

Projeto Prático

Adivinhação de Palavras

Universidade Tecnológica Federal do Paraná (UTFPR), *Campus Apucarana*
Curso de Engenharia de Computação
Fundamentos de Programação 2 – FDC02B - 2º Semestre 2025
Prof. Muriel de Souza Godoi

Nesta atividade os alunos deverão desenvolver um sistema que permita a criação e manipulação de baralhos no estilo “Adivinhação de Palavras”. O sistema deve ser desenvolvido utilizando a linguagem C e os conceitos visto em sala de aula. A atividade deverá ser desenvolvida em trios impreterivelmente.

O sistema desenvolvido deverá obrigatoriamente contemplar, mas não estar restrito, aos seguintes aspectos:

- **Mecânica do Jogo** - Cada equipe deverá escolher uma mecânica de jogo para implementar: 1. Força Clássica 2. Tentativas Limitadas (Wordle Style - letra correta / posição errada) 3. Modo Pluzze (Letras Embaralhadas) 4. Corrida contra o Tempo (limite de tempo para adivinhar; erros custam segundos). Mecânicas diferentes poderão ser aceitas desde que sejam previamente aceitas pelo professor.
- **Tema** - Escolha uma temática para o seu jogo. Exemplos: Futebol, Games, Séries, Lugares, Animes, Cultura Pop/Geek, História da Computação, Linguagens de Programação, Memes, Marvel.
- Para gerenciar uma coleção de perguntas para o jogo, defina uma estrutura **Pergunta** contendo, no mínimo: palavra (texto), uma ou mais dicas (texto), nível de dificuldade. O sistema deverá permitir cadastrar (inserir/ listar/ pesquisar/ alterar/ excluir) as palavras disponíveis. Essa relação deve aumentar e diminuir dinamicamente.
- O mecanismo da sequência de palavras pode ser escolhido pela equipe. Exemplo: várias palavras em nível crescente de dificuldade, pontuação, vidas, tempo, etc.
- Deverá permitir pesquisar e visualizar as palavras por nível de dificuldade e categoria, navegando entre elas. Para isso utilize cores, sons e sua criatividade.
- Deverá apresentar um menu inicial com as opções disponíveis. Caso necessário, submenus. A interface deverá ser fácil e intuitiva, seja criativo, utilize cores e beeps :) . Trate erros do usuário com mensagens e alertas.
- A lista com os dados para as palavras iniciais será construída lendo os dados de um arquivo .csv (valores separados por vírgula) construído pela equipe que deve ser lido e carregado em um vetor de tamanho dinâmico na primeira abertura do programa. Esse arquivo deve ter no mínimo 100 palavras
- Ao sair do programa, todos os dados devem ser salvos em arquivos binários no HD e recarregados novamente ao iniciar. Caso os arquivos não existam, eles devem ser criados e uma mensagem de boas vindas deve ser apresentada ao usuário;
- O sistema deverá exibir no menu uma opção de exportar os dados das questões em um arquivo texto no formato .CSV (separados por vírgula);

Observações sobre o código:

- Os códigos referentes a cada estrutura devem estar em um arquivo fonte (.c) separado com seu respectivo arquivo de cabeçalho(.h). Utilize outros arquivos para organizar seu código conforme necessidade;
 - O programa deverá acompanhar um arquivo makefile que permita as operações básicas de make (compilação), clean (limpeza) e run (execução);
 - Como parte do critério de avaliação, a qualidade do código-fonte do sistema será avaliada. Organize seu programa em arquivos .c e .h, funções. Evite o uso de variáveis globais. O código deverá estar comentado, indentado e com bons nomes de variáveis e funções. Siga o padrão de documentação apresentado na disciplina (Doxygen).
- Parte Criativa (40% da nota)**
 - Partida:** Adicione uma funcionalidade que permita jogar uma partida de Adivinhação de Palavras.
 - Adicione novas regras e funcionalidades ao jogo. Exemplo: diferentes esquemas de dicas e ajudas (podem ser baseadas no tema do jogo), interface gráfica(gtk, raylib), ranking, músicas e sons e etc. Use sua criatividade!
 - Obs:** Para o desenvolvimento da parte criativa, você pode adicionar mais campos ao arquivo .csv com os dados das questões.
 - Entrega via Github**
 - Cada equipe deverá criar um repositório no **Github**, compartilhado entre os membros e com o professor. O link do repositório deverá ser enviado ao professor até dia a data estabelecida na tabela abaixo juntamente com os dados iniciais.

Avaliação: Implementação + Apresentação

Além do envio do código fonte, os(as) alunos(as) deverão realizar uma apresentação projeto que terá duração de 10 e 15 minutos. Durante a apresentação, deverão descrever a abordagem utilizada e apresentar uma visão geral da implementação. Em seguida, deverão mostrar o sistema em funcionamento. Após a apresentação, serão arguidos individualmente pelo professor e pelos demais alunos da turma em relação ao desenvolvimento do projeto. A apresentação será avaliada de acordo com os seguintes critérios:

- Respeito ao tempo estipulado para a apresentação;
- Criatividade na apresentação e no desenvolvimento do sistema;
- Coerência do projeto desenvolvido e o que foi apresentado;
- Domínio do projeto verificado por meio da explicação do código fonte;

15/10	Dados iniciais: membros da equipe, tema do jogo, ideias da parte criativa do projeto (pelo menos 3 ideias de ajudas e funcionalidades) e link do repositório
07/11	Primeira Entrega (parte básica)
26/11	Segunda entrega (parte criativa)
26 e 28/11	Apresentações



Os códigos desenvolvidos por cada equipe serão também verificados por ferramentas de plágio.

Códigos iguais, similares, gerados e/ou copiados da Internet terão nota zero.