aparecem, na maioria das vezes que o jogo é executado, do centro para as extremidades da tela. Ou seja, os últimos pratos que devem aparecer, na maioria das vezes que o jogo é executado, são aqueles localizados nas extremidades da tela. *(2 pontos)*
- **Equilíbrio dos pratos (parte 1).** Um prato só pode ser equilibrado se o jogador estiver localizado exatamente abaixo dele, ou seja, encostado no poste que o segura. *(2 pontos)*
- **Equilíbrio dos pratos (parte 2).** Ao equilibrar um prato, o prato deve voltar gradualmente ao seu estado original, totalmente equilibrado. A maneira mais simples de exibir isso para o jogador é fazer com que a cor do prato mude gradualmente da atual para o branco original. *(2 ponto)*
- **Equilíbrio dos pratos (parte 3).** Um prato só pode ser equilibrado se o jogador pressionar a tecla que equilibra pratos (ex: espaço) enquanto estiver parado e encostado no poste que sustenta o prato. Se o jogador se movimentar enquanto aperta a tecla, a função de equilibrar o prato deve ser desativada. *(2 pontos)*

- **Equilíbrio dos pratos (parte 4).** Ao equilibrar um prato, algum efeito visual no poste deve ser exibido para o jogador. A maneira mais simples de fazer isso é alterar a cor do poste. *(1 ponto)*
- **Pontuação.** O jogo deve exibir a pontuação do jogo, que deve ser proporcional ao tempo total de jogo. *(1 ponto)*
- **Recorde.** O jogo deve ser capaz de registrar a maior pontuação alcançada por todos os jogadores que o jogaram. O recorde deve ser exibido na tela final do jogo e, caso a pontuação corrente seja maior que o recorde atual, isso deve ser informado para o jogador. *(3 pontos)*
- **Documentação.** Deve conter o manual de uso, que descreve como operar o jogo, e detalhes da implementação, que descreve brevemente os trechos de código e as estruturas de dados desenvolvidas por você. Exemplos de documentação podem ser baixados na página da disciplina ou através do link <https://drive.google.com/open?id=1KP15y2DVEZqTW-Rrhor5SFhGLffG0J0r>. *(2 pontos)*

2.1 Conhecimento do Código

Conhecimento do aluno sobre o código apresentado será verificado via prova oral, que será dada no formato de uma entrevista. Sua nota total será multiplicada pela sua nota da prova oral, que vale 1. Assim, se você tirar 0.5 na prova oral, sua nota será dividida por 2.

2.2 Pontos Extras

Além dos pontos acima, o professor pode atribuir até 10 pontos a mais caso o aluno implemente extras, tais como:

1. Usar imagens e animações do tipo *sprite*;
2. Gerar diferentes tipos de cenários;
3. Permitir diferentes tipos ações de equilíbrio, que produzem efeitos diferentes;
4. Colocar sons e músicas;
5. Implementar animações para o movimento dos pratos;
6. Implementar fases;
7. Implementar a opção de dois jogadores.
8. Criar *addons* e *power-ups* que podem, por exemplo, dar ações especiais para o jogador;
9. Implementar opção de salvar o jogo;
10. Implementar mais de um tipo jogador;
11. Implementar mais de um tipo prato;
12. Implementar menus e diferentes níveis de dificuldade;
13. **Qualquer outro extra que você ache interessante!**

IMPORTANTÍSSIMO: Pontos extras só serão dados aos alunos que obtiveram mais de 50% dos pontos nas provas, ou seja, mais de 36 no somatório das três provas.

3 Como eu faço?

Apesar da descrição fazer o trabalho parecer complicado, ele é bastante simples. Tudo que o aluno precisa saber para desenvolver este jogo são os conhecimentos adquiridos na disciplina e um pequeno entendimento de desenvolvimento de aplicações gráficas. Assim como são necessárias bibliotecas novas para a utilização de funções não nativas da linguagem C, como a `math.h`, uma biblioteca também é necessária para que se utilize funções gráficas. Para este trabalho, pede-se que se utilize a biblioteca Allegro5, que fornece inúmeras funções que podem ajudar no desenvolvimento deste trabalho. Os vídeos abaixo ensinam como instalar a biblioteca Allegro5 em um ambiente Windows com o MingW instalado:

<https://www.youtube.com/watch?v=AezxBP687n8>

<https://www.youtube.com/watch?v=cgqjzJzm00w>

4 Roteiro de Desenvolvimento Sugerido

Como o jogo é complexo, identificar a sequência de funcionalidades que devem ser desenvolvidas pode ser um problema. Assim, a seguir estão descritas etapas de desenvolvimento sugeridas, colocadas em ordem cronológica.

1. Desenhar o cenário.
2. Criar a estrutura do jogador.
3. Desenhar o jogador na tela.
4. Movimentar o jogador.
5. Garantir que ele não saia da tela.
6. Criar a estrutura para os pratos.
7. Desenhar os postes na tela.
8. Desenhar os pratos na tela.
9. Implementar o aparecimento gradual dos pratos.
10. Implementar o desequilíbrio dos pratos (parte 1).
11. Implementar o desequilíbrio dos pratos (parte 2).
12. Implementar o fim de jogo.
13. Implementar o desequilíbrio dos pratos (parte 3).
14. Implementar o desequilíbrio dos pratos (parte 4).
15. Implementar o sistema de pontuação e sua exibição na tela.
16. Implementar o recorde.
17. Escrever a documentação.