Міністерство освіти і науки України Національний технічний університет України «Київський політехнічний інститут» Фізико-технічний інститут

**Лабораторна робота №1**

З теми:

Експериментальна оцінка ентропії на символ джерела відкритого тексту

**Перевірила:**

Селюх К. І.

**Виконали:**

студенти ІІІ курсу

групи ФБ-95

Гурджия Валерія

групи ФБ-94

Золотов Іван

**Мета роботи**: Засвоєння понять ентропії на символ джерела та його надлишковості, вивчення та порівняння різних моделей джерела відкритого тексту для наближеного визначення ентропії, набуття практичних навичок щодо оцінки ентропії на символ джерела.

**Завдання:**

1. Написати програми для підрахунку частот букв і частот біграм в тексті, а також підрахунку H1 та H2 за безпосереднім означенням. Підрахувати частоти букв та біграм, а також значення H1 та H2 на довільно обраному тексті російською мовою достатньої довжини (щонайменше 1Мб), де імовірності замінити відповідними частотами. Також одержати значення H1 та H2 на тому ж тексті, в якому вилучено всі пробіли.

2. За допомогою програми CoolPinkProgram оцінити значення (10) H , (20) H , (30) H .

3. Використовуючи отримані значення ентропії, оцінити надлишковість російської мови в різних моделях джерела.

**Код програми**

Для написання даної лабораторної роботи ми використовували мову програмування python.

from array import \*

from math import\*

file\_source = "C:\\Users\\User\\Desktop\\Folder\\KPI\\5th\_term\\Криптография\\lab\_1\\file1.txt"

file\_noSpace = "C:\\Users\\User\\Desktop\\Folder\\KPI\\5th\_term\\Криптография\\lab\_1\\file2\_noSpace.txt"

file\_Space = "C:\\Users\\User\\Desktop\\Folder\\KPI\\5th\_term\\Криптография\\lab\_1\\file3\_Space.txt"

file\_probability\_noSpace = "C:\\Users\\User\\Desktop\\Folder\\KPI\\5th\_term\\Криптография\\lab\_1\\file4\_probability\_noSpace.txt"

file\_probability\_Space = "C:\\Users\\User\\Desktop\\Folder\\KPI\\5th\_term\\Криптография\\lab\_1\\file5\_probability\_Space.txt"

file\_bigramms\_Space\_1 = "C:\\Users\\User\\Desktop\\Folder\\KPI\\5th\_term\\Криптография\\lab\_1\\file6\_bigramms\_Space\_1.txt"

file\_bigramms\_Space\_2 = "C:\\Users\\User\\Desktop\\Folder\\KPI\\5th\_term\\Криптография\\lab\_1\\file7\_bigramms\_Space\_2.txt"

file\_bigramms\_noSpace\_1 = "C:\\Users\\User\\Desktop\\Folder\\KPI\\5th\_term\\Криптография\\lab\_1\\file8\_bigramms\_noSpace\_1.txt"

file\_bigramms\_noSpace\_2 = "C:\\Users\\User\\Desktop\\Folder\\KPI\\5th\_term\\Криптография\\lab\_1\\file9\_bigramms\_noSpace\_2.txt"

Alphabet1=['а','б','в','г','д','е','ё','ж','з','и','й','к','л','м','н','о','п','р','с','т','у','ф','х','ц','ч','ш','щ','ы','ь','ъ','э','ю','я']

Alphabet2=['а','б','в','г','д','е','ё','ж','з','и','й','к','л','м','н','о','п','р','с','т','у','ф','х','ц','ч','ш','щ','ы','ь','ъ','э','ю','я', ' ']

file1 = open(file\_source, "r")

cont\_source = file1.read()

###################################################################################

###############################   Открываем файл 2     ############################

file2 = open(file\_noSpace, "w")

# в этой функции мы меняем пpописные буквы на строчные и убраем пробелы

def swap\_case(s):

    swapped = []

    for char in s:

        if char.isalpha() == False:  # убраем все лишние символы

            char = ''

        if char.isupper():  # меняем прописные буквы на строчные

            swapped.append(char.lower())

        else:

            swapped.append(char)

    return ''.join(swapped)

file2.write(swap\_case(cont\_source))

file2.close()

