

## Contagem dos Itens Mais Frequentes (Algoritmo de Metwally et al)

João Ferreira, 80041, Mestrado em Engenharia Informática

**Resumo** – Este relatório aborda o desenvolvimento do algoritmo de Metwally et al. para a contagem dos caracteres mais frequentes em ficheiros de texto e foi desenvolvido no âmbito da UC Algoritmos Avançados da Universidade de Aveiro.

**Abstract** - This report addresses the development of the algorithm by Metwally et al. for counting the most frequent characters in text files and was developed within the course Advanced Algorithms of the University of Aveiro .

### I. INTRODUÇÃO

Este projecto visa determinar os itens mais frequentes de um conjunto de dados, explorando métodos que permitem processar conjuntos de dados de grande dimensão. Para atingir esse objectivo foi implementado, utilizando a linguagem de programação Python 3, o algoritmo de Metwally et al. Space-Saving-Count e também um contador exacto.

De modo a analisar o método desenvolvido foram utilizados ficheiros de texto de obras literárias (em Português e Inglês), retirados da biblioteca de eBooks Project Gutenberg. [1]

O método desenvolvido foi utilizado para determinar as letras mais frequentes e, numa fase final, para compara os resultados obtidos com os resultados do contador exacto.

### II. CONTADOR EXACTO

Inicialmente, são lidos todos os caracteres presentes no ficheiro de texto passado como argumento ao programa. De modo a realizar esta leitura de caracteres, são percorridas todas as linhas do ficheiro, sendo removidos os espaços em branco (método strip) e posteriormente a respectiva linha separada palavra a palavra (método split). Após, este processo, é realizada uma iteração pela palavra (caractere a caractere) e, no caso do caractere ser uma letra ou um número (isto é, se não for um caracter especial como, por exemplo, um ponto final), este caracter é adicionado a uma lista, que será retornada no fim desta função. Este processo pode ser desenvolvido pelo código exibido na figura seguinte.

```
def read_file(text_file):
    """
    returns an array of chars
    """
    chars = []
    try:
        with open(text_file, 'r') as file:
            for line in file:
                for word in line.strip().split():
                    for l in word:
                        if l.isalnum():
                            chars.append(l)
    except:
        logger.error('File not found.')
        sys.exit(2) #Code 2 - Command line syntax errors

    logger.info('Text file read with success.')

    return chars
```

Fig. 1 – Ler ficheiro

Tal como no segundo projecto desta Unidade Curricular, foi desenvolvido um contador exacto. Este contador retorna um dicionário no formato <letra, contagem> e começa por iterar sobre um array com todos os caracteres presentes no ficheiro de texto, array esse que foi retornado pela função read\_file.

Iterando sobre esse array, a função começa por transformar o caractere em questão num caractere equivalente mas minúsculo, recorrendo ao método lower. Após esta transformação, verifica se já existe essa key no dicionário a ser retornado e, em caso afirmativo, incrementa o valor que já lhe está associado. Caso contrário, guarda esse caractere com o valor 1 associado. Este processo pode ser desenvolvido pelo código exibido na figura seguinte.

```
def exact_counter(chars):
    """
    returns a dictionary <char,count>
    """
    counter = {}
    for char in chars:
        char = char.lower()
        if char in counter:
            counter[char] += 1
        else:
            counter[char] = 1

    logger.info('Completed Exact Counter.')
    return counter
```

Fig. 2 – Contador exacto

### III. ALGORITMO DE METWALLY ET AL.

O algoritmo de Metwally et al. - Space Saving Count é um dos algoritmos de data streaming, sendo que estes algoritmos processam data streams nos quais os inputs são definidos com uma sequência de itens e podem ser processados em poucos passos, tendo em conta que, maioritariamente, estes não possuem grande capacidade de memória.

Assim, o algoritmo Space Saving Count opera de modo a não utilizar muita memória.

