Київський національний університет імені Тараса Шевченка
Факультет радіофізики, електроніки та компютерних систем

Лабораторна робота №1
Дослідження кількості інформації при різних варіантах кодування

Виконав студент
2 курсу СА-КІ
Глушко Гліб
github/glebglushko/compsys

Київ 2019

1. Дослідження кількості інформації в тексті

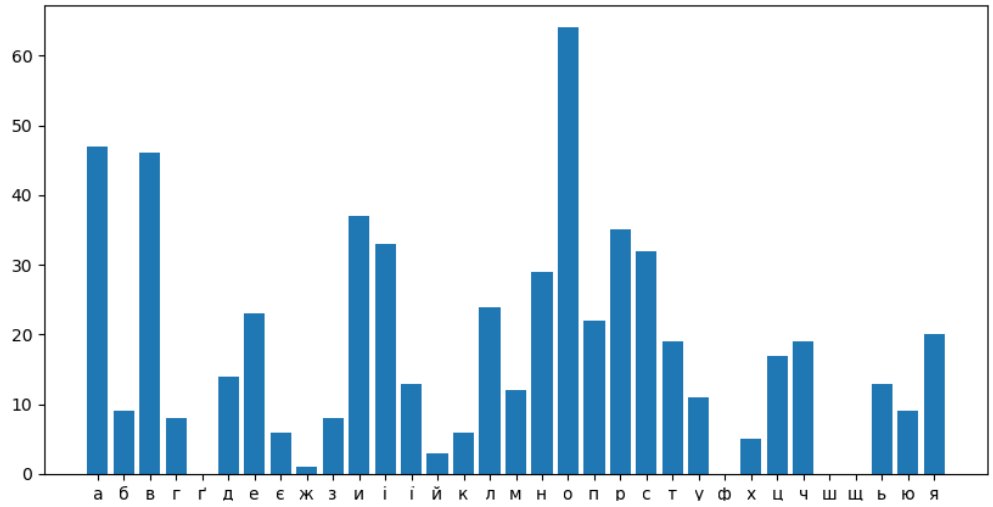
1. Оберіть 3 текстових файла різного тематичного та лінгвістичного спрямування

   Я вибрав тексти:

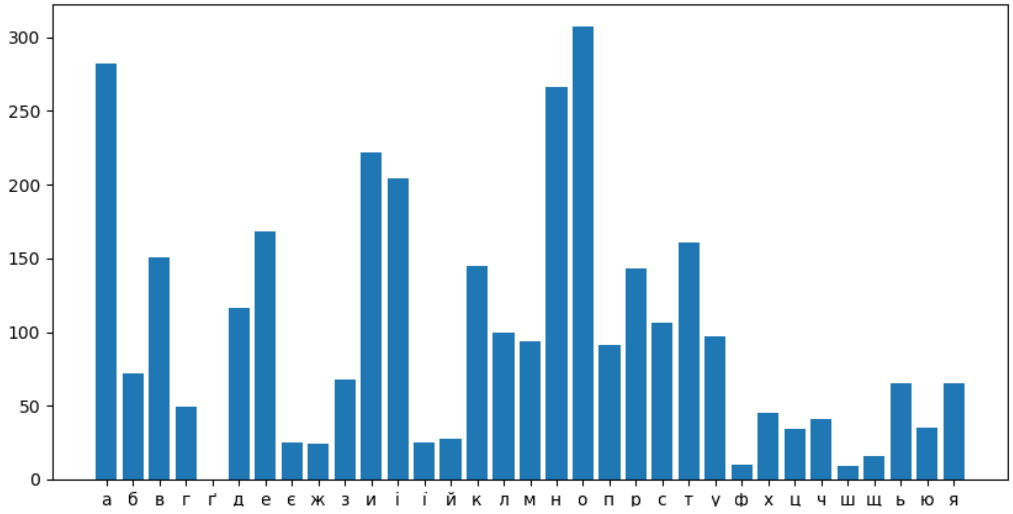
   1) Пісня Олега Винника – Вовчиця
   2) Стаття про петицію щодо легазізації медицинського канабісу, та плюси і мінуси канабісу
   3) Докер, основні можливості і процеси

