

**Universidade Federal de Viçosa – *Campus* Rio Paranaíba**

**SIN 211 – Algoritmos e Estruturas de dados**

**Professores:** Gilberto Oliveira e Pablo Munhoz

Observações:

- Entregar os exercícios na “**Área de Entrega de Atividades**” no PVAnet.
- Procure desenvolver todas as atividades, elas possuem grande importância para o bom entendimento do conteúdo da disciplina.
- Caso tenha dificuldades na resolução dos exercícios, tire suas **dúvidas** com o professor ou o monitor.

**Aula Prática 05**

O arquivo submetido ao PVAnet deve conter o cabeçalho com nome, matrícula e turma teórica. Não esqueça de submeter o “arquivo.c” com esses dados. **Obs:** Caso o aluno queira criar o arquivo de cabeçalho “.h”, o mesmo deverá colocar os 2 arquivos em uma pasta e compactá-la antes do envio.

1. Seu “Zé”, dono de uma pequena padaria em Rio Paranaíba, pretende modernizar seu sistema administrativo. Ele deseja que dois sistemas sejam apresentados para ele:
  - a. O primeiro “sistema” é para sua rotação de entrega às lojas da cidade. Cada dia da semana a padaria entrega diferentes lanches às lojas (ex: pão francês, bisnaguinhas, pão de queijo, etc). Cada dia da semana possui como informação apenas o **nome** do alimento a ser entregue. Implemente uma lista simplesmente encadeada circular para resolver esta parte do problema. A lista deve ter as seguintes funções:
    - Inicializar.
    - Lista vazia.
    - Inserir no final, inserir no início.
    - Remover no final, remover no início.
    - Imprimir lista.
    - Liberar lista.

b. O segundo “sistema” é para o padeiro controlar os seus clientes. Cada cliente tem um **nome** e um **saldo** devedor. Para controlar seus clientes, o padeiro exige que o sistema funcione como uma agenda, onde a partir de um cliente, você poderá ir para o próximo cliente ou para o cliente anterior. Para este problema, utilize a lista duplamente encadeada. A lista deve ter as seguintes funções:

- Inicializar.
- Lista Vazia.
- Inserir no final, inserir no início.
- Imprimir a partir do início.
- Imprimir a partir do final.
- Buscar um elemento na lista, imprimir o elemento se ele existir ou imprimir que o elemento não existe.
- Liberar lista.
- Remover no início, remover no final.