#### Algorithm 3: SPACE SAVING( $k$ )

```

 $n \leftarrow 0;$ 
 $T \leftarrow \emptyset;$ 
foreach  $i$  do
     $n \leftarrow n + 1;$ 
    if  $i \in T$  then  $c_i \leftarrow c_i + 1;$ 
    else if  $|T| < k$  then
         $T \leftarrow T \cup \{i\};$ 
         $c_i \leftarrow 1;$ 
    else
         $j \leftarrow \arg \min_{j \in T} c_j;$ 
         $c_j \leftarrow c_j + 1;$ 
         $T \leftarrow T \cup \{i\} \setminus \{j\};$ 

```

Fig. 3 – Algoritmo Space Saving Count [2]

O algoritmo em questão itera sobre uma lista de caracteres, extraída de um ficheiro de texto e regista o número de vezes que ocorrem, mas, com o intuito de poupar memória, é definido um limite para o número de contadores a guardar ( $k$ ). Sempre que o limite de contadores for atingido, o algoritmo substitui o menor contador com o novo item a guardar somando uma unidade ao menor contador, mantendo, assim, sempre o mesmo número de contadores. No final deste processo, a soma de cada valor dos contadores corresponderá ao tamanho da lista inicial.

Para a implementação deste algoritmo foi definida uma função, `ssc()`, que recebe uma lista com os caracteres e um valor  $k$  como argumentos.

Inicialmente, é criado um dicionário e, posteriormente, os elementos da lista são percorridos e verifica se já existem no dicionário. Caso existam, o valor do contador aumenta uma unidade. Caso não existam, verifica se o dicionário com o novo contador ultrapassa o número de contadores máximo e, se ultrapassar, adiciona uma unidade ao valor do menor contador e elimina-o, guardando este novo valor associado ao novo elemento. Se não ultrapassar o número

de contadores máximo, adiciona apenas um novo contador com o valor 1. Esta função termina retornando um dicionário de contadores. Este processo pode ser desenvolvido pelo código exibido na figura seguinte.

```

def ssc(chars, k):
    """
    returns a dictionary <char, count>
    """
    counters = {}
    for char in chars:
        char = char.lower()
        if char in counters:
            counters[char] += 1
        else:
            if len(counters) + 1 > k:
                min_counter = min(counters, key=counters.get)
                counters[char] = counters.pop(min_counter) + 1
            else:
                counters[char] = 1

    logger.info('Completed Space Saving Counter (k = {}).'.format(k))

    return counters

```

Fig. 4 – Código Algoritmo Space Saving Count

### IV. RESULTADOS

Foram realizados testes para diferentes ficheiros de textos correspondentes a obras literárias.

Para além deste relatório, foi criado um directório no directório raiz do projecto denominado “results” que contém ficheiros CSV com os resultados finais. As tabelas seguintes demonstram os resultados obtidos.

#### 1. Amor de Perdição de Camilo Castelo Branco

- Contador de caracteres

A primeira coluna corresponde ao caractere contabilizado, a segunda ao contador exacto, a terceira ao space saving count com  $k=10$ , a quarta ao space saving count com  $k=15$ , a quinta ao space saving count com  $k=20$ , a sexta ao space saving count com  $k=25$  e a sétima ao space saving count com  $k=50$ .

	29472	29472	29472	29472	29472	29472
a	1037	-	-	-	-	1037
á	2264	-	-	-	2266	2264
b	1962	19978	10517	4312	1965	1962
c	6522	-	-	6522	6522	6522
ç	1000	-	-	-	-	1000
d	11107	19979	11108	11107	11107	11107
e	28278	28280	28278	28278	28278	28278
é	599	-	-	-	-	599
ê	2	-	-	-	-	2
ë	289	-	-	-	-	289
f	2211	-	10518	4314	2212	2211
g	2432	-	-	-	2434	2432
h	4022	-	-	4318	4024	4022
i	13013	19979	13013	13013	13013	13013
í	21	-	-	-	-	21
j	746	-	-	-	-	746

