

CME4408 INTRODUCTION TO NATURAL LANGUAGE PROCESSING – ASSIGNMENT 1		09/12/2021
Name	:	ŞEFİKA ÖZLEM PUL
Number	:	2017510067

DEFINITION OF ALGORITHM

C# Code Algorithm

Main Function

First of all, the user was asked to enter the text file for which n-gram was created and to enter the n value with "Console.ReadLine()" in C#. For the file path to be used in the file reading process of the input file entered by the user, the file path was created by string concatenation with the "Novel-Samples" folder in which it is located. Since the Novel-Samples folder is the same as the directory where the program is running, there was no need to enter a long file path exactly. By using StreamReader, the "ReadLine()" function was used to read the file line by line in the while loop. Each line of the file read is added to the text variable of type string. Then, the white spaces in the resulting text variable were removed using "Trim()" and converted to lowercase with "ToLower()". The punctuation marks in the text were replaced with a one-character space (*Regex.Replace(text, @"[\w\d\s]", " ")*). Extra spaces after removing punctuation marks were removed and a single-character space was left. Finally, the "n_gram" method created is called to do all the rest.

n_gram Method

```
public static void n_gram(string file_text, int n)
```

It is called after the file reading operations are finished in the main function and takes the string type text variable that holds the read file and the value n as parameters. In the file_text string variable, the entire file is split according to the space character to separate its words and thrown into an array of string type. With the for loop, the loop continues as much as n minus the length of the words array from the zeroth index of the array with the words. The string value returned from the "append_words" function is assigned to the "result" variable of the string type defined as static, in order to combine n words in order to form n grams in the loop. Control cases were added to the created dictionary named ngram_dic, which takes string as key and int as value. If the result value (kept as a key) is checked with "ContainsKey" in the ngram_dic dictionary and it is true, the value of this result is incremented by one. But if false is returned, that is, if the result value is not in the dictionary, result as key and 1 as value is added to the ngram_dic dictionary (*ngram_dic.Add(result.Trim(), 1);*).

After obtaining the ngram_dic dictionary, which contains N-gram string parts and their count (value) values, the most repetitive 50 items were printed. In the print process, a variable named "datas" holding our data was assigned in the foreach loop and with the "OrderByDescending" function, the items in ngram_dic were ordered according to the value from the most to the least (*foreach (KeyValuePair<string, int> data in ngram_dic.OrderByDescending(datas) =>*

datas.Value).))) . In order to print only the first 50 of the sorted items, a count variable of type integer was kept and in the foreach loop, the count value was increased by one by printing the key and value until the count value was less than 50 and equal.

append_words Function

```
public static string append_words(string[] words, int start_index, int n)
```

In the *n_gram* method, after the words are split, it is a function that we call while navigating the words array with a for loop, one more than *n* minus the length of the words array, and returns a string value that takes the words array, *i* value, and the number *n* as a parameter. The *start_index* of type integer in the function is the *i* value in the for loop coming from the *n_gram* method. Looping the words in the array and combining them in *n*-gram format was done in a for loop. In the for loop, our initial value was determined as the variable “*int i = start_index*” and continued until the value “*start_index+n*”. If we need to give an example to explain this better, if the *start_index* value is 2 and the *n* value is 3, the words found in the 2nd index of the array of words and the words between the 5th index, that is, 3 words, are included in the process. In the for loop, the word in the *i*th index of the words array is added to the string variable *n_words* (*n_words = n_words + " " + words[i];*). The variable *n_words* is returned after exiting the loop.

