# INSTITUTO FEDERAL DO RIO GRANDE DO NORTE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS ALGORITMOS

## FELIPE MATEUS FREIRE PONTES FRANCISCO BENTO DA SILVA JÚNIOR

### IMPLEMENTAÇÃO INTERATIVA DE UMA PILHA

Neste trabalho, desenvolvemos um programa interativo para realizar operações com uma pilha de valores na linguagem C. Segue a explicação do código fonte utilizado:

#### stack.h

A seguinte estrutura é definida como nó padrão para a implementação da pilha:

```
struct Node
{
  int value;
  struct Node * next;
};
```

Temos também as seguintes variáveis definidas:

```
struct Node Stack_nodes[STACK_SIZE];
struct Node * Stack_lastOne;
unsigned int Stack length;
```

Respectivamente, correspondem ao vetor de nós (STACK\_SIZE é uma constante definida), ao último nó adicionado na pilha e o tamanho da mesma.

#### stack.c

```
void Stack Init(int initialValue)
```

Esta função é responsável por receber o valor inicial da pilha e configurar as variáveis (Stack\_lastOne e Stack\_length) para os métodos seguintes.

```
int Stack Length()
```

Esta função retorna o tamanho da pilha.

```
void Stack List()
```

Esta função é responsável por escrever a pilha na tela do usuário por meio do printf.

```
int Stack Pop()
```

Esta função é responsável por remover o último nó empilhado e retornar seu valor. Caso a lista esteja vazio, 0 é retornado para não quebrar a aplicação, porém quem chamar a função é responsável por controlar o limite da pilha.

```
int Stack Push(int value)
```

Esta função é responsável por empilhar valores na pilha. Há três possíveis retornos:

- -1 indica pilha não iniciada.
- 1 indica empilhamento realizado.
- 0 indica limite de elementos alcançado (STACK SIZE comentado anteriormente).

```
int Stack ToFile(const char filename[])
```

Esta função é responsável por salvar a pilha atual num arquivo dado pelo nome de arquivo recebido.

#### program.c

```
void showMenu()
```

Esta função é responsável por escrever as opções do menu para o usuário por meio do printf.

```
int handleOption(int option)
```

Esta função é responsável por mapear as opções do menu às operações disponíveis na pilha como empilhamento, desempilhamento, listar entre os outros citados anteriormente.

```
int handleMenu()
```

Esta função é responsável por garantir a interação com o usuário chamando handleOption() e showMenu(), enquanto ele não opta por sair.

```
int main(int argc, char const *argv[])
```

Esta função é responsável por chamar o handleMenu () e retornar o retorno.