



**UNIVERSIDADE FEDERAL DO CEARÁ - UFC**  
**CAMPUS CRATEÚS**  
**CIÊNCIA DA COMPUTAÇÃO**

**Disciplina:** Estrutura de Dados

Prof. Roberto Cabral

**Dupla:** - José Tarcísio De Sousa Araújo Neto| Matrícula: 402542

- Samuel Artur Sales Basilio| Matrícula: 402550

**RELATÓRIO TÉCNICO**

**1. INTRODUÇÃO(Descrição):**

O objetivo do trabalho é implementar um sistema de armazenamento de números inteiros de tamanho qualquer, utilizando uma estrutura de dados que armazena caracteres, logo a concatenação de cada lista da estrutura se tornará o inteiro desejado.

E assim, serão realizadas operações de soma, quadrado, multiplicação, fatorial, resto, mdc e comparação, entre uma ou duas listas duplamente encadeada. O código fonte, foi organizado em dois arquivos, lista.c, contendo as operações básicas de uma lista duplamente encadeada, e o operacoes.c, contendo as operações matemáticas, que serão realizadas sobre as listas.

**2. DIVISÃO DO TRABALHO:**

A divisão do trabalho foi realizada de acordo com os conhecimentos de cada membro da dupla, sempre um buscando cooperar e ajudar o outro, as funções foram implementadas da seguinte forma:

**Lista.h:**

<b>Função</b>	<b>Integrante</b>
Lista* cria();	Samuel Artur
Lista* inserir(Lista *l, char c);	Samuel Artur

int vazia(Lista *l);	Samuel Artur
Lista* inserir_fim(Lista *l, char x);	Tarcisio Neto
void imprimir(Lista *l);	Tarcisio Neto
void libera(Lista *l);	Tarcisio Neto
int tamanho_lista(Lista *l);	Tarcisio Neto
Lista* ultimo_no(Lista *l);	Tarcisio Neto
Lista* proximo_no(Lista *l);	Tarcisio Neto
Lista* no_anterior(Lista *l);	Tarcisio Neto
Lista* nova_igual(Lista *l);	Tarcisio Neto
char getValor(Lista *l);	Samuel Artur
void setValor(Lista *l, char c);	Samuel Artur

### **Operacoes.h:**

<b>Função</b>	<b>Integrante</b>
char converteParaChar(int n);	Samuel Artur
int converteParaInt(char c);	Samuel Artur
Lista* le_numero(Lista *l, int n);	Samuel Artur
Lista* soma(Lista *aux1, Lista *aux2, Lista *l3);	Tarcisio Neto, Samuel Artur
Lista* adicao(Lista *l1, Lista *l2);	Tarcisio Neto, Samuel Artur
int comparacao(Lista *l1, Lista *l2);	Samuel Artur
Lista* multiplicar(Lista *l1, Lista *l2);	Tarcisio Neto
Lista* multiplicacao(Lista *l1, Lista *l2);	Tarcisio Neto
Lista* quadrado(Lista *l);	Samuel Artur
Lista* maior_divisor(Lista *l1, Lista *l2);	Tarcisio Neto
Lista* mdc(Lista *l1, Lista *l2);	Tarcisio Neto
Lista* fatorial(Lista *l);	Tarcisio Neto
Lista* exponenciacao(Lista *l1, Lista *l2);	Tarcisio Neto, Samuel Artur
Lista* modulo(Lista *l1, Lista *l2);	Tarcisio Neto, Samuel Artur

### **3. RESOLUÇÃO DO PROBLEMA:**

Um dos problemas iniciais foi quebrar e converter um número inteiro para caractere, e vice-versa, para realizar essa funcionalidade, foi implementada funções destinadas a realizar essas conversões. Logo, foi implementadas funções que realizam operações como, multiplicação, adição e mdc.

### **4. DIFICULDADES ENCONTRADAS:**

Uma das principais dificuldades foi organizar o TAD de maneira correta, separando as operações, das funções básicas de listas duplamente encadeadas, e assim, fazer com que o sistema funcione corretamente.

## **5. MANUAL: COMO FUNCIONA O SISTEMA:**

O sistema funciona com uma lista duplamente encadeada, armazenando caracteres, e a concatenação de todas as lista da estrutura resulta em uma lista completa, que representa um número inteiro, o usuário digita o inteiro desejado, o mesmo é convertido para caractere e armazenado na lista.

Logo em seguida, o usuário pode selecionar a função desejada, e realizar a operação. Para compilar o programa basta entrar no terminal do linux, acessar a pasta do projeto e digitar make, logo a função principal(main) será executada.