BCC202 – Estrutura de Dados I

2s2015

Tipo Abstrato de Dados e Alocação Dinâmica de Memória

Exercícios [1]

Para todos os exercícios, é obrigatório o uso de TAD (*Tipo Abstrato de Dados*) e alocação dinâmica de memória. Lembrem-se, toda memória alocada dinamicamente deve ser também "desalocada" (se necessário, crie uma nova operação para este fim nas TADs descritas a seguir).

1.Implemente um TAD ***ContaBancaria***, com os campos número e saldo no qual os clientes podem fazer as seguintes operações:

a. Iniciar, realizando alocação dinâmica de memória, uma conta com um número e saldo inicial
b. Depositar um valor
c. Sacar um valor igual ou inferior ao saldo disponível
d. Imprimir o saldo

2. Implemente um TAD matriz (de inteiros) e inclua as seguintes operações.

- criarMatriz: criar uma matriz inicializando o seu número de linhas e colunas, realize alocação dinâmica de memória.
- adicionarElmento: dada uma matriz, permite adicionar elementos na posição, ou coordenada, especificada.
- acessarElemento: dada uma matriz e uma das suas coordendas, retorna o elemento que ocupada tal possição.
- getNumeroLinhas: retorna o número de linhas de uma dada matriz.
- getNumerColuna: retorna o número de colunas de uma dada matriz.
- liberarMemoriaMatriz: libera a memória alocada para uma dada matriz.

3. Implemente um TAD ***Circulo***, com os campos centro e raio. Crie as seguintes funções:

a. Criar um círculo, realizando alocação dinâmica de memória, passando um ponto de centro e seu raio
b. Calcular circunferência
c. Calcular área do círculo
d. Dado um ponto, verificar se o mesmo se encontra dentro do círculo
e. Dado um ponto, calcular a distância entre o centro e este ponto.

Obs.: Os pontos (centro, pontos passados nos argumentos) são outra TAD. Cada ponto possui uma coordenada *x e* uma coordenada *y*, ambas inteiras.

4. A *Google* está contratando desenvolvedores para trabalhar no projeto e implementação do tipo abstrato de dados ***Contato,*** que contempla as seguintes operações:

1 Exercícios baseados na lista elaborada pelo Prof. Saul Delabrida em 2s2014.

Observe os seguintes requisitos:

a. Um contato pode conter vários e-mails, eles podem ser classificados entre pessoal e profissional (implemente um TAD e-mail);
b. Um contato pode ter vários telefones. Os telefones têm
   - Código de área
   - Número
   - Operadora (TIM, VIVO, CLARO, OI)
   - Classificação (celular profissional, celular pessoal, fixo profissional, fixo pessoal, etc)

4.1 Crie as funções:

a. Criar um novo contato, realizando alocação dinâmica de memória, passando o nome dele do contato
b. Retornar o nome de um contato
c. Adicionar e-mail a um contato
d. Adicionar telefone a um contato
e. Modificar o nome de um contato

4.2 Crie o tipo de dados agenda e inclua as seguintes operações. Utilize alocação dinâmica de memória.

Agenda* *criarAgenda* (int quantidade_maxima_contatos);
void *insererContato* (Agenda* a, Contato* c);
Contato* *buscarContatoPorNome* (Agenda* a, char* nome);
int *getQuantidadeContatos* (Agenda* a);
int *getCapacidadeAgenda* (Agenda* a);

Obs.: a capacidade da agenda é a quantidade restante de contatos a ser adicionados, ou seja, se a agenda possui inicialmente uma capacidade para 10 contatos e foram inseridos 3 novos contatos a ela, sua capacidade é de 7 contatos.

Procedimento de Entrega

Você deve entregar os exercícios no Moodle até o dia **03/11/15** às **23h55**. Não serão aceitas soluções entregues após o horário estipulado.

Para cada questão:

1- Crie uma pasta para a questão: PrimeiroNome-UltimoNome-Questao (exemplo: jose-silva-1).
2- Crie os arquivos de código fonte necessarios para a solução da questão (arquivos .c e .h)
3- Compile na linha de comando usando g++ *.cpp -o prog.exe
4-. Execute usando redirecionamento de entrada: ./prog.exe < entrada.txt
5- Apague os arquivos gerados na compilação (mantenha apenas os arquivos .c e .h).

Compacte todas as pastas criadas anteriormente num único arquivo (“*.zip”*), que deve ser entregue via Moodle. ***O arquivo compactado deve também conter um arquivo “.txt” com seu nome e número de matrícula.***