

ACH2023 ALGORITMOS E ESTRUTURAS DE DADOS I

Semestre 2014-2 - Exercício prático 2 - Exclusão de Nível em Árvore Binária

Responsável - Ivandré Paraboni (ivandre@usp.br)

Estagiário PAE - Adilson Khouri (adilson.khouri.usp@gmail.com)

1. O objetivo do trabalho é implementar de forma correta e completa as seguintes funções utilizando uma ABB de inteiros (não é AVL).
2. Verifique no Tidia o modelo de código com as definições (*typedef*) que você deve utilizar.
3. A função recebe como entrada um ponteiro ****raiz** de uma ABB de inteiros e um inteiro **n** representando um nível garantidamente existente na árvore (a raiz está no nível 1).
4. A assinatura da função é a seguinte:

```
void excluirNivel(NO** raiz, int n)
```
5. O objetivo da função é excluir todas as chaves existentes no nível **n**, atualizando a estrutura de acordo. Note que a exclusão do nível 1 modifica o ponteiro ****raiz**, razão pela qual ele é passado por referência.
6. Restrições de implementação:
 - (a) *Não use nenhum vetor na sua implementação.* Estruturas auxiliares de implementação dinâmica (como filas e pilhas) podem ser utilizadas livremente, mas caso haja uso de qualquer tipo de estrutura estática o EP será considerado inválido.
 - (b) Não use variáveis globais. A função implementada deve definir localmente todas as variáveis e estruturas auxiliares, ou chamar funções auxiliares que o façam também em um escopo local.
 - (c) Não usar uma função do tipo "nível" (p.48 da apostila), que seriam extremamente ineficientes se aplicadas à árvore inteira. Seu algoritmo deve percorrer a estrutura uma única vez e excluir o que for preciso, sem fazer chamadas repetidas que percorram a árvore desde a raiz.
7. Não exiba nenhuma mensagem na tela, nem solicite que o usuário pressione nenhuma tecla etc. Apenas implemente a função solicitada.
8. A função **main()** serve apenas para seus testes particulares, e não precisa ser entregue. Caso você prefira mantê-la no corpo do programa, pede-se apenas que **main()** seja a última função do programa, ou seja, que não haja nenhum código abaixo dela.
9. O EP pode ser desenvolvido individualmente ou pelas mesmas duplas do EP1. Novas duplas não serão aceitas.
10. Não tente emprestar sua implementação para outros colegas (desta ou de outra turma), nem copiar deles, pois isso invalida o trabalho de todos os envolvidos.
11. O programa deve ser compilável no Dev-C++ versão 4.9.9.2. sob Windows Vista ou 7.
12. Programadores JAVA, cuidado: não existe inicialização automática de variáveis em C.

O que/como entregar:

- Envie apenas o código da função principal e das funções auxiliares que ela invoca como anexo de nome XXXXXX_YYYYYY, onde XXXXXX e YYYYYY são os números USP dos desenvolvedores. A extensão do arquivo pode ser .c ou .cpp - favor não compactar.
- Preencha as funções nroUSP1 e nroUSP2 do código C disponível no Tidia para que você seja identificado. Se o EP for individual, mantenha o valor do segundo nro. como zeros.
- Na primeira linha do código, escreva um comentário "//" com os nomes dos integrantes.
- O campo "Assunto" da mensagem deverá conter a expressão [AED-02] ou [AED-03] entre colchetes, conforme a turma (matutino=02/vespertino=03) do aluno.
- A mensagem deve ser enviada **simultaneamente** para o docente e estagiário responsáveis pela disciplina, nos endereços indicados no início deste documento.

Prazos e procedimento:

O email contendo o EP deve ser enviado **até 60 minutos antes da data e hora de início da P3**. A confirmação dos EPs recebidos será feita em aula, imediatamente antes do início da P3. Todos os alunos devem ter consigo uma cópia do EP para o caso de ele não ter sido entregue corretamente.

Não serão aceitos EPs entregues depois do prazo, independentemente do motivo. Entregas no último dia são assim por conta e risco do aluno, e nenhum tipo de imprevisto de última hora (e.g., problemas de saúde, indisponibilidade de rede etc.) pode ser usado como justificativa para o atraso. O EP é uma atividade para ser desenvolvida ao longo de um mês, não no último minuto do último dia.

Importante - descontos previstos: -1,0 ponto para qualquer solicitação não atendida

(e.g., emails sem o campo assunto correto, entregas múltiplas (mais de um email), código sem nome ou sem nro. USP etc.).

CrITÉRIOS de avaliação:

A função será testada com uma série de chamadas consecutivas, variando-se o tamanho e conteúdo das árvores de entrada. Um teste é considerado correto se a árvore resultante for exatamente como o esperado, ou incorreto em caso contrário. Erros de alocação de memória ou compilação invalidam o teste, assim como a ausência de funções auxiliares necessárias para a execução do programa.

Este EP deve ser desenvolvido obrigatoriamente por *todos* os alunos de AED1. Sua nota faz parte da 3ª. avaliação da disciplina. A nota do EP **não é passível de substituição**, e é a principal causa de reprovação na disciplina. Caso o EP não seja entregue, isso implica nota zero na 3ª. avaliação independentemente do resultado na P3 ou SUB.