2. Створіть програму (будь-якою зручною для вас мовою), яка в якості вхідних даних приймає текстовий файл, та аналізуючи його вміст:
   a. обраховує частоти (імовірності) появи символів в тексті
   b. обраховує середню ентропію алфавіту для даного тексту
   c. виходячи з ентропії визначає кількість інформації та порівнює її з розмірами файлів
   d. виводить на екран значення частот, ентропії та кількості інформації

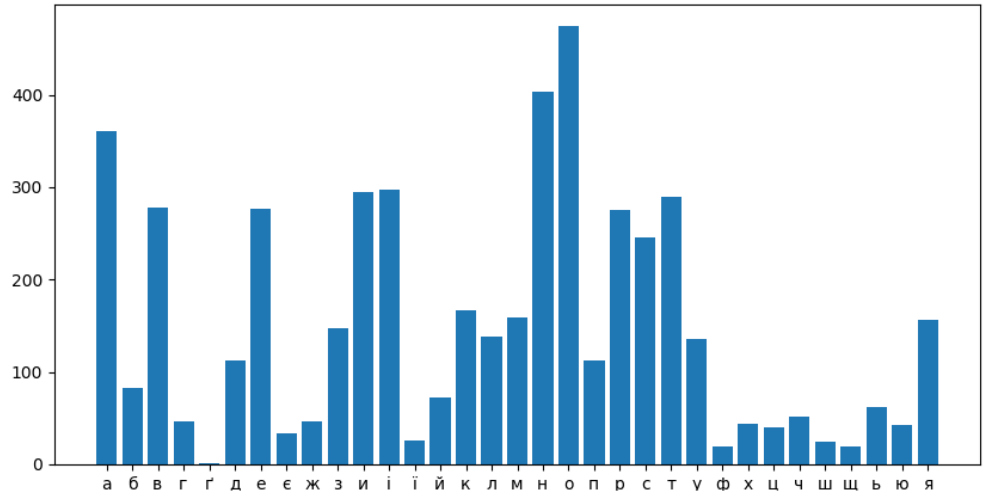
```
vinnyk.txt
entropy = 4.4891538663370945,
letters_count = 585,
total_symbols_count = 792,
file size = 1464,
predicted_file_size = 1313.0775059036002,
```



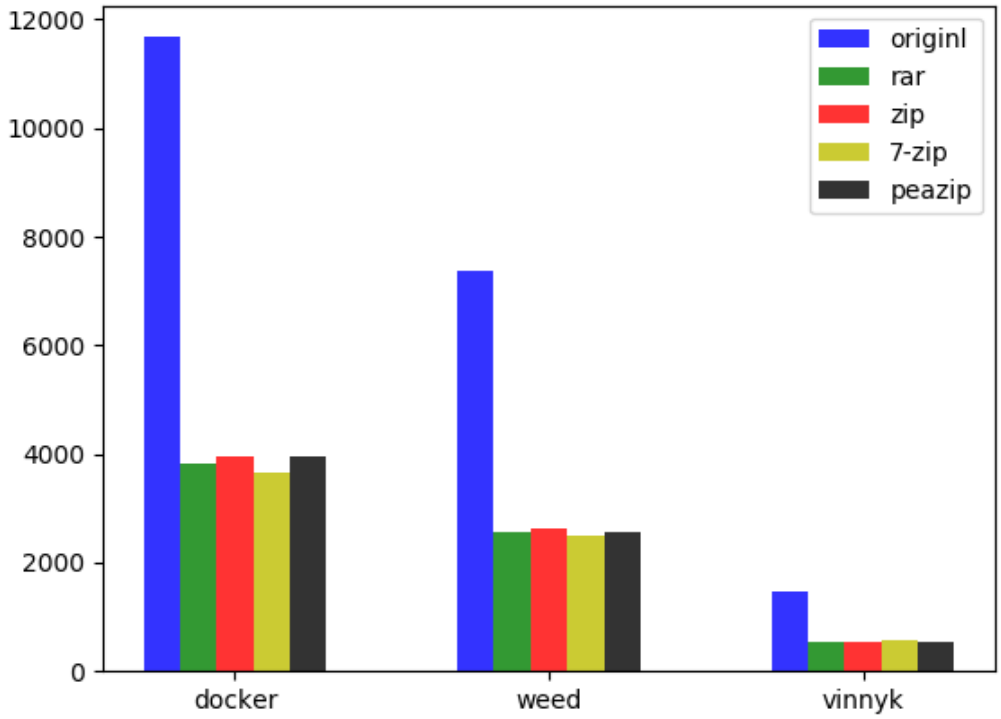
```
weed.txt
entropy = 4.5615542393162976,
letters_count = 3264,
total_symbols_count = 4021,
file size = 7381,
predicted_file_size = 7444.456518564198,
```




```
docker.txt
entropy = 4.539268288356095,
letters_count = 4938,
total_symbols_count = 6631,
file size = 11675,
predicted_file_size = 11207.4534039512,
```