l	6956	19978	10517	6956	6956	6956
m	10355	-	10631	10355	10355	10355
n	10500	19979	10691	10500	10500	10500
o	22095	22095	22095	22095	22095	22095
ó	205	-	-	-	-	205
ô	132	-	-	-	-	132
ö	120	-	-	-	-	120
p	5208	-	-	5212	5208	5208
q	3173	-	-	4315	3175	3173
r	14828	-	14828	14828	14828	14828
s	16406	19978	16407	16406	16406	16406
t	8681	-	10517	8682	8681	8681
u	10388	-	10587	10388	10388	10388
v	3581	-	-	-	3582	3581
0	15	-	-	-	-	15
1	39	19978	10517	4312	1558	39
2	11	-	-	-	-	11
3	6	-	-	-	-	6
4	8	-	-	-	1557	8
5	10	-	-	4311	1557	10
6	11	-	-	-	1557	11
8	21	-	-	-	-	21
ú	5	-	-	-	-	5
x	410	-	-	-	-	410
y	62	-	-	-	-	62
z	1478	-	-	-	-	1478
7	13	-	-	-	-	13
9	2	-	-	-	-	2

- Erro relativo (%)

A primeira coluna corresponde ao caractere contabilizado, a segunda ao erro associado ao space saving count com k=10, a terceira ao erro associado ao space saving count com k=15, a quarta ao erro associado ao space saving count com k=20, a quinta ao erro associado ao space saving count com k=25 e a sexta ao erro associado ao space saving count com k=50.

a	0	0	0	0	0
m	-	2.67	0	0	0
o	0	0	0	0	0
r	-	0	0	0	0
d	79.88	0.01	0	0	0
e	0.01	0	0	0	0
p	-	-	0.08	0	0
i	53.53	0	0	0	0
ç	-	-	-	-	0
ã	-	-	-	0.09	0
s	21.77	0.01	0	0	0
u	-	1.92	0	0	0
f	-	375.71	95.12	0.05	0
l	187.21	51.19	0	0	0
n	90.28	1.82	0	0	0
c	-	-	0	0	0
t	-	21.15	0.01	0	0
b	918.25	436.04	119.78	0.15	0
q	-	-	35.99	0.06	0
v	-	-	-	0.03	0
j	-	-	-	-	0
á	-	-	-	-	0
g	-	-	-	0.08	0
h	-	-	7.36	0.05	0
é	-	-	-	-	0
ô	-	-	-	-	0
2	-	-	-	-	0
1	51125.64	26866.67	10956.41	3894.87	0
8	-	-	-	-	0
6	-	-	-	14054.55	0
y	-	-	-	-	0
4	-	-	-	19362.5	0
x	-	-	-	-	0

ê	-	-	-	-	0
z	-	-	-	-	0
ô	-	-	-	-	0
0	-	-	-	-	0
3	-	-	-	-	0
5	-	-	43010	15470	0
7	-	-	-	-	0
ó	-	-	-	-	0
9	-	-	-	-	0
è	-	-	-	-	0
ú	-	-	-	-	0
í	-	-	-	-	0

## 2. Os Maias de Eça de Queirós

- Contador de caracteres

A primeira coluna corresponde ao caractere contabilizado, a segunda ao contador exacto, a terceira ao space saving count com k=10, a quarta ao space saving count com k=15, a quinta ao space saving count com k=20, a sexta ao space saving count com k=25 e a sétima ao space saving count com k=50.

char	exact	ssc10	ssc15	ssc20	ssc25	ssc50
e	118387	118389	118387	118387	118387	118387
ç	4116	-	-	-	6240	4116
a	141725	141725	141725	141725	141725	141725
d	50219	90441	50219	50219	50219	50219
q	10009	-	-	18291	10014	10009
u	43099	90441	48359	43100	43099	43099
i	54038	-	54038	54038	54038	54038
r	68080	90447	68080	68080	68080	68080
o	104760	104760	104760	104760	104760	104760
z	4956	-	-	-	6240	4956
s	73719	-	73719	73719	73719	73719
m	46685	90440	48405	46685	46685	46685
p	23458	-	48362	23462	23458	23458
v	17162	90440	48358	18305	17162	17162
n	48337	90439	48478	48337	48337	48337
t	41971	-	-	41971	41971	41971
c	35251	-	-	35251	35251	35251
l	38457	90440	48358	38457	38457	38457
h	16282	-	-	18292	16285	16282
g	12732	-	48357	18295	12733	12732
x	2276	-	-	-	-	2276
1	29	-	-	-	-	29
8	24	-	-	-	-	24
b	11701	-	-	18291	11702	11701
ã	6469	-	-	-	6639	6469
f	10281	-	48357	18297	10282	10281
ó	858	-	-	-	-	858
2	8	-	-	-	-	8
0	16	-	-	-	-	16
3	10	-	-	-	-	10
5	8	-	-	-	-	8
7	8	-	-	-	-	8
j	3046	-	-	-	6240	3046
á	3771	-	-	-	6239	3771

