

Giullio Emmanuel da Cruz Di Gerolamo

RA: 790965

Frequência F6

Arquivo Listadupla.cpp

```
#include <iostream>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define MALLOC(a) (a *) malloc ( sizeof(a) )

struct no
{
    int info;

    struct no *prox;
    struct no *ant;
};

// ponteiros de referência
struct no *inicio;
struct no *fim;

/*
* inserir - insere um novo dado na lista
*
* @ info - dado a ser inserido
*/
void inserir ( int info )
{
    struct no *novo = MALLOC ( struct no );
    struct no *atual;

    if ( !novo )
    {
        perror ( "Malloc: " );
        return ;
    }
}
```

```

// atribuição do novo valor...
novo->info = info;

// cria lista
if ( !inicio )
{
    novo->prox = NULL;
    novo->ant = NULL;

    inicio = novo;
    fim = novo;

    return ;
}

// se não for o primeiro elemento da lista...
atual = inicio;

while ( atual )
{
    if ( atual->info < info )
        atual = atual->prox;
    else
    {
        // elemento intermediário - caso 2
        if ( atual->ant )
        {
            novo->prox = atual;
            novo->ant = atual->ant;
            atual->ant->prox = novo;
            atual->ant = novo;

            return ;
        }
        // novo primeiro elemento - caso 1
        novo->prox = atual;
        novo->ant = NULL;
        atual->ant = novo;
        inicio = novo;

        return ;
    }
}
// novo último elemento - caso 3
fim->prox = novo;
novo->prox = NULL;
novo->ant = fim;

```

```

    fim = novo;

    return ;
}

/*
 * imprimeLista - imprime todos os dados da lista
 */
void imprimeLista ( void )
{
    struct no *atual = inicio;

    while ( atual )
    {
        printf ( "Info: %.2d\n", atual->info );
        atual = atual->prox;
    }
    return ;
}

/*
 * procurar - procura um elemento na lista
 *
 * @ info - dado a ser pesquisado
 *
 * Retorno: ponteiro para o dado encontrado ou NULL caso não encontre
 */
struct no *procurar ( int info )
{
    struct no *atual = inicio;

    while ( atual )
    {
        if ( atual->info == info)
            return atual;
        else
            atual = atual->prox;
    }
    return NULL;
}

/*
 * remover - remove um nó da lista
 *
 * @ dado - endereço do nó a ser removido
 */
void remover ( struct no *dado )
{

```

```

if ( !dado ) return ;

// item intermediário
if ( dado->prox && dado->ant )
{
    dado->ant->prox = dado->prox;
    dado->prox->ant = dado->ant;

    free ( dado );
    return ;
}

// primeiro item
if ( dado == inicio )
{
    inicio = dado->prox;
    inicio->ant = NULL;

    free ( dado );
    return ;
}

// último elemento
if ( dado == fim )
{
    fim = dado->ant;
    fim->prox = NULL;

    free ( dado );
    return ;
}
}

// main...
int main ( void )
{
    register int i;
    inicio = fim = NULL;

    for ( i = 1; i <= 10; i++ )
        inserir (i); // caso 3 - inserção no final...

    imprimeLista(); puts ("");

    inserir (0); // caso 1
    inserir (12); // caso 3
    inserir (11); // caso 2

```

```
imprimeLista (); puts ("");

// teste da função de procura
printf ("Procurar( 4 ): %.2d\\n\\n", procurar(4)->info );

remove ( procurar ( 0 ) ); // caso 1
remove ( procurar ( 5 ) ); // caso 2
remove ( procurar (12) ); // caso 3

imprimeLista(); puts ("");

return 0;
}
```

The screenshot shows the Dev-C++ IDE with the following components:

- Main Window:** Displays the source code for `main.cpp`. The code defines a linked list structure with a `Node` struct and a `LinkedList` struct. It includes functions for creating, inserting, deleting, and traversing the list. The `main` function provides a menu for the user to interact with the list.
- Compiler Window:** Shows the compilation results, indicating that the program was compiled successfully without any errors or warnings.
- Debug Window:** Shows the execution output, displaying the menu and user input. The output shows the menu being displayed, and the user inputting '1' to create a new node.