###########  тут стираются буквы, которых нет в алфавите ##################

file2 = open(file\_noSpace, "r")

cont\_noSpace = file2.read()

file2.close()

file2 = open(file\_noSpace, "w")

def alph(s):

    arr = []

    for char in cont\_noSpace:

        if char == 'ё':

            char = 'е'

        if char == 'ъ':

            char = 'ь'

        if char not in Alphabet1:

            char = ''

        else:

            arr.append(char)

    return ''.join(arr)

file2.write(alph(cont\_noSpace))

file2.close()

#####################    Частота букв в файле 4     ###############################

file2 = open(file\_noSpace, "r")

cont\_noSpace = file2.read()

arr = []

thisset = set()

for char in cont\_noSpace:

    thisset.add(char)

for c in thisset:

    arr.append(c)

arr.sort()

for i in range(0,len(arr)):

    k = (cont\_noSpace.count(arr[i]))/len(cont\_noSpace)

    strochka1 = arr[i] + "\t " +str(cont\_noSpace.count(arr[i])) + "\t " +  str(k)

    file4 = open(file\_probability\_noSpace, "a")

    file4.write(strochka1 + '\n')

    file4.close()

file2.close()

####################################################################################

############################   Открываем файл 3     ################################

file3 = open(file\_Space, "w")

# в этой функции мы меняем пpописные буквы на строчные

def swap\_case(s):

    swapped = []

    for char in s:

        if char.isupper():  # меняем прописные буквы на строчные

            swapped.append(char.lower())

        else:

            swapped.append(char)

    return ''.join(swapped)

file3.write(swap\_case(cont\_source))

file3.close()

###########  тут стираются буквы, которых нет в алфавите ##################

file3 = open(file\_Space, "r")

cont\_Space = file3.read()

file3.close()

file3 = open(file\_Space, "w")

def alph(s):

    arr = []

    for char in cont\_Space:

        if char == '\n':

            char = ' '

        if char == 'ё':

            char = 'е'

        if char == 'ъ':

            char = 'ь'

        if char not in Alphabet2:

            char = ''

        else:

            arr.append(char)

    return ''.join(arr)

###########  тут стираются ненужные пробелы ##################

def space(s):

    s = s.split()

    return ' '.join(s)

string = alph(cont\_Space)

file3.write(space(string))

file3.close()

#####################   Частота букв в файле 3     ###############################

file3 = open(file\_Space, "r")

cont\_Space = file3.read()

arr = []

thisset = set()

for char in cont\_Space:

    thisset.add(char)

for c in thisset:

    arr.append(c)

arr.sort()

for i in range(0,len(arr)):

    k = (cont\_Space.count(arr[i]))/len(cont\_Space)

    strochka2 = arr[i] + "\t " +str(cont\_Space.count(arr[i])) + "\t " +  str(k)

    file5 = open(file\_probability\_Space, "a")

    file5.write(strochka2 + '\n')

    file5.close()

file3.close()

#######################################################################################

###############################     Биграммы    #######################################

#############################  Файл 2 (без пробелов) с шагом 1      ###################

file2 = open(file\_noSpace, "r")

cont\_noSpace = file2.read()

arr1 = []

j = 0

for i in range(0,len(cont\_noSpace)-1):

    char = cont\_noSpace[i] + cont\_noSpace[i+1]

    arr1.append(char)

    j+=1

arr = []

thisset = set()

for char in arr1:

    thisset.add(char)

for c in thisset:

    arr.append(c)

arr.sort()

for i in range(0, len(arr)):

    k = (cont\_noSpace.count(arr[i]))/j

    strochka3 = arr[i] + "\t " +str(cont\_noSpace.count(arr[i])) + "\t " +  str(k)

    file6 = open(file\_bigramms\_noSpace\_1, "a")

    file6.write(strochka3 + '\n')

    file6.close()

file2.close()