é	2908	-	-	-	-	2908
y	469	-	-	-	-	469
ê	708	-	-	-	-	708
ô	931	-	-	-	-	931
ó	465	-	-	-	-	465
ú	34	-	-	-	-	34
w	101	-	-	-	-	101
ã	51	-	-	-	-	51
ë	45	-	-	-	-	45
í	94	-	-	-	-	94
k	177	-	-	-	-	177
9	2	-	-	-	-	3
o	2	-	-	-	-	-
6	7	-	-	-	-	7
4	4	-	-	-	-	4
ü	1	-	-	-	-	-
ù	1	-	-	-	-	-
ñ	9	-	-	-	-	10
â	5	-	-	-	-	7

- Erro relativo (%)

char ▲ ▼	error-ssc10 ▼	error-ssc15 ▼	error-ssc20 ▼	error-ssc25 ▼	error-ssc50 ▼
a	0	0	0	0	0
á	-	-	-	65.45	0
à	-	-	-	-	0
ã	-	-	-	-	40
ä	-	-	-	2.63	0
b	-	-	56.32	0.01	0
c	-	-	0	0	0
ç	-	-	-	51.6	0
d	80.09	0	0	0	0
e	0	0	0	0	0
é	-	-	-	-	0
è	-	-	-	-	0
ê	-	-	-	-	0
f	-	370.35	77.97	0.01	0
g	-	279.81	43.69	0.01	0
h	-	-	12.35	0.02	0
i	-	0	0	0	0
í	-	-	-	-	0
j	-	-	-	104.86	0
k	-	-	-	-	0
l	135.17	25.75	0	0	0
m	93.72	3.68	0	0	0

n	87.1	0.29	0	0	0
ñ	-	-	-	-	11.11
o	0	0	0	0	0
ó	-	-	-	-	0
ò	-	-	-	-	-
ô	-	-	-	-	0
ö	-	-	-	-	0
p	-	106.16	0.02	0	0
q	-	-	82.75	0.05	0
r	32.85	0	0	0	0
s	-	0	0	0	0
t	-	-	0	0	0
u	109.85	12.2	0	0	0
ú	-	-	-	-	0
ü	-	-	-	-	-
û	-	-	-	-	-
v	426.98	181.77	6.66	0	0
w	-	-	-	-	0
x	-	-	-	-	0
z	-	-	-	25.91	0
0	-	-	-	-	0
1	-	-	-	-	0
2	-	-	-	-	0
3	-	-	-	-	0
4	-	-	-	-	0
5	-	-	-	-	0
6	-	-	-	-	0
7	-	-	-	-	0
8	-	-	-	-	0
y	-	-	-	-	0
9	-	-	-	-	50

