

## ESTRUTURAS DE DADOS

### TRABALHO 3 – ÁRVORE BINÁRIA DE BUSCA

Escreva uma função que receba como único parâmetro um ponteiro para um vetor de caracteres (um array de array de caracteres), onde cada elemento desse array é uma palavra terminada com o caractere '\0'.

O final da lista de palavras é marcado pelo vetor "\0".

O protótipo da função deverá ser:

```
void processaPalavras(char *str[]);
```

Essa função deverá montar uma árvore binária de busca que conta a quantidade de vezes que cada palavra aparece.

Ao final da execução a função deverá mostrar todas as palavras, em ordem alfabética, e mostrar a quantidade de vezes que cada palavra aparece.

Exemplo: Suponha que sejam fornecidas as palavras: "carro", "navio", "carro", "aviao", "carro", "camelo", "carro", "motocicleta", "carro", " navio", " aviao", "carro" e "navio".

O programa deverá mostrar como saída:

```
aviao (2)  
camelo (1)  
carro (6)  
motocicleta (1)  
navio (3)
```

Assumir que cada palavra terá, no máximo, 15 letras.

Entrega:

- O trabalho deverá ser entregue até a data estipulada no Teams.
- Trabalhos entregues após esta data, se aceitos, sofrerão redução da nota.
- Trabalhos iguais ou com conteúdo muito próximo serão desconsiderados. **Julgue se vale a pena disponibilizar seu trabalho para “ajudar” outros.**
- O trabalho deverá ser enviado via Teams em .c ou .txt.