1) Faça um programa simples e legível, em qualquer linguagem de programação, mas sem uso de bibliotecas "prontas" ou "externas", que

- leia um arquivo de texto e receba um parâmetro N > 0
- gere uma saída do arquivo dado com as frequências ordenadas dos N-gramas (https://en.wikipedia.org/wiki/N-gram)

Por exemplo, dado o seguinte arquivo em um terminal linux:

```
$ cat texto.txt
Se cada um vai a casa de cada um é porque cada um quer que cada um vá lá
Porque se cada um não fosse a casa de cada um é porque cada um não queria que cada um fosse lá
```

Podemos chamar o programa e calcular os unigramas com N=1 (também conhecido como contador de palavras)

```
$ ./ngram texto.txt 1
8 - cada
8 - um
3 - porque
2 - a
2 - casa
2 - de
2 - fosse
2 - lá
2 - não
2 - que
2 - se
2 - é
1 - quer
1 - queria
1 - vai
1 - vá
```

Ou calcular os bigramas com N=2

```
$ ./ngram texto.txt 2
8 - cada um
2 - a casa
2 - casa de
2 - de cada
2 - porque cada
2 - que cada
2 - se cada
2 - um não
```

```
2 - um é
2 - é porque
1 - fosse a
1 - fosse lá
1 - não fosse
1 - não queria
1 - porque se
1 - quer que
1 - queria que
1 - um fosse
1 - um quer
1 - um vai
1 - um vá
1 - vai a
1 - vá lá
```

2) Refaça o programa anterior, na mesma linguagem usada, ainda sem uso de bibliotecas "prontas" ou "externas", mas ao invés de focar em código simples e legível, foque em um código performático, isto é, que tenha pelo menos algumas das seguintes características:

- Menor uso de cpu
- Menor uso de memória ram
- Menos operações de entrada/saída (I/O)
- Menor complexidade algorítimica segundo a notação big O (https://en.wikipedia.org/wiki/Big_O_notation)

3) Refaça um dos programas anteriores, mas agora podendo usar qualquer tipo de biblioteca (provavelmente o que você faria em um ambiente profissional)

Comentários:

Nota 1: é desejável, mas não necessário se preocupar com pontuação, números e outros caracteres especiais

Nota 2: é desejável que os parâmetros sejam recebidos como argumentos na linha de comando, mas é também aceitável eles serem recebidos pela entrada padrão, por chamada de api rest ou em uma interface web

Bonus point 1: faça testes unitários
Bonus point 2: faça um benchmark dos programas comparando-os em entradas de diferentes tamanhos
Super bonus point: escreva o programa usando algum framework (ex: spark) ou data warehouse (ex: big query) que permita fazer os cálculos de forma distribuída e escalável