3. Проведіть стиснення кожного вхідного файлу за допомогою 5 різних алгоритмів стиснення (zip, rar, gzip, bzip2, xz, або будь-які інші на ваш вибір, можна використовувати готові програмні засоби для стиснення).




```
archive_name = ('orginal', 'rar', 'zip', '7-zip','PeaZip')
archive_size_docker = (11675, 3809, 3961, 3643, 3953)
archive_size_weed = (7381, 2551, 2616, 2480, 2556)
archive_size_vinnyk = (1464, 533, 533, 555, 542)
```

2. Дослідження способів кодування інформації на прикладі Base64

1. Для практичного засвоєння методу кодування, створіть програму, що кодує довільний файл в Base64 (шляхом реалізації алгоритму вручну, а не виклику бібліотечної функції)

   Мій алгоритм

```
def tobase64(s):
    bs = ''
    res = ''
    for x in s:
        bs+= str(bin(int.from_bytes(bytearray(x, 'utf-8'), byteorder='big')))[2:].zfill(8
        while len(bs) >= 6:
            res += b64s[int(bs[:6], 2)]
            bs = bs[6:]
    if len(bs) > 0:
        while len(bs) < 6:
            bs += '0'
        res += b64s[int(bs[:6], 2)]

    return res
```

   Використання бібліотеки **base64**

```
libtxt64 = str(base64.b64encode(bytes(txt, encoding='utf-8')))[2:-1]
```

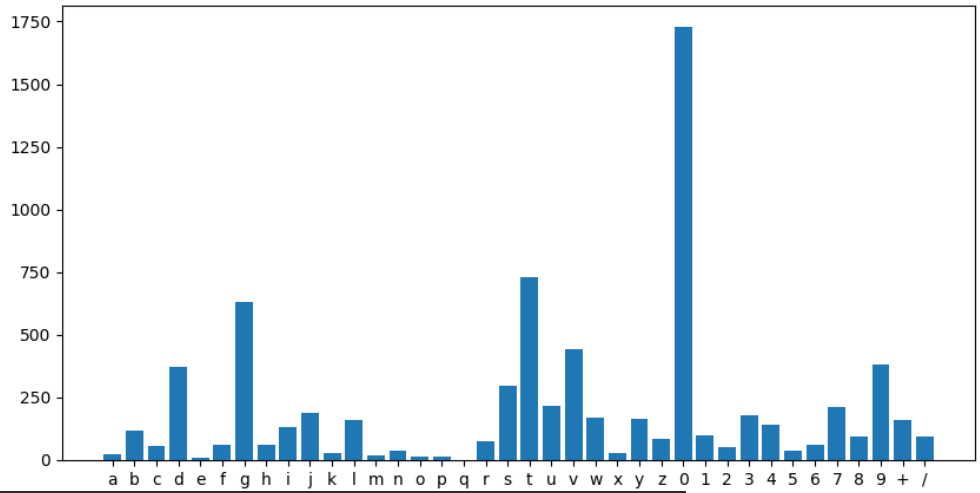
a. перевірте коректність роботи програми, порівнявши результат з існуючими програмними засобами

```
input:  Glory To Ukraine
my result:   R2xvcnkgVG8gVWtyYWluZQ==
lib result:  R2xvcnkgVG8gVWtyYWluZQ==
True
```

