

**ACH2023 ALGORITMOS E ESTRUTURAS DE DADOS I**Prof. Ivandré Paraboni [ivandre@usp.br](mailto:ivandre@usp.br)

Semestre 01/2009

**Trabalho Prático 1 – Matrizes Esparsas**

**Descrição:** A partir do projeto exemplo disponibilizado no COL (*epMatrizesEsparsas.zip*), editar o módulo *trabalho.cpp* implementando as seguintes operações sobre uma estrutura de dados do tipo matriz esparsa:

1. A estrutura a ser manipulada é uma matriz esparsa contendo até 100 x 100 inteiros (posições numeradas de 1 a 100), que deve ser representada através de listas cruzadas de encadeamento simples.
2. O projeto exemplo fornecido contém uma função *main.cpp* que pode ser usada para execução e teste do seu programa, um header *aed1a.h* contendo os tipos de dados manipulados e outras especificações globais, e o módulo *trabalho.cpp* onde deve ser realizado o trabalho propriamente dito.
3. O objetivo do trabalho é implementar de forma correta e completa as 4 funções a seguir, que devem estar inteiramente contidas no módulo *trabalho.cpp*:
  - **bool inserir(MATRIZ \*m, int lin, int col, int ch)**  
Insere o valor *ch* nas coordenadas (*lin,col*) da matriz *\*m*, retornando *true* se a operação for bem sucedida ou *false* em caso contrário (e.g., coordenadas inválidas). Note que a inserção de um valor zero pode na verdade implicar a exclusão do elemento que está nas coordenadas indicadas, se houver um.
  - **int valor(MATRIZ m, int lin, int col)**  
Retorna o valor do elemento na posição indicada (ou zero).
  - **int colunaCoincidente(MATRIZ m, int lin1, int lin2)**  
Retorna o número da primeira coluna que contenha o mesmo valor em ambas as lnhas. Se as linhas forem a mesma, se forem inválidas ou se não houver nenhum elemento coincidente, deve retornar 0. Por exemplo: dadas as linhas (1,1,2,3,4) e (0,2,1,3,4) a coluna coincidente é a quarta, e portanto a função retorna o número 4. Note que valores zero (i.e., elementos nulos) também contam como coincidentes se estiverem na mesma coluna.
  - **void copiarColuna(MATRIZ \*m, int colOrigem, int colDestino)**  
Copia todos os elementos da *colOrigem* na *colDestino*. Os elementos de destino são sobrescritos, o que pode ocasionar uma simples mudança de valor, ou inserções / exclusões.
4. Note que para testar a sua implementação e garantir sua correção você provavelmente terá de criar várias outras funções auxiliares (e.g., entrada e exibição de dados, percurso em linhas e colunas etc.) que não fazem parte do trabalho propriamente dito e que não serão avaliadas.
5. Tenha em mente que tudo que for necessário para executar as funções solicitadas deve obrigatoriamente estar em *trabalho.cpp*, ou seu EP estará incompleto.
6. Além das 4 funções acima, solicita-se que o aluno complete as funções *aluno*, *turma* e *nrousp* também presentes no módulo *trabalho.cpp* para fins de identificação do autor.
7. **IMPORTANTE:** o seu programa será corrigido de forma *automática*, e por isso você não pode alterar as assinaturas das funções solicitadas, nem os tipos de dados ou especificações no header *aed1a.h* que acompanha o projeto exemplo. Pelo mesmo motivo, caso você não implemente alguma das funções solicitadas deve mantê-la com o código vazio em *trabalho.cpp* para que o programa possa ser compilado.

8. Restrições de implementação: seu programa deve ser  *muito eficiente*. Em especial:

- a. Não chame a função *valor* de dentro das demais, pois ela implica nova busca sequencial desde o início da linha/coluna;
- b. Não crie estruturas de dados auxiliares (como vetores ou listas ligadas).

**Modalidade de desenvolvimento:**

Este trabalho é de caráter estritamente individualmente. Por favor não tente emprestar sua implementação para outros colegas, nem copiar deles, pois isso invalida a avaliação de todos os envolvidos.

**Ferramentas de desenvolvimento:**

O programa deve ser compilável no Dev-C++ versão 4.9.9.2. sob Windows XP ou Vista.

**O que entregar:**

Apenas o arquivo *trabalho.cpp* extraído do projeto implementado, observando que ele deve conter todas as rotinas necessárias para a execução das funções solicitadas.

**Como entregar:**

A entrega será realizada via sistema COL até a data e hora de início da P2. EPs entregues após este horário serão desconsiderados.

**CrITÉrios de avaliação:**

- Função *inserir + valor* (serão testadas em conjunto): até 3,0 pontos.
- Função *colunaCoincidente*: até 3,0 pontos (necessita que as funções *inserir* e *valor* estejam corretas).
- Função *copiarColuna*. até 4,0 pontos (necessita que as funções *inserir* e *valor* estejam corretas).

Tendo em vista a necessidade de garantir a correção das funções mais simples para testar as mais complexas, sugere-se fortemente que estas sejam implementadas na ordem acima.

O número de testes realizados na avaliação é necessariamente pequeno, e um programa que esteja funcionando apenas de forma parcial pode facilmente ficar com uma nota muito abaixo do esperado. Por este motivo, sugere-se que o programa seja testado de forma *exaustiva*, com vários tipos de entradas e com especial atenção a casos excepcionais, para assim garantir um resultado satisfatório.

**Penalidade prevista:**

Semelhança com outros trabalhos: nota zero a todos os envolvidos (inclusive entre turmas).

**Lembrete:**

A nota deste EP *não é passível de substituição*, e a avaliação será calculado como o *mínimo* entre EP e a prova correspondente.