Universidade Tecnológica Federal do Paraná (UTFPR)
Departamento Acadêmico de Informática (DAINF)
Prof: Rodrigo Minetto (rodrigo.minetto@gmail.com)

Estrutura de dados II
Trabalho Prático

1 Objetivo

O objetivo desse trabalho prático é aprofundar a compreensão em alguns dos tópicos visto no curso: árvores binárias, árvores binárias de busca, árvores balanceadas (AVL), árvores B, filas de prioridades, tabelas de dispersão (hash) ou grafos. Você **deve escolher uma aplicação ou problema** usando algum(ns) destes conceitos. Obs: trabalhos triviais, ou seja, que considerem simplesmente operações (inserção, remoção, seleção...) nestas estruturas receberão nota zero.

Exemplos de aplicações ou problemas interessantes: uso de árvores em jogos, uso de árvores em banco de dados, uso de árvores em compactação de arquivos, uso de árvores em imagens, resolução de expressões, análise sintática, DNS, uso de grafos em redes sociais, estudo de outras árvores que não irão ser vistas no curso (árvores de intervalos (para operações eficientes nesse tipo de aplicação), árvores rubro-negras (concorrentes da AVL), splay tree (cache dos elementos mais acessados), quadtree (uso em imagens), patricia tree (para cadeias de texto), huffman (compactação de arquivos), merkle tree (uso em criptografia),)..., heap binomia, heap de fibonacci, algoritmos em Grafos (com exceção de Busca em profundida e largura que serão vistos no curso), uma lista de problemas interessantes pode ser vista em https://en.wikipedia.org/wiki/List_of_data_structures.

2 Composição do trabalho

O trabalho, realizado de forma individual, é composto pelas partes (enviado através do Moodle em único ZIP):

- Código-fonte (zip contendo somente Makefile, .c e .h).
- **Relatório** (pdf), com a descrição da aplicação ou problema, e como o tópico escolhido aborda o(s) conceito(s) visto(s) na disciplina. É necessário uma descrição dos testes, bem como soluções encontradas **OU** um **vídeo** (que pode ser postado no youtube) com no máximo 7 minutos e que contenha uma apresentação do trabalho (descrição do problema, motivação, etc) com execuções de alguns testes para mostrar o funcionamento do programa.

Modularize o que for possível no seu código, implemente funções razoavelmente curtas, separe a parte de definição das operações (tipo abstrato de dados) da parte de implementação (.c e .h). Crie um **Makefile** para compilar o seu código. Seu código deve ser comentado e indentado. Ao executar o Makefile, devem ser gerados executáveis para cada um dos testes.

Obs1: a falta de Makefile (funcionando no linux) e/ou a presença de qualquer código relacionado a codeBocks (e IDEs ...) implicarão na não correção do seu trabalho (todo o trabalho).

Obs2: a apresentação do trabalho não está descartada, e pode ser usada para identificar se o estudante realmente fez o trabalho.

3 Informações sobre a entrega

- **Data de entrega no moodle** do relatório (ou vídeo no youtube ou enviado em formato avi dentro do ZIP) e código (em arquivo único compactado): 22/Agosto/2021 (domingo) até 23h59.

4 Informações adicionais

- Cópias (com ou sem eventuais disfarces) receberão nota ZERO.
- Trabalho atrasados serão descontados em 50% da nota por dia de atraso.
- O relatório e os códigos devem ser entregues via **Moodle** (se preferir o vídeo pode ser postado no youtube, basta passar o link no ZIP submissão em um README). Identifique o trabalho com o registro acadêmico do aluno (sem espaços): XXXX.{zip} (só esse formato de compactação).
- A nota do trabalho será composta por: 40% referente a implementação e 60% referente ao relatório/vídeo (trabalhos sem código funcionando invalidam o relatório/vídeo).
- Relatórios devem ter capa, descrição do problema, solução, testes, ... A ausência de cada um desses itens reduz em 25% a nota do relatório.
- As notas serão proporcionais a dificuldade do problema, ou seja, problemas mais complexos e soluções mais elaboradas receberão notas mais altas que trabalhos com problemas mais simples.
- **Certifique-se de que os arquivos e o zip entregues funcionam. O trabalho considerado é, necessariamente, aquele entregue via Moodle até a data especificada.**