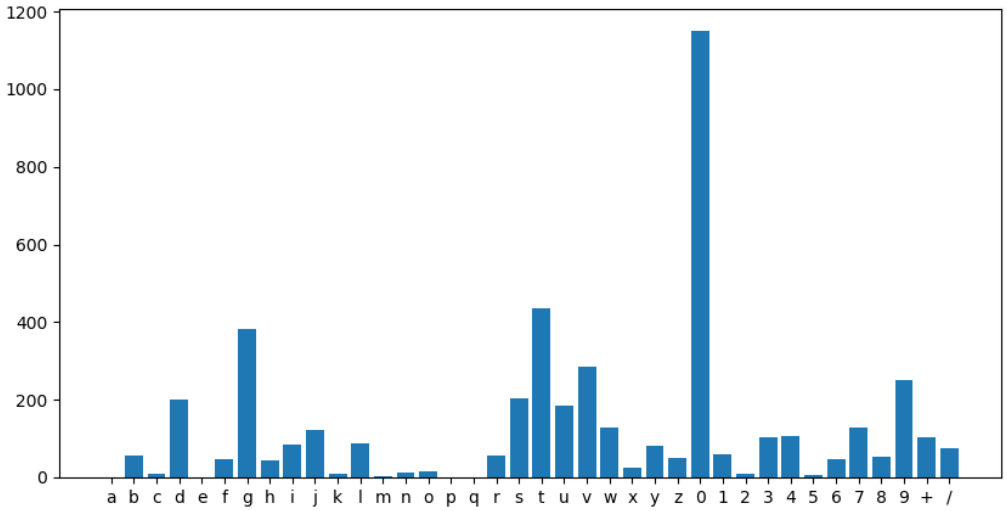
```
input:  Слава Україні!!
my result:   0KHQu9Cw0LLQsCDQo9C60YDQsNGX0L3RliEh
lib result:  0KHQu9Cw0LLQsCDQo9C60YDQsNGX0L3RliEh
True
```

```
Хтось її почує,
Як її пахне любов
Знов вона відчує.
Приспів:
Зорі прозорі вгорі,
Вам спиться, чи не спиться?
Вила на місяць новий
Молода вовчиця,
Вірила, там угорі
Хтось її почує,
Як її пахне любов
Знов вона відчує.
my result:  77u/0J/QsNC00LDQsiDQsdC10LfQttCw0LvRltGB0L3QviDQs9GA0LDQtCwK0JHQuNC70LAg0LHQu9C40YHQutCw0LLQuNGG0Y8sCtCT0YDRltC70LAg0L/RltC0INGB0LXRgNGG0LXQvCDQstC
 +0LLRh9Cw0YIK0JzQvtC70L7QtNCwINCy0L7QstGH0LjRhtGPLArQl9C90LDRjiDQv9GA0L4g0LLRltGA0L3RgyDQu9GO0LHQvtCyLArQm9GW0YHQvtCy0LUg0L7Qt9C10YDRhtC1LArQn9C70LDRh9C1INC90ZbRh9C90L7RlyDQv9C
 +0YDQuArQntC00LjQvdC+0LrQtSDRgdC10YDRhtC1LgrQn9GA0LjRgdC/0ZbQsjoK0JfQvtGA0ZYg0L/RgNC+0LfQvtGA0ZYg0LLQs9C+0YDRliwK0JLQsNC8INGB0L/QuNGC0YzRgdGPLCDRh9C4INC90LUg0YHQv9C40YLRjNGB0Y8
 /CtCS0LjQu9CwINC90LAg0LzRltGB0Y/RhtGMINC90L7QstC40LkK0JzQvtC7L7QtNCwINCy0L7QstGH0LjRhtGPLArQktGW0YDQuNC70LAsINGC0LDQvCDRg9Cz0L7RgNGWCtCl0YLQvtGB0Ywg0ZfRlyDQv9C+0YfRg9GULArQr9C6INGX0Zcg0L
 /QsNGF0L3QtSDQu9GO0LHQvtCyCtCX0L3QvtCyINCy0L7QvdCwINCy0ZbQtNGH0YPRlC4K0JfQs9GA0LDRjiDRgtGA0LjQvNCw0LvQsCDQsiDQt9GD0LHQsNGFCtCh0YLQtdC
 /0L7QstCwINGG0LDRgNC40YbRjywK0JLRltGH0L3QviDQvdCw0LLQvtC00LjQu9CwINGB0YLRgNCw0YUK0KPQtNC+0LQsC3QstC+0LLRh9C40YbRjywK0JIgOL3QtdCx0ZYg0Lcn0Y/QstC40LvQvtGB0Y8g0LfQvdC+0LIK0JzRltGB0Y
 /RhtGPINC70Y7RgdGC0LXRgNGG0LUsCtCR0LDRh9C40YLRjCDRgdCyOL7RjiDRgtCw0Lwg0LvRjtCx0L7QsgrQntC00LjQvdC+0LrQtSDRgdC10YDRhtC1LgrQn9GA0LjRgdC/0ZbQsjoK0JfQvtGA0ZYg0L/RgNC+0LfQvtGA0ZYg0LLQs9C
 +0YDRliwK0JLQsNC8INGB0L/QuNGC0YzRgdGPLCDRh9C4INC90LUg0YHQv9C40YLRjNGB0Y8/CtCS0LjQu9CwINC8INGB0L/QuNGC0YzRgdGPLCDRh9C4INC90LUg0YHQv9C40YLRjNGB0Y8
 /RhtGMINC90L7QstC40LkK0JzQvtC7L7QtNCwINCy0L7QstGH0LjRhtGPLArQktGW0YDQuNC70LAsINGC0LDQvCDRg9Cz0L7RgNGWCtCl0YLQvtGB0Ywg0ZfRlyDQv9C+0YfRg9GULArQr9C6INGX0Zcg0L
 /QsNGF0L3QtSDQu9GO0LHQvtCyCtCX0L3QvtCyINCy0L7QvdCwINCy0ZbQtNGH0YPRlC4K0JfQs9GA0LDRjiDRgtGA0LjQvNCw0LvQsCDQsiDQt9GD0LHQsNGFCtCh0YLQtdC
 /0L7QstCwINGG0LDRgNC40YbRjywK0JLRltGH0L3QviDQvdCw0LLQvtC00LjQu9CwINGB0YLRgNCw0YUK0KPQtNC+0LQsC3QstC+0LLRh9C40YbRjywK0JIgOL3QtdCx0ZYg0Lcn0Y/QstC40LvQvtGB0Y8g0LfQvdC+0LIK0JzRltGB0Y
 /RhtGPINC70Y7RgdGC0LXRgNGG0LUsCtCR0LDRh9C4INC90LUg0YHQv9C40YLRjNGB0Y8/CtCS0LjQu9CwINC8INGB0L/QuNGC0YzRgdGPLCDRh9C4INC90LUg0YHQv9C40YLRjNGB0Y8
 +0YDRliwK0JLQsNC8INGB0L/QuNGC0YzRgdGPLCDRh9C4INC90LUg0YHQv9C40YLRjNGB0Y8/CtCS0LjQu9Cw
```

```
docker.txt
True
entropy = 4.246146498023844,
letters_count = 7372,
total_symbols_count = 15516,
file size = 15565,
predicted_file_size = 15651.295991715888,
```




```
weed.txt
True
entropy = 4.13324380032741,
letters_count = 4616,
total_symbols_count = 9808,
file size = 9857,
predicted_file_size = 9539.526691155663,
```



