

## MAC 323 – Algoritmos e Estruturas de Dados II

Primeiro semestre de 2022

### Tabela de Símbolos – Entrega: 16 de maio

O objetivo deste exercício-programa é testar diferentes implementações para uma tabela de símbolos ordenada. As operações a serem implementadas são:

- void add (Key key, Item val);
- Item value (Key key);
- int rank (Key key);
- Key select (int k);

Você deverá fazer vários testes com diferentes implementações para tabelas de símbolos. Você deverá, **pelo menos**, implementar a tabela de símbolos usando:

- vetor dinâmico ordenado;
- árvore de busca binária
- treaps
- árvores 2-3
- árvores rubro-negras

Para cada uma destas implementações vocês deverão fazer testes empíricos para analisar o tempo que cada uma das 4 operações acima executa.

Em seus testes a tabela de símbolos deverá contabilizar a frequência com que palavras ocorrem em um texto muito grande. Para os textos grandes você pode usar os disponíveis no Projeto Gutenberg. Aí você encontra grandes livros escritos em várias línguas. Teste com outros tipos de dados, como dicionários, códigos muito grandes e textos gerados aleatoriamente, no estilo *lorem ipsum*.

Neste exercício-programa o relatório é muito importante. Faça vários testes para mostrar o desempenho de cada implementação.

## Descrição da entrada para os testes

Um caso teste do EP2 consiste de diversas linhas de entrada. A primeira linha indica a estrutura a ser testada (V0, ABB, TR, A23, ARN). Em seguida, na segunda temos um inteiro  $N$  com o número de palavras do texto. Então, temos  $N$  palavras que constituem o texto, que poderão ocupar diversas linhas do input. Logo após o texto, temos um inteiro  $Q$ , que representa o número de operações a serem realizadas na tabela de símbolos. Em sequência, temos  $Q$  linhas, cada uma com uma instrução do tipo:

- 1 **x** - Adicione as próximas **x** palavras do texto a tabela de símbolos
- 2 **s** - Quantas vezes a palavra **s** apareceu no texto até o momento - operação value(s).
- 3 **s** - Quantas palavras são menores que **s** - operação rank(s).
- 4 **k** - Qual a **k**-ésima chave da tabela - operação select(k);

Para as operações do tipo 2, 3 e 4 seu programa deverá imprimir uma resposta.

---

### Caso teste 1: Exemplo abstrato

---

```
1 E // Estrutura a ser testada
2 N // Número de Palavras do texto
3 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis
4 feugiat tellus vel dui varius mollis ac id lacus. Donec vitae
5 mauris a tellus semper. // Texto qualquer com N palavras
6 Q // Número de operações a serem feitas
7 Q linhas com uma das 4 operações descritas anteriormente
```

---

Veja os casos da próxima seção para melhor entendimento.

## Exemplos

---

### Caso teste 2: Machado de Assis

---

```
1 V0
2 18
3 ao verme que primeiro roeu as frias carnes do meu cadaver dedico
4 como saudosa lembranca estas memorias postumas
5 3
6 1 18 // Adiciona todo o texto
7 4 0
8 3 as
// Fim da Entrada
// Saída
9 ao // Esta é a menor chave em todo o texto
10 1 // Somente a chave 'ao' vem antes de 'as'
```

---

---

**Caso teste 3: BBB21**

---

```
1 A23
2 29
3 voce e um orgulho para esse reality e um dos maiores personagens da
4 historia do programa voce e indestrutivel e tudo ja esta bem volte
5 sempre gil do vigor
6 4
7 1 4 // Adiciona as palavras: voce, e, um, orgulho
8 2 voce
9 1 25 // Adiciona o restante do texto
10 2 voce
// Fim da Entrada
// Saída
11 1
12 2
```

---

---

**Caso teste 4: Aleatório**

---

```
1 ARN
2 20
3 a aa aaa aa aaa aa aa aaaa aa a bbb b bbbbbb aaaa a a a a a
4 8
5 1 4
6 2 aa
7 3 aaa
8 4 2
9 1 5
10 2 aa
11 1 11
12 3 b
// Fim da Entrada
// Saída
13 2
14 2
15 aaa
16 5
17 4
```

---