##########################   Файл 2 (без пробелов) с шагом 2      ####################

file2 = open(file\_noSpace, "r")

cont\_noSpace = file2.read()

arr1 = []

j = 0

i = 0

while i < len(cont\_noSpace)-1:

    char = cont\_noSpace[i] + cont\_noSpace[i+1]

    arr1.append(char)

    i+=2

    j+=1

arr = []

thisset = set()

for char in arr1:

    thisset.add(char)

for c in thisset:

    arr.append(c)

arr.sort()

for i in range(0, len(arr)):

    k = (cont\_noSpace.count(arr[i]))/j

    strochka3 = arr[i] + "\t " +str(cont\_noSpace.count(arr[i])) + "\t " +  str(k)

    file7 = open(file\_bigramms\_noSpace\_2, "a")

    file7.write(strochka3 + '\n')

    file7.close()

file2.close()

##########################   Файл 3 (с пробелами) с шагом 1      #####################

file3 = open(file\_Space, "r")

cont\_Space = file3.read()

arr1 = []

j = 0

for i in range(0,len(cont\_Space)-1):

    char = cont\_Space[i] + cont\_Space[i+1]

    arr1.append(char)

    j+=1

arr = []

thisset = set()

for char in arr1:

    thisset.add(char)

for c in thisset:

    arr.append(c)

arr.sort()

for i in range(0, len(arr)):

    k = (cont\_Space.count(arr[i]))/j

    strochka3 = arr[i] + "\t " +str(cont\_Space.count(arr[i])) + "\t " +  str(k)

    file8 = open(file\_bigramms\_Space\_1, "a")

    file8.write(strochka3 + '\n')

    file8.close()

file3.close()

########################   Файл 3 (с пробелами) с шагом 2      #######################

file3 = open(file\_Space, "r")

cont\_Space = file3.read()

arr1 = []

j = 0

i = 0

while i < len(cont\_Space)-1:

    char = cont\_Space[i] + cont\_Space[i+1]

    arr1.append(char)

    i+=2

    j += 1

arr = []

thisset = set()

for char in arr1:

    thisset.add(char)

for c in thisset:

    arr.append(c)

arr.sort()

for i in range(0, len(arr)):

    k = (cont\_Space.count(arr[i]))/j

    strochka3 = arr[i] + "\t " +str(cont\_Space.count(arr[i])) + "\t " +  str(k)

    file9 = open(file\_bigramms\_Space\_2, "a")

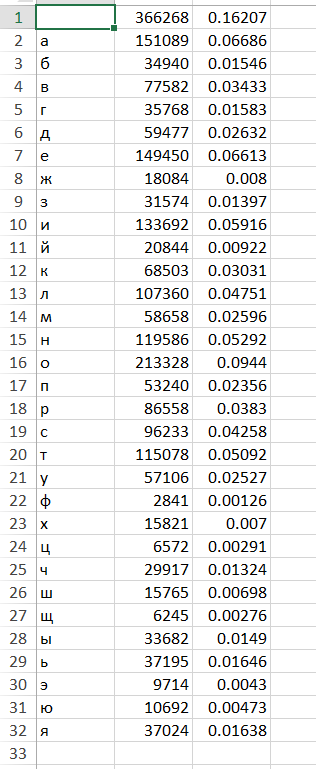
    file9.write(strochka3 + '\n')

    file9.close()

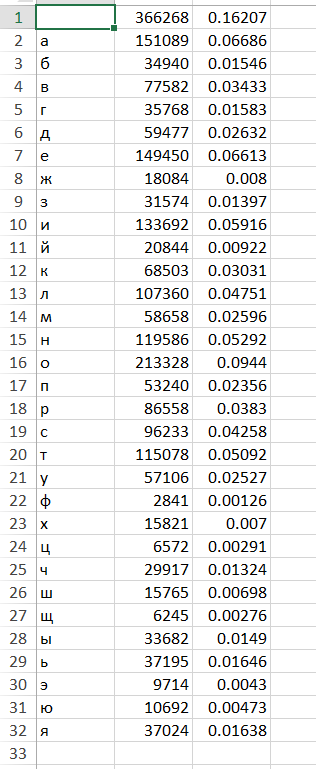
file3.close()