### 3. Os Lusíadas de Luís Vaz de Camões

- Contador de Caracteres

a	31064	31064	31064	31064	31064	31064
á	1377	-	-	-	2272	1377
à	158	-	-	5226	2273	158
ã	79	-	-	-	-	79
ä	1620	-	-	-	-	1620
b	2392	-	-	-	2396	2392
c	7071	-	-	7073	7071	7071
ç	863	-	-	-	-	863
d	12308	22760	12308	12308	12308	12308
e	31591	31592	31591	31591	31591	31591
é	825	-	-	-	-	825
è	2	-	-	-	-	2
ê	377	-	-	-	-	377
f	3054	-	-	-	3054	3054
g	3598	-	-	-	3598	3598
h	2584	-	-	-	2598	2584
i	12528	-	12529	12528	12528	12528
í	519	-	-	-	-	519
j	1023	22760	11935	5227	2274	1023
l	6096	-	11935	6096	6096	6096
m	10919	-	11942	10919	10919	10919
n	13452	22760	13453	13452	13452	13452
o	27249	27255	27249	27249	27249	27249
ó	660	-	-	5226	2274	660
ò	1	-	-	-	-	1
ô	52	-	-	-	-	52
ö	115	-	-	-	-	115
p	5557	-	-	5560	5557	5557
q	4113	-	11935	5228	4113	4113
r	16835	22760	16836	16835	16835	16835
s	20650	22784	20650	20650	20650	20650
t	11936	22760	11954	11936	11936	11936
u	10620	-	11938	10620	10620	10620
v	4254	22760	11936	5241	4254	4254
x	368	-	-	5226	2273	368

z	919	-	-	-	-	919
0	148	-	-	-	-	148
1	365	-	-	-	-	365
2	235	-	-	-	-	235
3	234	-	-	-	-	234
4	226	-	-	-	-	226
5	218	-	-	-	-	218
ú	163	-	-	-	-	163
6	210	-	-	-	-	210
7	208	-	-	-	-	208
8	205	-	-	-	-	205
9	193	-	-	-	-	193
ü	14	-	-	-	-	14
y	7	-	-	-	-	7

- Erro relativo (%)

a	0	0	0	0	0
á	-	-	-	65	0
à	-	-	3207.6	1338.61	0
ã	-	-	-	-	0
ä	-	-	-	-	0
b	-	-	-	0.17	0
c	-	-	0.03	0	0
ç	-	-	-	-	0
d	84.92	0	0	0	0

e	0	0	0	0	0
é	-	-	-	-	0
è	-	-	-	-	0
ê	-	-	-	-	0
f	-	-	-	0	0
g	-	-	-	0	0
h	-	-	-	0.54	0
i	-	0.01	0	0	0
í	-	-	-	-	0
j	2124.83	1066.67	410.95	122.29	0
l	-	95.78	0	0	0
m	-	9.37	0	0	0
n	69.19	0.01	0	0	0
o	0.02	0	0	0	0
ó	-	-	691.82	244.55	0
ò	-	-	-	-	0
ô	-	-	-	-	0
õ	-	-	-	-	0
p	-	-	0.05	0	0
q	-	190.18	27.11	0	0
r	35.2	0.01	0	0	0
s	10.33	0	0	0	0
t	90.68	0.15	0	0	0
u	-	12.41	0	0	0
v	435.03	180.58	23.2	0	0
x	-	-	1320.11	517.66	0
z	-	-	-	-	0
0	-	-	-	-	0
1	-	-	-	-	0
2	-	-	-	-	0
3	-	-	-	-	0
4	-	-	-	-	0
5	-	-	-	-	0
ú	-	-	-	-	0

6	-	-	-	-	0
7	-	-	-	-	0
8	-	-	-	-	0
9	-	-	-	-	0
ü	-	-	-	-	0
y	-	-	-	-	0

#### 4. Hamlet de William Shakespeare

- Contador de caracteres

A primeira coluna corresponde ao caractere contabilizado, a segunda ao contador exacto, a terceira ao space saving count com  $k=10$ , a quarta ao space saving count com  $k=15$ , a quinta ao space saving count com  $k=20$ , a sexta ao space saving count com  $k=25$  e a sétima ao space saving count com  $k=50$ .

