

**UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO - UFES**  
**CENTRO TECNOLÓGICO - CT**  
**CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**GUILHERME SFALSIN SCOPEL**  
**ANDRÉ BARRETO SILVEIRA**

**ADMINISTRAÇÃO DE LEITURA DE CONSUMO DE**  
**ENERGIA ELÉTRICA**

Vitória - ES  
2014

GUILHERME SFALSIN SCOPEL  
ANDRÉ BARRETO SILVEIRA

## **ADMINISTRAÇÃO DE LEITURA DE CONSUMO DE ENERGIA ELÉTRICA**

Estrutura de Dados, Curso de Ciência  
da Computação, Departamento de  
Informática, Centro Tecnológico,  
Universidade Federal do Espírito  
Santo.  
Prof.: Thomas W. Rauber

Vitória - ES  
2014

## SUMÁRIO

<b>1.INTRODUÇÃO .....</b>	<b>4</b>
<b>2.OBJETIVOS .....</b>	<b>5</b>
<b>3.METODOLOGIA .....</b>	<b>6</b>
<b>4.RESULTADOS E AVALIAÇÃO .....</b>	<b>7</b>
<b>5. REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>9</b>

## 1. INTRODUÇÃO

Este projeto foi baseado na linguagem de programação de alto nível C, e através do conhecimento adquirido em Estrutura de Dados I, foi possível construir um programa que, através de um arquivo de entrada, utilizando de lista encadeada pudemos gerar todo um sistema de uma Cidade, desde uma casa, rua, bairro até a Cidade como um todo, podendo assim, por exemplo, medir o consumo de cada região separadamente e obter estatísticas para uma empresa fornecedora de energia elétrica por meio deste.

## **2. OBJETIVOS**

Representação e manipulação de informação estruturada por linguagem de programação de alto nível.

Tem-se que ler um arquivo de entrada de dados, interpretar todos os comandos ali dispostos e processa-los corretamente de modo que no final haja um arquivo de saída demonstrando todos os erros se existirem, e também indicar que a ação dita pela entrada foi concluída com sucesso.

Também é objetivo Demonstrar uma aplicação de algumas estruturas de dados, tal qual o uso de listas, especialmente o caso de listas encadeadas.

### 3. METODOLOGIA

Como aplicação foi considerada a rotina de um leiturista de uma empresa de energia elétrica. Existem cidades com bairros, ruas, casas e/ou prédios com apartamentos. A empresa administra o cadastro dos consumidores e mede o consumo através do leiturista que percorre um roteiro programado.

Sendo assim com o uso de tipos abstratos de dados (TAD's) Foi criado uma biblioteca *lista.h* e o TAD *tadlista.c* para o desenvolvimento de listas encadeadas com o objetivo de estruturar todo o problema proposto.

Além da biblioteca *lista.h* foram utilizadas as bibliotecas *stdio.h*, para a interação com arquivos, tanto a entrada de dados como a saída, *string.h* manipulação de todas as variáveis do tipo *string*, *stdlib.h* para o uso da alocação dinâmica de memória, recurso essencial para o uso e manipulação de estruturas de dados.

Também foi utilizada a ferramenta valgrind para verificação de vazamento de memória e/ou a não liberação do espaço de memória de um ponteiro alocado dinamicamente.

## 4. RESULTADOS E AVALIAÇÃO

Foram usados diferentes tipos de arquivos de entrada como meio de uma bateria de teste, para assegurar que o sistema é robusto em relação a erros,

Aqui temos a imagem de um arquivo de teste que foi utilizado

```
Arquivo testel.txt:
rua incluir    17 3 "Av. Fernando Ferrari"
rua incluir    17 3 "Av. Fernando Ferrari"
rua incluir    13 3 "Rua Carlos Martins"
rua incluir    13 7 "Av. Munir Hilal"
rua eliminar   13 7
rua eliminar   13 7
rua incluir    17 7 "Av. Munir Hilal"
casa incluir   13 7 8 1 123 "Jose Manuel Sacramento"
casa incluir   17 7 9 34 111 "Diogo Sacramento"
casa incluir   13 7 2 99 133 "Rodrigo Sacramento"
casa eliminar  13 7 8
casa consumir  17 7 9 255
bairro medir   17
cidade medir
```

E com Sucesso todos os erros de entrada foram tratados, e foram processados os dados corretos para gerar o arquivo de saída.

```
Arquivo saida.txt:
Rua 'Av. Fernando Ferrari' id=3 incluída com sucesso no bairro id=17
ERRO: Rua não incluída. (Rua id=3, Bairro id=17)
Rua 'Rua Carlos Martins' id=3 incluída com sucesso no bairro id=13
Rua 'Av. Munir Hilal' id=7 incluída com sucesso no bairro id=13
Rua id=7 removida com sucesso do bairro id=13
ERRO: Rua não removida. (Rua id=7, Bairro id=13)
Rua 'Av. Munir Hilal' id=7 incluída com sucesso no bairro id=17
ERRO: Casa não incluída. (Casa id=8, Rua id=7, Bairro id=13)
Casa id=9 de 'Diogo Sacramento' incluída com sucesso na Rua id=7 do Bairro id=17
ERRO: Casa não incluída. (Casa id=2, Rua id=7, Bairro id=13)
ERRO: Casa não removida. (Casa id=8, Rua id=7, Bairro id=13)
Consumo 255.00 atribuído a Casa id=9, Rua id=7, Bairro id=17
Consumo do Bairro = 255.00 (Bairro id=17)
Consumo da Cidade = 255.00
```

Então pudemos utilizar da ferramenta de depuração *valgrind* para verificar a existência de algum erro de alocação de memória ou também vazamento da mesma, com sucesso nenhum erro foi obtido e também todo espaço de memória alocado foi liberado.

```
==11036== Memcheck, a memory error detector
==11036== Copyright (C) 2002-2013, and GNU GPL'd, by Julian Seward et al.
==11036== Using Valgrind-3.10.0.SVN and LibVEX; rerun with -h for copyright info
==11036== Command: ./programa
==11036==
Digite o nome do arquivo de entrada: teste1.txt
Arquivo 'saida.txt' criado.
==11036==
==11036== HEAP SUMMARY:
==11036==     in use at exit: 0 bytes in 0 blocks
==11036==   total heap usage: 156 allocs, 156 frees, 3,650 bytes allocated
==11036==
==11036== All heap blocks were freed -- no leaks are possible
==11036==
==11036== For counts of detected and suppressed errors, rerun with: -v
==11036== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
```



## **5. REFERÊNCIAS BIBLIOGRÁFICAS**

Introdução a Estruturas de Dados, Ed11, Waldemar Celes