

Estruturas de Dados Básicas I - UFRN 2015.2  
Relatório do Programa ELIS  
Editor de textos orientado a LinhaS

## ELIS

### *Características*

O Projeto ELIS tem como principal característica o uso da estrutura de dados de Lista Encadeada para a resolução de problemas práticos. A aplicação da TAD lista encadeada será na construção de um editor de texto orientado a linhas.

### *Equipe*

Dayane Pabla  
Ana Clara Mendes

Desenvolvedor  
Desenvolvedor

Dezembro/2015

# 1. Resumo

O objetivo deste projeto de programação é utilizar estruturas de dados do tipo lista encadeada para resolução de problemas práticos. O ELIS é um editor de texto orientado a linhas que utiliza essa estrutura de dados TAD lista encadeada para a leitura do que é escrito pelo usuário a cada linha da interface.

## 2. O Projeto

O editor de texto orientado a linhas (ELIS) é um editor sem interface gráfica, seu layout é simples pois utiliza o próprio terminal da máquina como interface. A ideia de executar o projeto foi a de que a estrutura de dados TAD lista encadeada seria utilizada para uma execução real, um problema prático do dia a dia que depende do seu usuário.

O uso da TAD e o seu desenvolvimento foram o que deram origem ao ELIS pois este só funciona se puder receber os dados escritos pelo usuário e os comandos executados por esse para manipulação do arquivo, afinal a ideia é criar um editor de texto. Portanto, a lista é quem instância quem é o nó no momento da leitura, a cabeça da lista e demais constantes iteradoras do arquivo. A aplicação do ELIS foi desenvolvida em diferentes classes e arquivos, porém o arquivo *"application.cpp"* irá trabalhar com os comandos suportados em que o usuário poderá executar ao acessar o programa. São ao todo 10 comandos possíveis a serem executados no ELIS, com o adicional do comando "desfazer/undo", no qual o usuário poderá retomar a última edição feita após apagá-la erroneamente, por exemplo. Portanto o ELIS será capaz de resgatar os dados recém apagados pois um arquivo armazenará os logs do programa, resgatando assim a última linha escrita pelo usuário, por exemplo.

Porém, atualmente o ELIS não está executando todos os comandos em sua totalidade, apenas o comando de salvar, sair do programa, assim como a leitura da escrita do usuário (enter, espaço, apagar o que foi escrito na linha, etc). Por padrão ao usuário escrever qualquer coisa no editor e salvar, o programa irá salvar o arquivo com o nome *"elis.txt"* porém se ele quiser salvar com outro nome é só escrevê-lo, junto com a extensão *".txt"* (pois no momento é o tipo de arquivo trabalhado e lido no ELIS), no momento que executar o comando salvar *":w"*. Portanto, se o usuário escrever o comando *"q"* e não salvar anteriormente, o seu arquivo não será salvo.

O desenvolvimento do projeto todo utilizando a linguagem C++, por ser a mais recente utilizada pelos desenvolvedores do projeto, assim como por facilidade em utilizarmos a biblioteca *curses*. A maior dificuldade encontrada foi fazer com que a lista armazene o que o usuário está digitando para recuperarmos depois e a manipulação desses dados após o arquivo estar salvo, ou seja, a edição do arquivo anterior ou qualquer comando dado quando o programa está sendo executado.

### 3. Layout (interface)

A interface do programa será a do próprio terminal seguindo o exemplo de programas já conhecidos como o Vi. Para que isso fosse possível foi necessário utilizar a biblioteca `curses`, essa provê uma API para o desenvolvimento em modo texto.

```
$ ./elis readme.txt
1> Esta é a primeira linha↵
2> enquanto que esta é a segunda linha↵
3> mais uma linha↵
4> ↵
5> última linha editada↵
```

**Figura 1:** Exemplo de uso do ELIS no modo edição. O símbolo '↵' representa o pressionamento do <ENTER>.