**Скріншоти виконання**

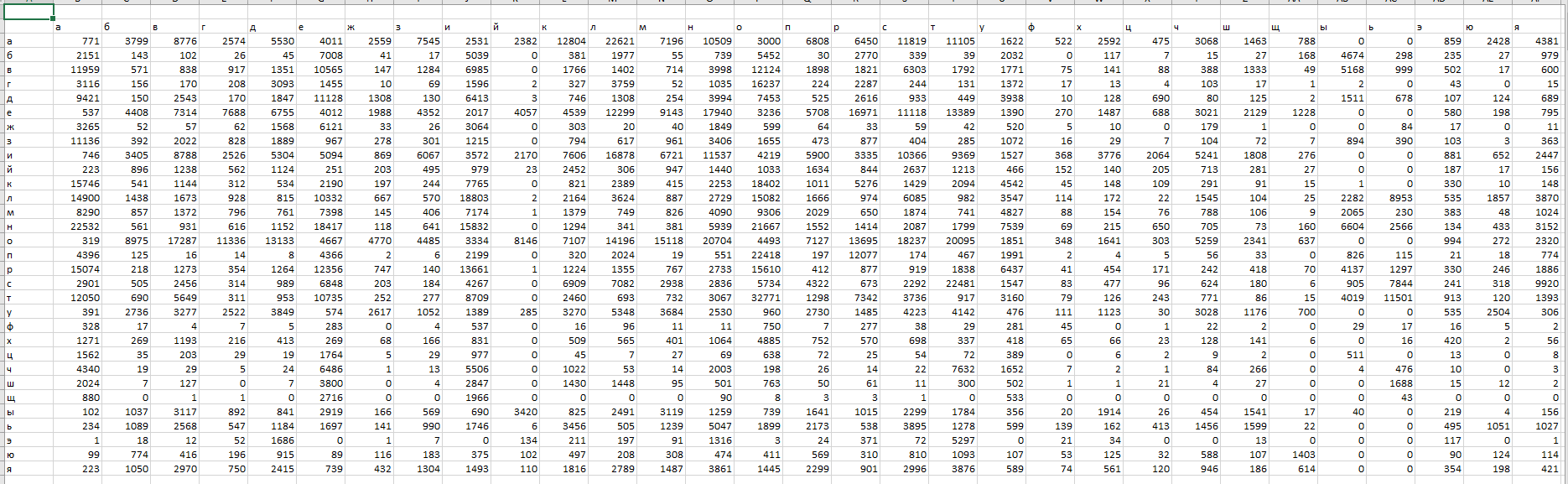
Ймовірність та кількість букв у файлі без пробілів



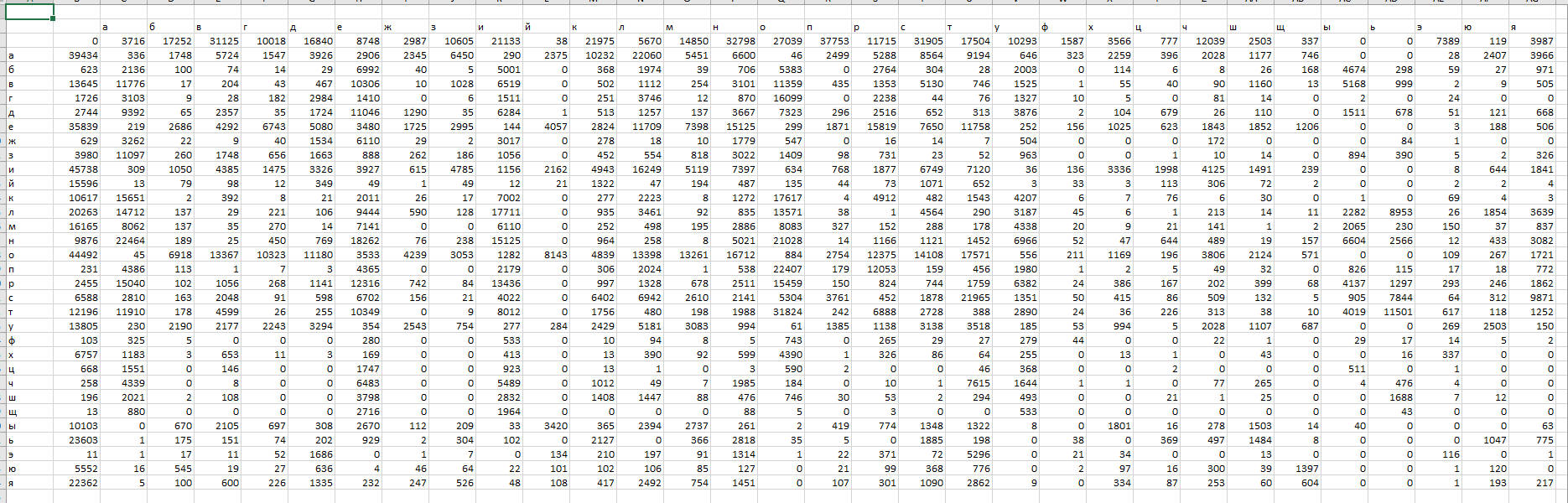
Ймовірність та кількість букв у файлі з пробілами



