

Universidade Federal do Ceará - Campus Quixadá
QXD0115 – Estrutura de Dados Avançada – Turma 01A
Curso de Ciência da Computação
Prof. Atílio Gomes Luiz

Primeiro Projeto — Aplicação de Árvores Balanceadas

1. Nesse trabalho, deverá ser desenvolvida uma aplicação que utilize alguma das árvores binárias de busca balanceadas vistas em aula (AVL ou Rubro-Negra).

A aplicação a ser desenvolvida consiste na leitura de um arquivo CSV contendo informações de pessoas e armazenamento destes dados na memória principal de modo a possibilitar rápido acesso aos registros por meio de consultas por campos individuais.

O trabalho deverá passar por três etapas distintas:

- (1) entendimento do problema a ser abordado;
- (2) construção da implementação, documentação, depuração;
- (3) execução e apresentação do trabalho para o professor.

Este trabalho deve, obrigatoriamente, ser feito de forma individual.

Descrição do Problema

Deverá ser desenvolvida uma aplicação que seja capaz de carregar um arquivo de entrada contendo as seguintes informações de pessoas e na seguinte ordem: número do CPF (numérico), primeiro nome da pessoa (string), sobrenome da pessoa (string), data de nascimento da pessoa (data no formato MM/DD/AAAA) e nome da cidade onde nasceu (string).

O arquivo a ser lido estará em um formato CSV, de modo que os campos de cada registro sejam separados por vírgula (,). Seguem algumas linhas de um exemplo de como será o arquivo a ser lido:

```
388.624.732-57,Tiago,Cunha,7/19/1989,Rio Branco
992.809.969-32,Kai,Gomes,10/29/1953,Jundiá
518.376.278-35,Estevan,Azevedo,1/3/1938,Nilópolis
```

O arquivo de exemplo completo está no Moodle, juntamente a esta descrição do projeto. Após carregar o arquivo em objetos da classe **Pessoa**, deverá ser possível realizar as seguintes consultas:

- Consultar uma única pessoa pelo seu CPF e exibir seus dados na tela.
- Consultar todas as pessoas cujo nome comece com uma string informada pelo usuário e exibir na tela todos os dados dessas pessoas na forma de lista.
- Consultar todas as pessoas cuja data de nascimento esteja em um intervalo estabelecido pelo usuário e exibir na tela todos os dados dessas pessoas na forma de lista.

Implementação

O programa deverá ser escrito seguindo as seguintes restrições:

- O programa deve ser desenvolvido na linguagem de programação C++.
- Deve usar o paradigma de programação orientado a objeto.
- **Disposição dos dados na memória:** Os registros lidos do arquivo deverão ser, obrigatoriamente, indexados pelos campos supracitados de modo que todas as consultas possam ser executadas de forma bem rápida. Para isso, será necessária e OBRIGATÓRIA a criação dessa indexação em cada um dos campos já mencionados de uma Pessoa.

Utilize a estrutura de árvore binária de busca balanceada AVL ou Rubro-Negra (tanto faz), sendo que, para cada campo indexado (no caso, CPF, nome da pessoa e data de nascimento), crie uma árvore binária de busca balanceada. Observe que não está sendo solicitada a implementação de três classes de árvores distintas. Nesse ponto, a criação se refere à instanciação da árvore AVL ou Rubro-Negra três vezes, sendo uma voltada para cada campo. Assim sendo, o aluno pode e é incentivado a implementar sua árvore utilizando **templates** e deixando a implementação genérica o suficiente para ser reutilizada em qualquer uso de propósito geral.

- **Observação:** Os registros (objetos **Pessoa**) não deverão estar duplicados na memória. A indexação é feita apenas com chaves e endereços dos registros, e não com cópias inteiras dos registros na memória.
- **Interface com o usuário:** Deverá ser criada uma interface minimamente navegável (via terminal) possibilitando a realização das consultas e visualização dos dados retornados. Não é necessário utilizar quaisquer recursos de interfaces ricas (como QT, GTK, OpenGL, etc). Porém, se você souber usá-las e quiser usar para fazer uma interface gráfica para o seu projeto, pode usar.

Entrega

O trabalho deverá ser entregue em duas etapas: a primeira é a entrega dos arquivos de código fonte via Moodle. Os arquivos de código fonte deverão ser compactados em um único arquivo (ZIP, RAR, 7Z, etc.) e enviados **até o dia 21/05/2023, 23h59**. Deverá conter na raiz do arquivo compactado, um arquivo de nome LEI-AME.TXT explicando como compilar e executar o programa. Adicione scripts de compilação/execução caso seja possível/necessário.

A segunda etapa é a apresentação do trabalho em sala de aula diretamente para o professor.

Apresentação

Todos os trabalhos deverão ser apresentados individualmente em sala de aula nos dias **23 e 24 de maio**. Caso seja necessário mais tempo do que isso para as apresentações, o professor alocará fora do horário de aula, de comum acordo com os alunos.

A ordem de apresentações será definida por acordo/sorteio um dia antes da primeira apresentação.

A apresentação será realizada individualmente e irá considerar a capacidade do aluno de explicar todo o código desenvolvido ou partes específicas determinadas pelo professor. Além disso, poderão ser solicitadas alterações no dia da apresentação que obviamente deverão ser executadas.

Avaliação

A avaliação será realizada em duas etapas: uma referente ao programa e a outra referente à apresentação.

Na avaliação, o professor considerará o funcionamento correto do programa, o atendimento de todos os requisitos e a obrigatoriedade da implementação correta da árvore AVL ou Rubro-Negra.

A apresentação vai considerar a explicação do aluno acerca do código desenvolvido e a capacidade de o aluno responder perguntas sobre o código ou sobre o problema atacado. O professor poderá ainda solicitar alterações no código que o aluno deverá ser capaz de realizar.

Informações adicionais

- Clareza, indentação e comentários no programa serão considerados na nota;
- O trabalho deve ser feito **individualmente**;
- **Não copie trabalho.** Trabalhos copiados receberão nota **ZERO** (para ambas as partes, a que cedeu o código e a que copiou);
- Trabalhos entregues em atraso (por email) **NÃO** serão aceitos, portanto, não deixe para enviar na última hora pois sua internet pode cair. Segundo o regimento da universidade, **NÃO** existe segunda chamada para projetos e trabalhos. Logo, se você não entregar esse trabalho, ficará com nota **ZERO** nesta avaliação.
- Vou reservar dois dias de atendimento (tira-dúvidas) para esse projeto. O local será na minha sala, que fica no bloco 3:
 - dia **03/05/2023** das 13h às 15h
 - dia **10/05/2023** das 13h às 15h