t	10853	-	10853	10853	10853	10853
h	8207	11773	8210	8207	8207	8207
e	16335	16335	16335	16335	16335	16335
r	7146	11773	7146	7146	7146	7146
a	9024	11773	9024	9024	9024	9024
g	2238	-	-	2529	2238	2238
d	4583	11772	6192	4583	4583	4583
i	7938	-	7941	7938	7938	7938
o	10441	11772	10441	10441	10441	10441
f	2487	11772	6195	2539	2487	2487
m	3967	11772	-	3967	3967	3967

l	5450	-	6193	5450	5450	5450
c	2364	-	6192	2576	2364	2364
u	4598	-	-	4598	4598	4598
s	7566	-	7569	7566	7566	7566
p	1834	-	6192	2531	1834	1834
n	7611	11773	7612	7611	7611	7611
b	1704	-	-	-	1704	1704
w	2890	-	-	2891	2890	2890
y	2948	-	-	2974	2948	2948
v	498	-	-	-	498	498
k	1220	11773	6193	2529	1220	1220
q	195	-	-	-	195	195
x	137	-	-	-	137	137
z	49	-	-	-	54	49
j	1	-	-	-	-	1
1	4	-	-	-	-	4

- Erro relativo (%)

t	-	0	0	0	0
h	43.45	0.04	0	0	0
e	0	0	0	0	0
r	64.75	0	0	0	0
a	30.46	0	0	0	0
g	-	-	13	0	0
d	156.86	35.11	0	0	0
i	-	0.04	0	0	0
o	12.75	0	0	0	0
f	373.34	149.1	2.09	0	0
m	196.75	-	0	0	0
l	-	13.63	0	0	0
c	-	161.93	8.97	0	0
u	-	-	0	0	0
s	-	0.04	0	0	0
p	-	237.62	38	0	0
n	54.68	0.01	0	0	0
b	-	-	-	0	0
w	-	-	0.04	0	0
y	-	-	0.88	0	0
v	-	-	-	0	0
k	865	407.62	107.3	0	0
q	-	-	-	0	0
x	-	-	-	0	0
z	-	-	-	10.2	0
j	-	-	-	-	0
1	-	-	-	-	0

#### 5. A Bíblia

- Contador de caracteres

char	exact2	ssc10	ssc15	ssc20	ssc25	ssc50
1	27087	-	-	-	34126	27087
a	275701	325593	275701	275701	275701	275701
b	48853	325594	171569	82318	48853	48853
c	55056	-	-	82469	55056	55056
d	158086	-	-	158086	158086	158086
e	412160	412160	412160	412160	412160	412160
f	83534	-	-	83817	83534	83534
g	55279	-	-	82316	55279	55279
h	282657	-	282657	282657	282657	282657
i	193926	-	193934	193926	193926	193926
j	8880	-	-	-	33937	8880
k	22281	325593	171567	82316	33944	22281
l	129919	-	171567	129921	129919	129919
m	79929	-	171566	82322	79929	79929
n	225026	325593	225026	225026	225026	225026
o	243143	325601	243143	243143	243143	243143
p	43248	-	-	82332	43374	43248
q	964	-	-	-	-	964
r	170301	-	172846	170301	170301	170301
s	189997	325594	190040	189997	189997	189997
t	317696	325693	317696	317696	317696	317696
v	30362	-	-	-	34523	30362
z	18976	-	-	-	-	18976
u	83462	325593	171568	83784	83462	83462
w	65478	325593	171567	82319	65478	65478

0	5318	-	-	-	-	5318
3	12259	-	-	-	-	12259
4	9150	-	-	-	-	9150
5	7053	-	-	-	33931	7053
y	58568	-	-	-	58569	58568
6	6472	-	-	-	-	6472
7	5917	-	-	-	-	5917
8	5778	-	-	-	-	5778
9	5641	-	-	-	-	5641
x	1478	-	-	-	-	1478
z	2972	-	-	-	-	2972

