

## BCC202 – Estruturas de Dados I (2018-02)

Departamento de Computação - Universidade Federal de Ouro Preto - MG

### Aula Prática 05 – Listas

**Data de entrega: 12/10/2018 até 22:00.**

**Atenção:** *O que vale é o horário do RunCodes, e não do seu, ou do meu, relógio.*

#### Procedimento para a entrega:

1. Para cada questão, implemente sua solução.
  2. Para testar suas soluções, implemente um único método *main()*, que poderá conter, por exemplo, um *Menu* de interações e possibilidades do usuário especificar os dados de entrada.
  3. Especifique o *Makefile* com as instruções necessárias para compilação e execução do seu código, sendo que o *Makefile* deve conter também o redirecionamento da entrada e da saída (e.g., **`./prog.exe < input.txt > output.txt`**.)
  4. Compacte em um único diretório o seu código fonte juntamente com o *Makefile*, o arquivo de entrada e o arquivo de saída usados para testes (e.g., *input.txt*, *output.txt*).
  5. Faça a entrega do arquivo compactado, obrigatoriamente em formato *.zip*, no *RunCodes*, na tarefa correspondente.
- Não utilize caracteres acentuados ou especiais para nomes de pastas, arquivos e na especificação de comentários no código.
  - Implemente em conformidade com boas práticas para reuso e modularização do código.

**- Bom trabalho!**

#### Questão 02

Implemente um TAD lista simplesmente encadeada que contemple a especificação a seguir acrescida das seguintes operações: *void* *lst\_inserir\_ultima\_posicao*(Lista \*l) e *int* *lst\_retira\_ultima\_posicao*(Lista \*l). Sua lista, além de um ponteiro para o primeiro elemento deverá conter também um ponteiro para seu último elemento. Observe que será necessário implementar também o *struct* *ListaNo*, conforme visto em sala de aula. A lista será de *int*. Implemente também um método em seu *main* para exemplificar sua solução.

```
/*TAD: l i s t a d e i n t e i r o s */

typedef struct lista Lista;

Lista* lst_cria (void );
void lst_libera (Lista* l);

void lst_insere (Lista* l, int v);
void lst_insere_ordenado (Lista* l, int v);
void lst_retira (Lista* l, int v);

int lst_vazia (Lista* l);
int lst_pertence (Lista* l, int v);
void lst_imprime (Lista* l);
```