Python Code Algorithm

Main Function

The user is prompted for the file name and *n* value. While the file is being read, the blank lines are replaced with a single blank character. In this way, blank lines are deleted. The punctuation marks (*re.sub(r'[^\w\s]',' ', line)*) in the file have been removed. After the lowercase and split operation, the “*ngram_create*” function was called to convert it to *n* gram format and the result returned from this function was assigned to the “*result*” string variable defined as static. Then, each index of the result variable is added to the dictionary named *ngram_dict*, which is defined as static, giving its value 1 if it is not in the dictionary, and its value is increased by one if it exists. To sort the most recurring values, “*sorted*” was used and a variable named *sorted_d* was assigned in the dictionary structure (*sorted_d = dict(sorted(ngram_dict.items(), key=lambda item: item[1],reverse=True))*). By checking the count variable of type Integer, 50 items in the sequential dictionary were printed on the screen.

ngram_create Fonksiyonu

```
def ngram_create(words, n):
```

This function, which takes string and integer type parameters, returns the created *n*-gram array at the end. Starting from zero with the for loop, the string type words that it takes as a parameter continue until the length of the variable is *n* minus one more. In the loop, a new string part is assigned to the *ngram* variable of string type, which is joined by “*join*” the words elements between the *i* index of the words and the *i+n* index. *ngram* variable in string type *n* gram format

string part is added to ngrams array. When the loop ends, it is assigned to the result variable in the array main, which holds the string parts in n gram format.