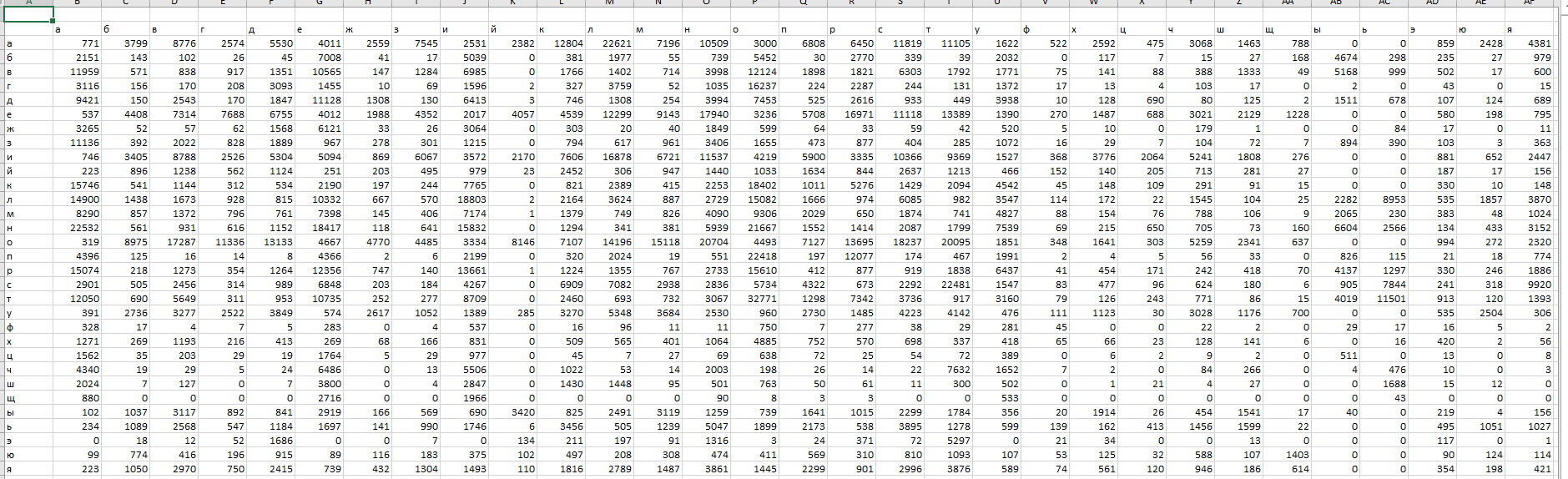
Частота біграм з шагом 1 у файлі без пробілів



