

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

```
//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  $O(1)$ 
```

```
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 right, 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);
```

```
//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 *)
```