

Universidade Federal do Paraná
Departamento de Informática
Disciplina CI056 - Algoritmos & Estruturas de Dados II
2º Período do BCC, IBM e MI
Prof. Elias P. Duarte Jr.

Prova 1 (2014/2 - 10 de outubro de 2014)

- Prova individual sem consulta.
- A interpretação das questões faz parte da prova.
- Clareza, limpeza e legibilidade são itens importantes na correção da prova. Em cada questão, mostre claramente como se chegou ao resultado final!
- A prova pode ser feita a lápis.

1. Mostre o resultado da execução do código abaixo, use figuras para os registros da mesma forma que usamos em sala de aula.

```
struct TipoNode {
    int Valor;
    struct TipoNode *Prox;
    struct TipoNode *Ant;
};
struct TipoNode *aux;
aux = (struct TipoNode *) malloc (sizeof (struct TipoNode));
aux->Ant = (struct TipoNode *) malloc (sizeof (struct TipoNode));
aux->Prox = (struct TipoNode *) malloc (sizeof (struct TipoNode));
aux->Ant->Prox = aux; aux->Prox->Ant = aux;
aux->Ant->Prox->Valor = 100; aux->Valor = 200; aux->Prox->Valor = 300;
```

2. Sobre alocação dinâmica de memória responda:
- (A) Qual a principal vantagem de implementar um Tipo Abstrato de Dados usando alocação dinâmica de memória, em comparação com alocação estática de memória?
 - (B) O que é um *leaking pointer*?
 - (C) O que é um *dangling pointer*?
3. Leia as duas recomendações abaixo. Para cada recomendação reflita e responda: esta recomendação pode ser facilmente evitada ou é obrigatória? Explique suas respostas!
- (a) Ao implementar a fila em vetor recomenda-se que este seja circular.
 - (b) Ao implementar a lista com apontadores recomenda-se use um nodo cabeça.
4. Escreva uma função para checar se os abre e fecha parêntesis de uma expressão estão corretos. Escreva esta função utilizando as operações definidas em sala de aula para os Tipos Abstratos de Dados que considerar necessários.
5. Insistimos em sala de aula para o fato de que para haver um algoritmo recursivo, antes é necessário haver uma definição correspondente recursiva. Responda:
- (A) Considere um vetor com N números inteiros positivos. Escreva uma definição recursiva para somar todos os N elementos deste vetor.
 - (B) Agora escreva a função recursiva correspondente a esta definição.