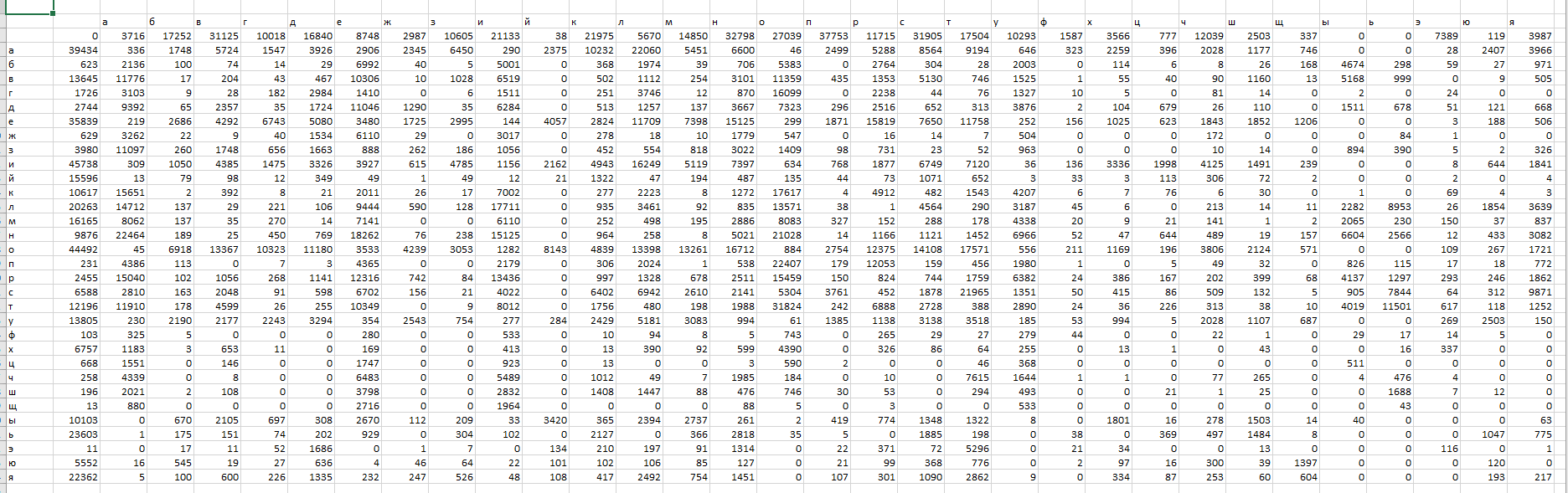
Частота біграм з шагом 1 у файлі з пробілами



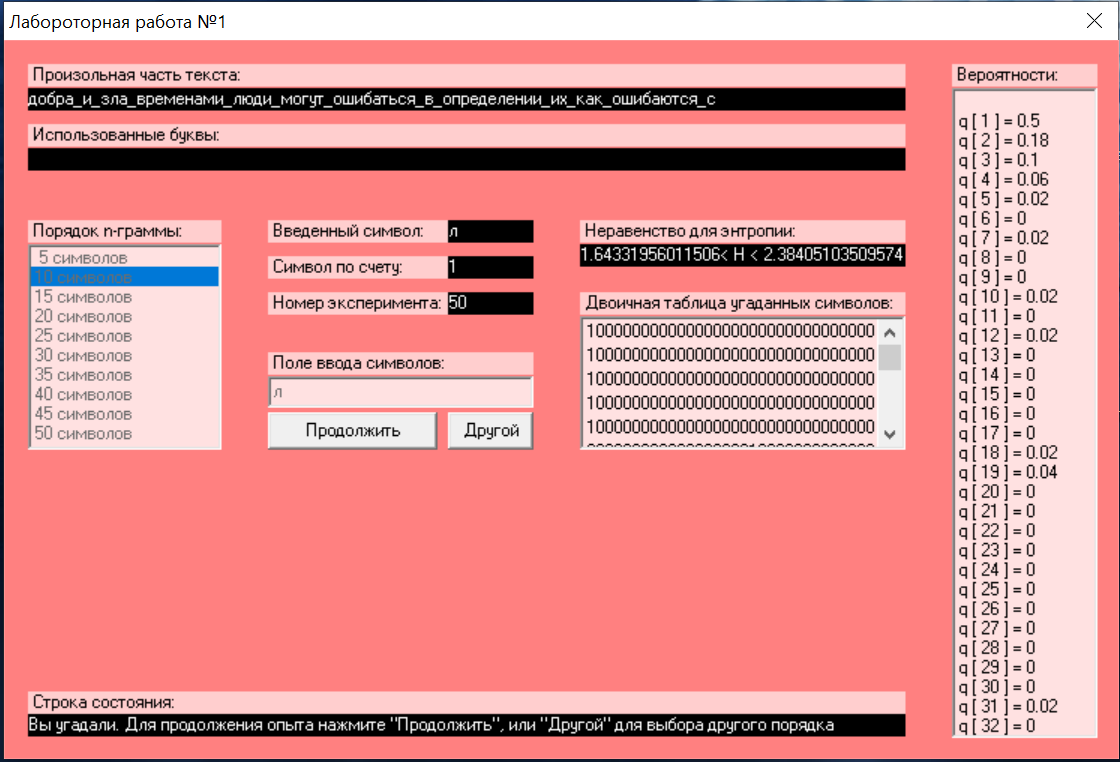
Частота біграм з шагом 2 у файлі без пробілів