SCREENSHOTS OF PROGRAM

C# Code Results Screenshots

```
Enter a file name(xxx.txt) : BİLİM İŞ BAŞINDA.txt
Enter n value : 1
1: bir - 454
2: ve - 203
3: bu - 154
4: için - 90
5: de - 75
6: çok - 73
7: ile - 71
8: daha - 66
9: da - 58
10: olarak - 52
11: ancak - 50
12: elektrik - 47
13: veya - 43
14: su - 43
15: kadar - 41
16: büyük - 37
17: biçimde - 30
18: gibi - 30
19: olan - 29
20: iki - 29
21: enerji - 29
22: doğru - 27
23: pek - 24
```

```
24: yaklaşık - 24
25: ilk - 24
26: zaman - 22
27: oysa - 22
28: diğer - 22
29: iyi - 22
30: olduğu - 20
31: üzerinde - 20
32: sonra - 20
33: deniz - 20
34: önce - 19
35: cam - 19
36: her - 18
37: ne - 18
38: en - 18
39: suyun - 17
40: basınç - 17
41: biri - 17
42: bilim - 16
43: şey - 16
44: o - 16
45: geri - 16
46: olduğunu - 16
47: küçük - 16
48: sol - 16
49: sağ - 16
50: yük - 16
Elapsed Time is 41 ms
```

```
Enter a file name(xxx.txt) : BİLİM İŞ BAŞINDA.txt
Enter n value : 2
1: pek çok - 19
2: bir biçimde - 19
3: ile ilgili - 11
4: deniz suyu - 10
5: bu nedenle - 9
6: uygun bir - 9
7: bir elektrik - 9
8: herhangi bir - 9
9: yine de - 9
10: iyi bir - 9
11: yıl önce - 8
12: bir ses - 8
13: elektrik yükü - 8
14: söz konusu - 7
15: küçük bir - 7
16: büyük bir - 7
17: ve bu - 6
18: bir kısmı - 6
19: için bir - 6
20: bir cam - 6
21: daha fazla - 6
22: ve bir - 6
23: bir şey - 6
```

```
24: daha iyi - 6
25: deniz suyunu - 6
26: ancak bu - 6
27: sağlam bir - 5
28: sonucuna vardı - 5
29: hem de - 5
30: tam olarak - 5
31: ince bir - 5
32: elde edilen - 5
33: ultrasonik bir - 5
34: kimi zaman - 5
35: ne yazık - 5
36: yazık ki - 5
37: bir enerji - 5
38: diğer bir - 5
39: artı yüklü - 5
40: bir deyişle - 5
41: ne kadar - 5
42: o kadar - 5
43: yaygın olarak - 5
44: ağırlık merkezi - 5
45: sağa doğru - 5
46: olmadan da - 4
47: bilim adamı - 4
48: daha önce - 4
49: her iki - 4
50: dahil olmak - 4
Elapsed Time is 44 ms
```

```
Enter a file name(xxx.txt) : BİLİM İŞ BAŞINDA.txt
Enter n value : 3
1: ne yazık ki - 5
2: dahil olmak üzere - 4
3: başka herhangi bir - 4
4: diğer bir deyişle - 4
5: söz konusu olduğunda - 4
6: da dahil olmak - 3
7: bir ses dalgası - 3
8: ultrasonik bir ses - 3
9: büyük miktarda enerji - 3
10: bir kabın içine - 3
11: için yarı geçirgen - 3
12: işe yarar bir - 3
13: yaygın olarak kullanılan - 3
14: bir elektrik yükü - 3
15: termodinamiğin ikinci yasası - 3
16: buzun erime noktası - 3
17: pek çok şey - 2
18: the glasgow heraldda - 2
19: bir kısmı da - 2
20: için de çok - 2
21: geçmişte tam olarak - 2
22: ince bir su - 2
23: bir su tabakası - 2
```

```
24: yaklaşık 40 yıl - 2
25: 40 yıl önce - 2
26: kar yağışı nedeniyle - 2
27: bir dağ kulübesinde - 2
28: ders kitaplarının yanlışı - 2
29: olması gerektiği sonucuna - 2
30: gerektiği sonucuna vardı - 2
31: pek çok ilginç - 2
32: bir elektrik sinyali - 2
33: bir mikrofon olarak - 2
34: mikrofon olarak da - 2
35: olarak da kullanılabilir - 2
36: bir ses dalgasının - 2
37: ultrasonik bir sinyal - 2
38: geri yansır ve - 2
39: da hâlâ tam - 2
40: hâlâ tam olarak - 2
41: deniz suyu içmek - 2
42: tuzu sudan ayırmak - 2
43: çok pahalı olduğu - 2
44: çok aşamalı ani - 2
45: aşamalı ani etkili - 2
46: ani etkili damıtma - 2
47: alçak basınç altında - 2
48: basınç altında tutulan - 2
49: bir kaba aktarılır - 2
50: suyun bir kısmı - 2
Elapsed Time is 45 ms
```

```
Enter a file name(xxx.txt) : BOZKIRDA.txt
Enter n value : 2
1: diye sordu - 45
2: bir sesle - 44
3: bir şey - 33
4: diye bağırdı - 28
5: bir tavırla - 22
6: karşılık verdi - 21
7: diye karşılık - 19
8: ben de - 15
9: ağır ağır - 15
10: hem de - 14
11: bir şeyler - 13
12: sert bir - 12
13: bir süre - 12
14: ya da - 11
15: o zaman - 11
16: ne var - 11
17: belki de - 10
18: öyle bir - 10
19: dedi yakov - 10
20: anladın mı - 10
21: sen de - 10
22: öyle mi - 10
23: kimi zaman - 10
```

```
24: ne olacak - 9
25: bak sen - 9
26: yine de - 9
27: ne diye - 9
28: ne kadar - 8
29: her şey - 8
30: ne de - 8
31: daha çok - 8
32: değil mi - 8
33: her zaman - 8
34: güzel bir - 7
35: boğuk bir - 7
36: ben mi - 7
37: sözlerini sürdürdü - 7
38: arada bir - 6
39: tuhaf bir - 6
40: o da - 6
41: bu kadar - 6
42: söze başladı - 6
43: o kadar - 6
44: bir kahkaha - 6
45: bir yandan - 6
46: ne olmuş - 6