По-перше, я змінив алфавіт на англійський, по-друге, додав до алфавіту символи та цифри, які я використовув у кодуванні в **base64**, тому що, якщо рахувати лише символи, які є літерами, то виходить неправильне значення ентропії і неправильний результат ентропії, тому что текст складається с великої кількості цифр і символів, ними не можна знехтувати, як у текстах на оригінальній мові. Можна побачити, що цифра 0 дуже часто використовується

.

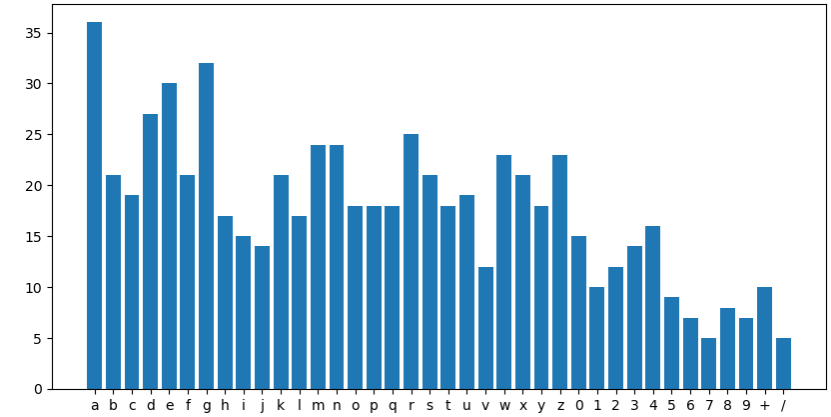
a. Порівняйте отримане значення з кількістю інформації вихідного файлу
b. Зробіть висновки з отриманого результату