- Erro Relativo (%)

char	error-ssc10	error-ssc15	error-ssc20	error-ssc25	error-ssc50
1	-	-	-	25.99	0
i	-	0	0	0	0
n	44.69	0	0	0	0
t	2.52	0	0	0	0
h	-	0	0	0	0
e	0	0	0	0	0
b	566.48	251.19	68.5	0	0
g	-	-	48.91	0	0
o	33.91	0	0	0	0
d	-	-	0	0	0
c	-	-	49.79	0	0
r	-	1.49	0	0	0
a	18.1	0	0	0	0
v	-	-	-	13.71	0
2	-	-	-	-	0
w	397.26	162.02	25.72	0	0
s	71.37	0.02	0	0	0
u	290.11	105.56	0.39	0	0
f	-	-	0.34	0	0

m	-	114.65	2.99	0	0
k	1361.3	670.02	269.45	52.35	0
p	-	-	90.37	0.29	0
3	-	-	-	-	0
l	-	32.06	0	0	0
4	-	-	-	-	0
5	-	-	-	381.09	0
y	-	-	-	0	0
6	-	-	-	-	0
7	-	-	-	-	0
8	-	-	-	-	0
9	-	-	-	-	0
0	-	-	-	-	0
x	-	-	-	-	0
j	-	-	-	282.17	0
z	-	-	-	-	0

## 6. Romeu e Julieta de William Shakespeare

- Contador de caracteres

char	exact2	ssc10	ssc15	ssc20	ssc25	ssc50
t	11051	11789	11051	11051	11051	11051
h	7405	-	7439	7405	7405	7405
e	14279	14279	14279	14279	14279	14279
p	1984	-	-	2870	1984	1984
r	7686	-	7686	7686	7686	7686
o	9938	11770	9938	9938	9938	9938
j	375	-	-	-	383	375
c	2751	-	-	2982	2751	2751
g	2260	-	-	2866	2260	2260
u	4089	11762	6316	4089	4089	4089
n	7493	11762	7494	7493	7493	7493
b	2022	11763	6316	2868	2022	2022
k	987	11762	6314	2865	987	987
f	2477	-	-	-	2478	2477
m	3812	-	6314	3818	3812	3812
a	9153	11762	9154	9153	9153	9153
d	4440	-	-	4444	4440	4440
l	5285	-	6315	5285	5285	5285
i	7854	-	7855	7854	7854	7854
y	2938	-	-	2973	2938	2938

w	2856	11762	6315	2872	2856	2856
s	7382	11763	7388	7383	7382	7382
v	1169	-	-	-	1172	1169
z	25	-	-	-	-	25
5	15	-	-	-	238	15
0	19	-	-	-	-	19
1	94	-	-	-	-	94
9	16	-	-	-	-	16
7	7	-	-	-	-	7
x	160	-	-	-	-	160
3	16	-	-	-	-	16
z	35	-	-	-	-	35
q	76	-	-	-	238	76
8	9	-	-	-	-	9
4	9	-	-	-	-	9
6	7	-	-	-	-	7

- Erro Relativo (%)

char	error-ssc10	error-ssc15	error-ssc20	error-ssc25	error-ssc50
t	6.68	0	0	0	0
h	-	0.46	0	0	0
e	0	0	0	0	0
p	-	-	44.66	0	0
r	-	0	0	0	0
o	18.43	0	0	0	0
j	-	-	-	2.13	0
c	-	-	8.4	0	0
g	-	-	26.81	0	0
u	187.65	54.46	0	0	0
n	56.97	0.01	0	0	0
b	481.75	212.36	41.84	0	0
k	1091.69	539.72	190.27	0	0
f	-	-	-	0.04	0
m	-	65.64	0.16	0	0
a	28.5	0.01	0	0	0
d	-	-	0.09	0	0
l	-	19.49	0	0	0
i	-	0.01	0	0	0
y	-	-	1.19	0	0
w	311.84	121.11	0.56	0	0

s	59.35	0.08	0.01	0	0
v	-	-	-	0.26	0
2	-	-	-	-	0
5	-	-	-	1486.67	0
0	-	-	-	-	0
1	-	-	-	-	0
9	-	-	-	-	0
7	-	-	-	-	0
x	-	-	-	-	0
3	-	-	-	-	0
z	-	-	-	-	0
q	-	-	-	213.16	0
8	-	-	-	-	0
4	-	-	-	-	0
6	-	-	-	-	0