47: yakov babasına - 6
48: sonra da - 6
49: içini çekti - 6
50: sordu yakov - 6
Elapsed Time is 73 ms
```

```
Enter a file name(xxx.txt) : DEĞİŞİM.txt
Enter n value : 1
1: bir - 541
2: ve - 256
3: bu - 164
4: gregor - 159
5: ama - 146
6: de - 123
7: da - 103
8: gibi - 95
9: gregorun - 95
10: için - 93
11: kızkardeşi - 93
12: daha - 82
13: diye - 77
14: ne - 74
15: babası - 66
16: çünkü - 61
17: üzerine - 57
18: ancak - 56
19: sonra - 55
20: her - 54
21: hiç - 53
22: hemen - 53
23: pek - 53
```

```
24: o - 51
25: kadar - 50
26: müdür - 49
27: şey - 48
28: böyle - 46
29: bütün - 46
30: şimdi - 45
31: yere - 44
32: yine - 44
33: annesi - 43
34: çok - 43
35: kızkardeşinin - 41
36: ki - 40
37: biraz - 39
38: üzerinde - 37
39: kendini - 34
40: en - 34
41: bile - 34
42: doğru - 33
43: belki - 33
44: anne - 33
45: onu - 33
46: artık - 33
47: başka - 31
48: babasının - 31
49: samsa - 30
50: kendisini - 30
Elapsed Time is 51 ms
```

```
Enter a file name(xxx.txt) : DENEMELER.txt
Enter n value : 3
1: kitap 3 bölüm - 65
2: kitap 1 bölüm - 56
3: kitap 2 bölüm - 55
4: 2 bölüm 12 - 17
5: 3 bölüm 9 - 11
6: başka bir şey - 10
7: 1 bölüm 26 - 10
8: o kadar ki - 10
9: 3 bölüm 13 - 9
10: 3 bölüm 12 - 9
11: ben kendi hesabıma - 8
12: 3 bölüm 2 - 7
13: 3 bölüm 5 - 7
14: pek o kadar - 6
15: ruh ve beden - 6
16: öyle sanıyorum ki - 6
17: 1 bölüm 14 - 6
18: bir şey değildir - 5
19: bir o yana - 5
20: o yana bir - 5
21: yana bir bu - 5
22: bir bu yana - 5
23: bana öyle geliyor - 5
```

```
24: öyle geliyor ki - 5
25: ve daha başka - 5
26: hiçbir şey yoktur - 4
27: çok daha fazla - 4
28: 2 bölüm 16 - 4
29: bu kadar büyük - 4
30: kadar büyük bir - 4
31: 1 bölüm 20 - 4
32: daha büyük bir - 4
33: ve demiş ki - 4
34: ne kadar az - 4
35: o kadar az - 4
36: 3 bölüm 10 - 4
37: 3 bölüm 1 - 4
38: etienne de la - 3
39: bir tek insan - 3
40: da çok kez - 3
41: ay sonra ölüyor - 3
42: ne büyük bir - 3
43: hem de en - 3
44: 1 bölüm 25 - 3
45: diye cevap vermiş - 3
46: insan ne kadar - 3
47: lucretius her şey - 3
48: de onun için - 3
49: 1 bölüm 39 - 3
50: 2 bölüm 20 - 3
Elapsed Time is 901 ms
```

```
Enter a file name(xxx.txt) : DENEMELER.txt
Enter n value : 3
1: kitap 3 bölüm - 65
2: kitap 1 bölüm - 56
3: kitap 2 bölüm - 55
4: 2 bölüm 12 - 17
5: 3 bölüm 9 - 11
6: başka bir şey - 10
7: 1 bölüm 26 - 10
8: o kadar ki - 10
9: 3 bölüm 13 - 9
10: 3 bölüm 12 - 9
11: ben kendi hesabıma - 8
12: 3 bölüm 2 - 7
13: 3 bölüm 5 - 7
14: pek o kadar - 6
15: ruh ve beden - 6
16: öyle sanıyorum ki - 6
17: 1 bölüm 14 - 6
18: bir şey değildir - 5
19: bir o yana - 5
20: o yana bir - 5
21: yana bir bu - 5
22: bir bu yana - 5
23: bana öyle geliyor - 5
```

```
24: öyle geliyor ki - 5
25: ve daha başka - 5
26: hiçbir şey yoktur - 4
27: çok daha fazla - 4
28: 2 bölüm 16 - 4
29: bu kadar büyük - 4
30: kadar büyük bir - 4
31: 1 bölüm 20 - 4
32: daha büyük bir - 4
33: ve demiş ki - 4
34: ne kadar az - 4
35: o kadar az - 4
36: 3 bölüm 10 - 4
37: 3 bölüm 1 - 4
38: etienne de la - 3
39: bir tek insan - 3
40: da çok kez - 3
41: ay sonra ölüyor - 3
42: ne büyük bir - 3
43: hem de en - 3
44: 1 bölüm 25 - 3
45: diye cevap vermiş - 3
46: insan ne kadar - 3
47: lucretius her şey - 3
48: de onun için - 3
49: 1 bölüm 39 - 3
50: 2 bölüm 20 - 3
Elapsed Time is 1400 ms
```

```
Enter a file name(xxx.txt) : grimms-fairy-tales_P1.txt
Enter n value : 2
1: in the - 60
2: and the - 56
3: to the - 47
4: of the - 35
5: said the - 32
6: into the - 27
7: on the - 24
8: the fox - 22
9: then the - 21
10: he was - 21
11: the king - 20
12: but the - 20
13: at last - 20
14: i will - 20
15: as he - 20
16: and said - 20
17: so he - 19
18: came to - 18
19: the golden - 17
20: to be - 17
21: then he - 17
22: all the - 16
23: he had - 16
```

```
24: by the - 15
25: for the - 15
26: the princess - 15
27: the ass - 15
28: and when - 14
29: he went - 14
30: and they - 14
31: him and - 14
32: the horse - 14
33: the old - 14
34: with a - 14
35: began to - 13
36: the bird - 13
37: it and - 13
38: he could - 13
39: a great - 13
40: the wolf - 13
41: and in - 12
42: it was - 12
43: he came - 12
44: could not - 12
45: king and - 12
46: she was - 12
47: as the - 11
48: when he - 11
49: the castle - 11
50: out of - 11
Elapsed Time is 36 ms
```