Частота біграм з шагом 2 у файлі з пробілами

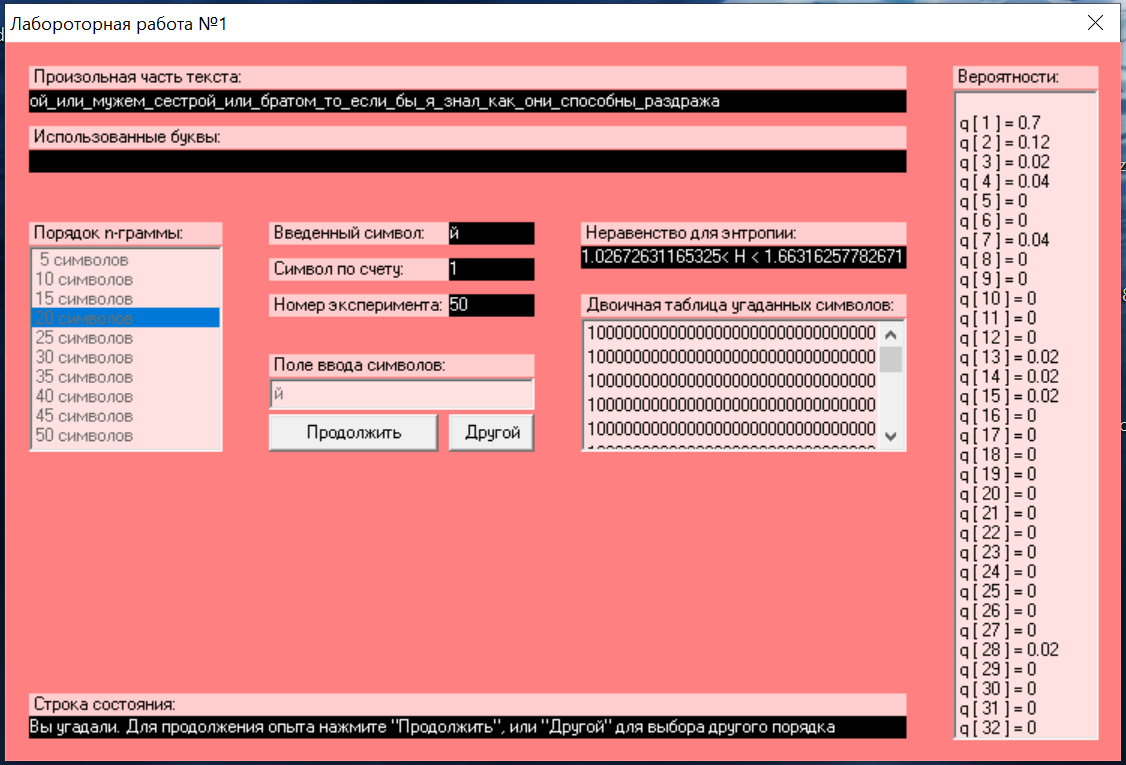


**H10**

****

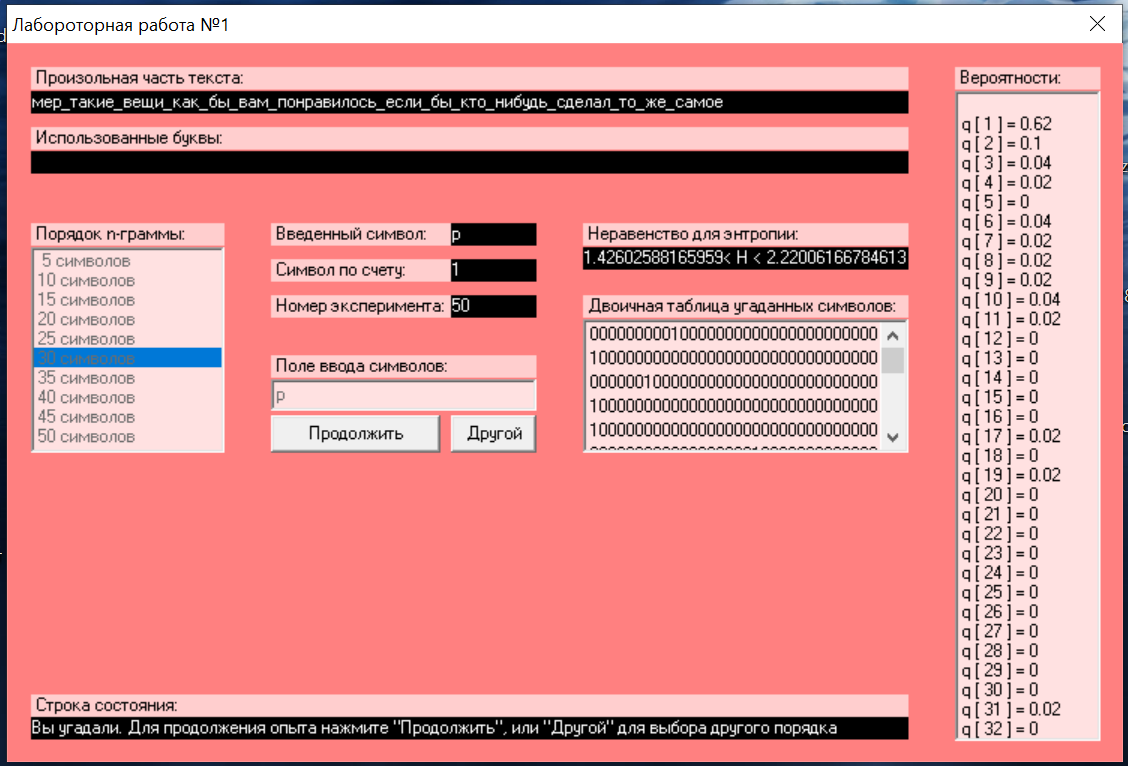
0.494688480055972 < R < 0.7176708727575805

**H20**

****

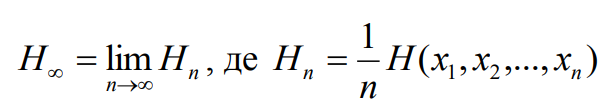
0.23756209939443898 < R < 0.3848195808139691

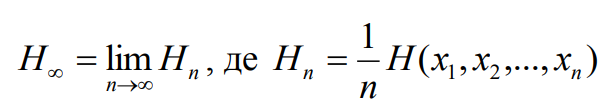
**H30**

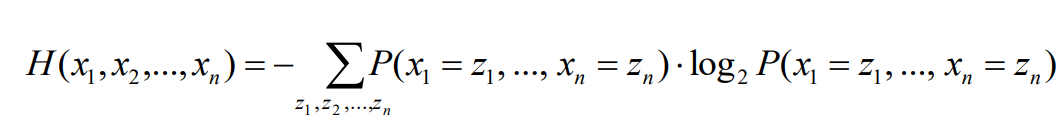
****

0.29061701170509674 < R < 0.4524375721425297

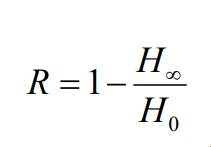
Ентропія на символ стаціонарного джерела визначається як

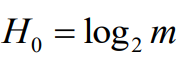
****

****

****

Надлишковість джерела відкритого тексту:

****

****

|  |  |  |
| --- | --- | --- |
|  | **Ентропія** | **Надлишковість** |
| **Без пробілів з шагом 1** | 4.167734 | 0.158747 |
| **Без пробілів з шагом 2** | 3.167661 | 0.360611 |
| **З пробілами з шагом 1** | 3.994742 | 0.201052 |
| **З пробілами з шагом 2** | 3994742 | 0.201052 |

**Висновок:** виконуючи цю лабораторну роботу, ми навчилися підраховувати частоти біграм та монограм, визначати ентропію на символ джерела та надлишковості джерела відкритого тексту.