

Projeto Prático

Super Trunfo

Universidade Tecnológica Federal do Paraná (UTFPR), Campus Apucarana
Curso de Engenharia de Computação
Fundamentos de Programação 2 – FDC02B - 2º Semestre 2024
Prof. Muriel de Souza Godoi

Nesta atividade os alunos deverão desenvolver um sistema que permita a criação e manipulação de baralhos no estilo “Super Trunfo”. 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:

- Escolha uma temática para o seu Deck: utilize um já existente ou invente um. Exemplos já existentes: Países, Dinossauros, Predadores, Marvel, Gatos, Tubarões, Animais Selvagens, Liga da Justiça, Aves, Mitologia Grega, Disney Princesas, Mitologia Nórdica.
- Para gerenciar um Deck de cartas no estilo Super Trunfo, defina uma estrutura Carta contendo, no mínimo: nome (texto), uma letra(char), um número(int), um booleano para saber se é a carta super trunfo (bool) e mais 4 atributos numéricos (int ou float). Deverá permitir cadastrar (inserir/listar/pesquisar/alterar/excluir) as cartas disponíveis. Essa relação deve aumentar e diminuir dinamicamente. Cada deck tem no mínimo 32 cartas com os códigos compostos por letras de A a D e números de 1 a 8 (ou mais caso necessário).
- Deverá permitir pesquisar e visualizar as cartas por características específicas, por atributo (maior ou menor que um valor), por letra e por número, 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 iniciais para o Deck 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.
- 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 cartas 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, edentado 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 jogar uma partida de supertrunfo contra o computador utilize a regra oficial do supertrunfo como base.
 - Adicione novas regras e funcionalidades ao jogo. Exemplo: partida contra outro jogador local ou remoto, campeonato de partidas com pontuação, interface gráfica(GTK, raylib), 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 cartas.

- **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 20/11 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;

20/11	Dados iniciais: membros da equipe, ideias para a parte criativa do projeto (pelo menos 3 ideias de novas regras/ funcionalidades) e link do repositório
20/12	Primeira Entrega (parte básica)
12/02	Segunda entrega (parte criativa)
12 e 14/02	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.