#### V. MÁXIMO, MÍNIMO E MÉDIA DO ERRO RELATIVO (%)

Os dados seguintes estão organizados pela seguinte ordem:

##### 1. Nome do ficheiro:

- Space Saving Count k=X  
[MAX,MIN,AVG]

##### 1. Amor de Perdição

- Space Saving Count k=10  
[51125.641 , 0 , 5247.66]
- Space Saving Count k=15  
[26866 , 0 , 1850.48]
- Space Saving Count k=20  
[43010 , 0 , 2711.24]
- Space Saving Count k=25  
[19362 , 0 , 2111.3]
- Space Saving Count k= 50  
[0, 0 , 0 ]

##### 2. Os Maias

- Space Saving Count k=10  
[426.978 , 0 , 0 ]
- Space Saving Count k=15  
[370.373 , 0 , 65.33]
- Space Saving Count k=20  
[82.746 , 0 , 13.99]
- Space Saving Count k=25  
[104.859 , 0 , 10.02]
- Space Saving Count k= 50  
[50 , 0 , 2.02]

##### 3. Os Lusíadas

- Space Saving Count k=10  
[2124 , 0 , 285.02 ]
- Space Saving Count k=15  
[1066.667 , 0 , 103.68]
- Space Saving Count k=20  
[3207.595 , 0 , 284.04]
- Space Saving Count k=25  
[1388.608 , 0 , 91.55]
- Space Saving Count k= 50  
[0, 0 , 0 ]

##### 4. Hamlet

- Space Saving Count k=10  
[865 , 0 , 179.8]
- Space Saving Count k=15  
[407.623 , 0 , 67.01]
- Space Saving Count k=20  
[107.295 , 0 , 8.51]
- Space Saving Count k=25  
[10.204 , 0 , 0.41]
- Space Saving Count k= 50  
[0, 0 , 0]

##### 5. A Bíblia

- Space Saving Count k=10  
[1361.303 , 0 , 278.57]
- Space Saving Count k=15  
[670.015 , 0 , 89.13]
- Space Saving Count k=20  
[269.445 , 0 , 27.82]
- Space Saving Count k=25  
[381.086 , 0 , 30.22]
- Space Saving Count k= 50  
[0, 0 , 0]

##### 6. Romeu e Julieta

- Space Saving Count k=10  
[1091.692 , 0 , 224.29]
- Space Saving Count k=15  
[539.716 , 0 , 67.56]
- Space Saving Count k=20  
[190.274 , 0 , 15.7]
- Space Saving Count k=25  
[1486.667 , 0 , 68.09]
- Space Saving Count k= 50  
[0, 0 , 0 ]

#### VI. CONCLUSÃO

Analisando os dados relativos ao cálculo do erro é possível afirmar que:

- para k=10 e k=15 os contadores têm um erro associado bastante elevado;
- para k=50 o erro é quase nulo, uma vez que o k é muito maior que o número de letras possíveis. Apesar do erro ser quase nulo, o esforço computacional é consideravelmente maior em relação a contadores com um k menor.

Assim, é possível concluir que o melhor valor de k seria 20 ou 25 (dependendo do idioma – português ou inglês). De salientar que todos os contadores (independentemente do valor de k) acertaram pelo menos uma vez num caractere (erro mínimo igual a 0).

Em suma, é possível concluir que quanto maior for o valor de k (número de contadores) menor será o

erro observado. Quanto menor for a diferença entre  $k$  e o número de letras distintas menor será o erro relativo, uma vez que as letras serão melhor distribuídas pelos respectivos contadores. No entanto, quanto maior for o valor de  $k$  maior será o esforço computacional.

#### REFERÊNCIAS

[1] – Project Gutenberg - <https://www.gutenberg.org/>

[2] AA 11 – Data Stream Algorithms I  
[https://elearning.ua.pt/pluginfile.php/2940932/mod\\_resource/content/0/AA\\_11\\_Data\\_Stream\\_Algorithms\\_I.pdf](https://elearning.ua.pt/pluginfile.php/2940932/mod_resource/content/0/AA_11_Data_Stream_Algorithms_I.pdf)