

Complexidade de Tempo das Funções  
Enya Luísa Gomes dos Santos  
UFOP - Universidade Federal de Ouro Preto  
Ciência da Computação - Estrutura de Dados I  
Prática 05 - Listas

**//vetor.c**

//Essa função é  $O(1)$

Lista \*lst\_cria();

//Essa função é  $O(n)$

void lst\_imprime(Lista \*);

//Essa função é  $O(n)$

int lst\_pertence(Lista \*, int);

//Essa função é  $O(n)$

void lst\_insere\_ordenado(Lista \*, int);

//Essa função é  $O(n)$

void lst\_retira(Lista \*, int);

//Essa função é  $O(n)$

void lst\_liberar(Lista \*)

**//lista\_vetor.c**

//Essa função é  $O(1)$

Lista \*lst\_cria();

//Essa função é  $O(\log n)$

int buscaBinaria(int \*array, int begin, int end, int valor);

//Essa função é  $O(n)$

void lst\_imprime(Lista \*l);

//Essa função é  $O(\log n)$

int lst\_pertence(Lista \*l, int valor);

//Essa função é  $O(n)$

void lst\_organiza(int \*array, int begin, int end, int valor);

//Essa função é  $O(n)$

```
int lst_busca_index(int *array, int left, int rigth, int valor);
```

```
//Essa função é  $O(1)$ 
```

```
void lst_insere_ordenado(Lista *l, int valor);
```

```
//Essa função é  $O(n^2)$ 
```

```
void lst_retira(Lista *l, int valor);
```

```
//Essa função é  $O(1)$ 
```

```
void lst_liberar(Lista *l);
```