У закодованому вигляді, розміри файлів збільшились у 1.33 рази, тому що тепер для 1 байт замінюєтсья на символи, розміри яких теж 1 байт, але 1символ кодує лише 6 біт. Після кодування, я обрахував ентропію і кількість інформації, і кількість інформації теж пропорційно збільшилась

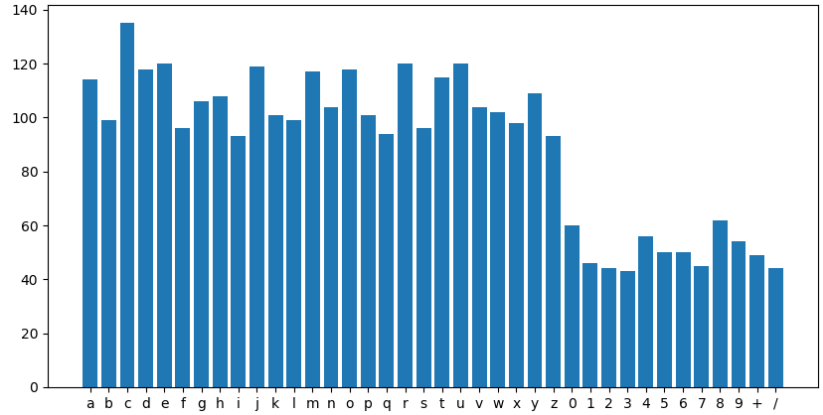
Закодуйте в Base64 стиснені кращим з алгоритмів текстові файли

1. Обрахуйте кількість інформації в base64-закодованому варіанті стисненого файлу
2. Порівняйте отримане значення з кількістю інформації вихідного файлу та base64-закодованого файлу

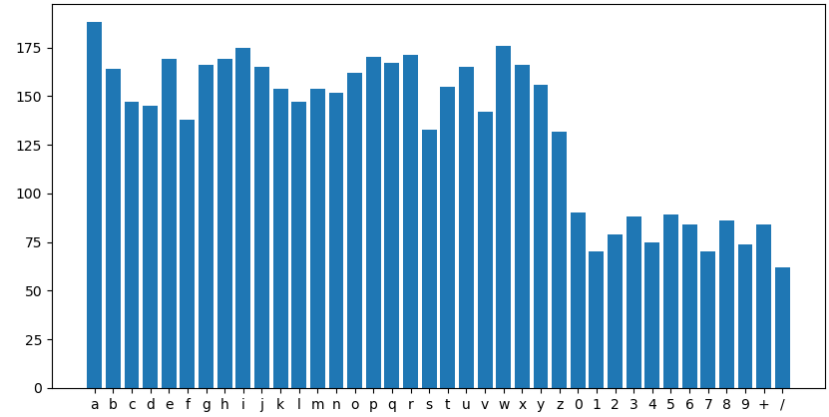
```
vinnyk.rar
entropy = 5.122065211699272,
letters_count = 670,
total_symbols_count = 672,
file size = 502,
predicted_file_size = 857.9459229596281,
```




```
weed.rar
entropy = 5.168992046299582,
letters_count = 3402,
total_symbols_count = 3404,
file size = 2551,
predicted_file_size = 4396.2277353777945,
```




```
docker.rar
entropy = 5.181218947076895,
letters_count = 5079,
total_symbols_count = 5080,
file size = 3809,
predicted_file_size = 6578.852758050887,
```



Як можна побачити, що у заархівованому вигляді, у нас більша ентропія і більше спрогнозованої інформації, тому що у заархівованому вигляді, файли намагаються, якнайкраще зберегти інформацію з меншим розміром, отже на один символ(байт) інформації буде більша кількість інформації. Це можна просто пояснити на прикладах, які краще розповісти усно, по бажанню викладача.

Висновок: Під час лабораторної роботи, було досліджено і розроблено алгоритм кодування у base64, було досліджено, що українські літери займають два байти у кодуванні UTF-8, було порівняно різні архіватори і результати архівавування. Досліджено явище ентропії і вирахувано кількість інформації, яка приблизно співпадає з реальними розмірами. Неточності можно пояснити, тим, що ми не брали до уваги деякі, символи, а лише алфавіт.