Python Code Results Screenshots

```
===== RESTART: C:\Users\Lenovo\Desktop\2017510067.py
Enter file name (ppp.txt) : BİLİM İŞ BAŞINDA.txt
Enter n value : 2
1 : pek çok - 19
2 : bir biçimde - 19
3 : ile ilgili - 11
4 : deniz suyu - 10
5 : bu nedenle - 9
6 : uygun bir - 9
7 : bir elektrik - 9
8 : herhangi bir - 9
9 : yine de - 9
10 : iyi bir - 9
11 : yıl önce - 8
12 : bir ses - 8
13 : elektrik yükü - 8
14 : söz konusu - 7
15 : küçük bir - 7
16 : büyük bir - 7
17 : ve bu - 6
18 : bir kısmı - 6
19 : için bir - 6
20 : bir cam - 6
21 : daha fazla - 6
22 : ve bir - 6
23 : bir şey - 6
```

```
24 : daha iyi - 6
25 : deniz suyunu - 6
26 : ancak bu - 6
27 : sağlam bir - 5
28 : sonucuna vardı - 5
29 : hem de - 5
30 : tam olarak - 5
31 : ince bir - 5
32 : elde edilen - 5
33 : ultrasonik bir - 5
34 : kimi zaman - 5
35 : ne yazık - 5
36 : yazık ki - 5
37 : bir enerji - 5
38 : diğer bir - 5
39 : artı yüklü - 5
40 : bir deyişle - 5
41 : ne kadar - 5
42 : o kadar - 5
43 : yaygın olarak - 5
44 : ağırlık merkezi - 5
45 : sağa doğru - 5
46 : olmadan da - 4
47 : bilim adamı - 4
48 : daha önce - 4
49 : her iki - 4
50 : dahil olmak - 4
Elapsed time is 627.5076866149902 ms
```

 C:\Users\Lenovo\Desktop\dist\2017510067.exe


```
Enter file name (ppp.txt) : BİLİM İŞ BAŞINDA.txt
Enter n value : 2
1 : pek çok - 19
2 : bir biçimde - 19
3 : ile ilgili - 11
4 : deniz suyu - 10
5 : bu nedenle - 9
6 : uygun bir - 9
7 : bir elektrik - 9
8 : herhangi bir - 9
9 : yine de - 9
10 : iyi bir - 9
11 : yıl önce - 8
12 : bir ses - 8
13 : elektrik yükü - 8
14 : söz konusu - 7
15 : küçük bir - 7
16 : büyük bir - 7
17 : ve bu - 6
18 : bir kısmı - 6
19 : için bir - 6
20 : bir cam - 6
21 : daha fazla - 6
22 : ve bir - 6
23 : bir şey - 6
```

```
24 : daha iyi - 6
25 : deniz suyunu - 6
26 : ancak bu - 6
27 : sağlam bir - 5
28 : sonucuna vardı - 5
29 : hem de - 5
30 : tam olarak - 5
31 : ince bir - 5
32 : elde edilen - 5
33 : ultrasonik bir - 5
34 : kimi zaman - 5
35 : ne yazık - 5
36 : yazık ki - 5
37 : bir enerji - 5
38 : diğer bir - 5
39 : artı yüklü - 5
40 : bir deyişle - 5
41 : ne kadar - 5
42 : o kadar - 5
43 : yaygın olarak - 5
44 : ağırlık merkezi - 5
45 : sağa doğru - 5
46 : olmadan da - 4
47 : bilim adamı - 4
48 : daha önce - 4
49 : her iki - 4
50 : dahil olmak - 4
Elapsed time is 28.90920639038086 ms
```



```
Enter file name (ppp.txt) : DEĞİŞİM.txt
Enter n value : 1
1 : bir - 541
2 : ve - 256
3 : bu - 164
4 : gregor - 159
5 : ama - 146
6 : de - 123
7 : da - 103
8 : gibi - 95
9 : gregorun - 95
10 : için - 93
11 : kızkardeşi - 93
12 : daha - 82
13 : diye - 77
14 : ne - 74
15 : babası - 66
16 : çünkü - 61
17 : üzerine - 57
18 : ancak - 56
19 : sonra - 55
20 : her - 54
21 : hiç - 53
22 : hemen - 53
23 : pek - 53
```

```
24 : o - 51
25 : kadar - 50
26 : müdür - 49
27 : şey - 48
28 : böyle - 46
29 : bütün - 46
30 : şimdi - 45
31 : yere - 44
32 : yine - 44
33 : annesi - 43
34 : çok - 43
35 : kızkardeşinin - 41
36 : ki - 40
37 : biraz - 39
38 : üzerinde - 37
39 : kendini - 34
40 : en - 34
41 : bile - 34
42 : doğru - 33
43 : belki - 33
44 : anne - 33
45 : onu - 33
46 : artık - 33
47 : başka - 31
48 : babasının - 31
49 : samsa - 30
50 : kendisini - 30
Elapsed time is 1038.219690322876 ms
```

 C:\Users\Lenovo\Desktop\dist\2017510067.exe

```
Enter file name (ppp.txt) : DEĞİŞİM.txt
Enter n value : 1
1 : bir - 541
2 : ve - 256
3 : bu - 164
4 : gregor - 159
5 : ama - 146
6 : de - 123
7 : da - 103
8 : gibi - 95
9 : gregorun - 95
10 : için - 93
11 : kızkardeşi - 93
12 : daha - 82
13 : diye - 77
14 : ne - 74
15 : babası - 66
16 : çünkü - 61
17 : üzerine - 57
18 : ancak - 56
19 : sonra - 55
20 : her - 54
21 : hiç - 53
22 : hemen - 53
23 : pek - 53
```

```
24 : o - 51
25 : kadar - 50
26 : müdür - 49
27 : şey - 48
28 : böyle - 46
29 : bütün - 46
30 : şimdi - 45
31 : yere - 44
32 : yine - 44
33 : annesi - 43
34 : çok - 43
35 : kızkardeşinin - 41
36 : ki - 40
37 : biraz - 39
38 : üzerinde - 37
39 : kendini - 34
40 : en - 34
41 : bile - 34
42 : doğru - 33
43 : belki - 33
44 : anne - 33
45 : onu - 33
46 : artık - 33
47 : başka - 31
48 : babasının - 31
49 : samsa - 30
50 : kendisini - 30
Elapsed time is 27.143239974975586 ms
```



```
Enter file name (ppp.txt) : DENEMELER.txt
Enter n value : 2
1 : o kadar - 75
2 : kitap 3 - 65
3 : 3 bölüm - 65
4 : ne kadar - 56
5 : kitap 1 - 56
6 : 1 bölüm - 56
7 : kitap 2 - 55
8 : 2 bölüm - 55
9 : bir şey - 51
10 : ya da - 49
11 : her şeyi - 42
12 : başka bir - 40
13 : bu kadar - 37
14 : büyük bir - 37
15 : hiç de - 34
16 : daha fazla - 32
17 : her şey - 32
18 : daha çok - 31
19 : hem de - 28
20 : her zaman - 27
21 : bölüm 12 - 27
22 : ki bu - 27
23 : böyle bir - 26
```

```
24 : çok daha - 25
25 : onun için - 25
26 : der ki - 24
27 : bir tek - 23
28 : daha az - 23
29 : için de - 23
30 : en çok - 22
31 : her gün - 22
32 : olduğu için - 22
33 : o zaman - 21
34 : ne de - 21
35 : daha iyi - 20
36 : ve daha - 20
37 : en büyük - 20
38 : her şeyin - 20
39 : var ki - 19
40 : o da - 19
41 : hiçbir şey - 18
42 : çok kez - 17
43 : en iyi - 17
44 : bir gün - 17
45 : kimi zaman - 17
46 : en az - 16
47 : bir sürü - 16
48 : öyle bir - 15
49 : bir insan - 15
50 : her yerde - 15
Elapsed time is 1085.7234001159668 ms
```

```
C:\Users\Lenovo\Desktop\dist\2017510067.exe
Enter file name (ppp.txt) : DENEMELER.txt
Enter n value : 2
1 : o kadar - 75
2 : kitap 3 - 65
3 : 3 bölüm - 65
4 : ne kadar - 56
5 : kitap 1 - 56
6 : 1 bölüm - 56
7 : kitap 2 - 55
8 : 2 bölüm - 55
9 : bir şey - 51
10 : ya da - 49
11 : her şeyi - 42
12 : başka bir - 40
13 : bu kadar - 37
14 : büyük bir - 37
15 : hiç de - 34
16 : daha fazla - 32
17 : her şey - 32
18 : daha çok - 31
19 : hem de - 28
20 : her zaman - 27
21 : bölüm 12 - 27
22 : ki bu - 27
23 : böyle bir - 26
```

```
24 : çok daha - 25
25 : onun için - 25
26 : der ki - 24
27 : bir tek - 23
28 : daha az - 23
29 : için de - 23
30 : en çok - 22
31 : her gün - 22
32 : olduğu için - 22
33 : o zaman - 21
34 : ne de - 21
35 : daha iyi - 20
36 : ve daha - 20
37 : en büyük - 20
38 : her şeyin - 20
39 : var ki - 19
40 : o da - 19
41 : hiçbir şey - 18
42 : çok kez - 17
43 : en iyi - 17
44 : bir gün - 17
45 : kimi zaman - 17
46 : en az - 16
47 : bir sürü - 16
48 : öyle bir - 15
49 : bir insan - 15
50 : her yerde - 15
Elapsed time is 78.95135879516602 ms
```

TEST RESULTS

C# Code

When n=1

BİLİM İŞ BAŞINDA.txt → the results vary between 30ms and 45ms on average.

BOZKIRDA.txt → the results vary between 52ms and 72ms on average.

DEĞİŞİM.txt → the results vary between 32ms and 58ms on average.

DENEMELER.txt → the results vary between 895ms and 1390ms on average.

grimms-fairy-tales_P1.txt → vary between 24ms and 35ms on average.

When n=2

BİLİM İŞ BAŞINDA.txt → the results vary between 40ms and 50ms on average.

BOZKIRDA.txt → the results vary between 48ms and 75ms on average.

DEĞİŞİM.txt → the results vary between 35ms and 58ms on average.

DENEMELER.txt → the results vary between 900ms and 1450ms on average.

grimms-fairy-tales_P1.txt → vary between 25ms and 45ms on average.

When n=3

BİLİM İŞ BAŞINDA.txt → the results vary between 35ms and 50ms on average.

BOZKIRDA.txt → the results vary between 70ms and 80ms on average.

DEĞİŞİM.txt → the results vary between 37ms and 58ms on average.

DENEMELER.txt → the results vary between 890ms and 1350ms on average.

grimms-fairy-tales_P1.txt → vary between 27ms and 40ms on average.

When the code of the program written in C# is run in the Visual Studio environment and in the file with the .exe extension, they run at the same run times on average.

The program code written in C# is like this according to the average running time intervals n and file names. In the program written in C#, we can conclude that the larger the file size, the longer the runtime can be, by looking at the runtime of the DENEMELER text file.

Python Code (.exe)

When n=1

BİLİM İŞ BAŞINDA.txt → the results vary between 24ms and 30ms on average.

BOZKIRDA.txt → the results vary between 27ms and 45ms on average.

DEĞİŞİM.txt → the results vary between 33ms and 50ms on average.

DENEMELER.txt → the results vary between 62ms and 100ms on average.

grimms-fairy-tales_P1.txt → vary between 21ms and 25ms on average.

When n=2

BİLİM İŞ BAŞINDA.txt → the results vary between 20ms and 35ms on average.

BOZKIRDA.txt → the results vary between 32ms and 45ms on average.

DEĞİŞİM.txt → the results vary between 30ms and 52ms on average.

DENEMELER.txt → the results vary between 75ms and 115ms on average.

grimms-fairy-tales_P1.txt → vary between 18ms and 27ms on average.

When n=3

BİLİM İŞ BAŞINDA.txt → the results vary between 22ms and 38ms on average.

BOZKIRDA.txt → the results vary between 33ms and 48ms on average.

DEĞİŞİM.txt → the results vary between 40ms and 48ms on average.

DENEMELER.txt → the results vary between 85ms and 120ms on average.

grimms-fairy-tales_P1.txt → vary between 19ms and 30ms on average.

When the code of the program written in Python is run using the Python IDE, the running time varies between 600 and 1100 milliseconds. When the same code is run with the “.exe” extension, the running times are much shorter as given above.

Comparison Of The Program Written In C# And The Program Written In Python:

Outputs of n gram algorithms written in C# and Python give the same results. To find the running times of the programs written in C# and Python, elapsed times were calculated in milliseconds. While the program of the code written in Python works in the .exe file with an approximate running time, as in C#, the runtime increases as the file size grows. For example, since the size of the DENEMELER text file is larger than the size of other text files, the running time of the program takes more time.

The running time of the program written in C# code is much shorter than the running time of the .py file in the Python IDE of the program written in Python. When the program code written in Python is run with the .exe extension, they are very close to the running time of the code of the program written in C#. However, when run with .exe extension in Python, the DENEMELER file, which has a larger file size compared to other files, is much shorter than the run time in C#. If we are going to run the program written in C# and Python code with .exe extension, Python offers us a slightly better performance than C#. However, if the code of the program written in Python is run with a .py extension, the code of the program written in C # will give us faster results. Based on these observations, it would be a better choice to choose the C# programming language for the conversion of text files to n gram format, considering the time.

REFERENCE OF THE ALGORITHM

<https://stackoverflow.com/questions/3656762/n-gram-generation-from-a-sentence>

<https://www.techcoil.com/blog/how-to-generate-n-grams-in-python-without-using-any-external-libraries/>