# МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА ФАКУЛЬТЕТ РАДІОФІЗИКИ, ЕЛЕКТРОНІКИ ТА КОМП’ЮТЕРНИХ СИСТЕМ

Кафедра комп’ютерної інженерії

# Звіт

**з лабораторної роботи №1**

«**Дослідження кількості інформації при різних варіантах кодування**»

Роботу виконав : студент 3-го курсу

напряму підготовки: «Комп’ютерна інженерія» спеціалізація: «Системний адміністратор»

Костюченко Данило

Київ 2019

**Мета**: Дослідити імовірнісні параметри української мови для оцінки кількості інформації текстів. Дослідити вплив різних методів кодування інформації на її кількість.

# Хід роботи

## Дослідження кількості інформації в тексті

1. Оберіть 3 текстових файла різного тематичного та лінгвістичного спрямування (наприклад, вірш Тараса Шевченка “Мені тринадцятий минало”, “Казка про рєпку” Леся Подерв'янського та специфікацію інерфейсу PCI)

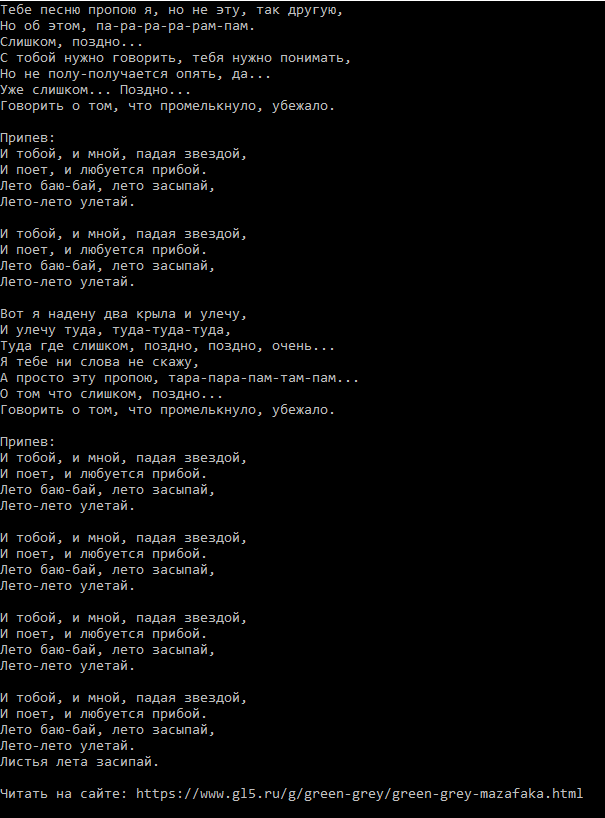
* [https://uk.wikipedia.org/wiki/REST](https://uk.wikipedia.org/wiki/REST" \o "REST\" )
* <https://uk.wikipedia.org/wiki/Bpm'online_CRM>
* <https://genius.com/Green-grey-mf-lyrics>

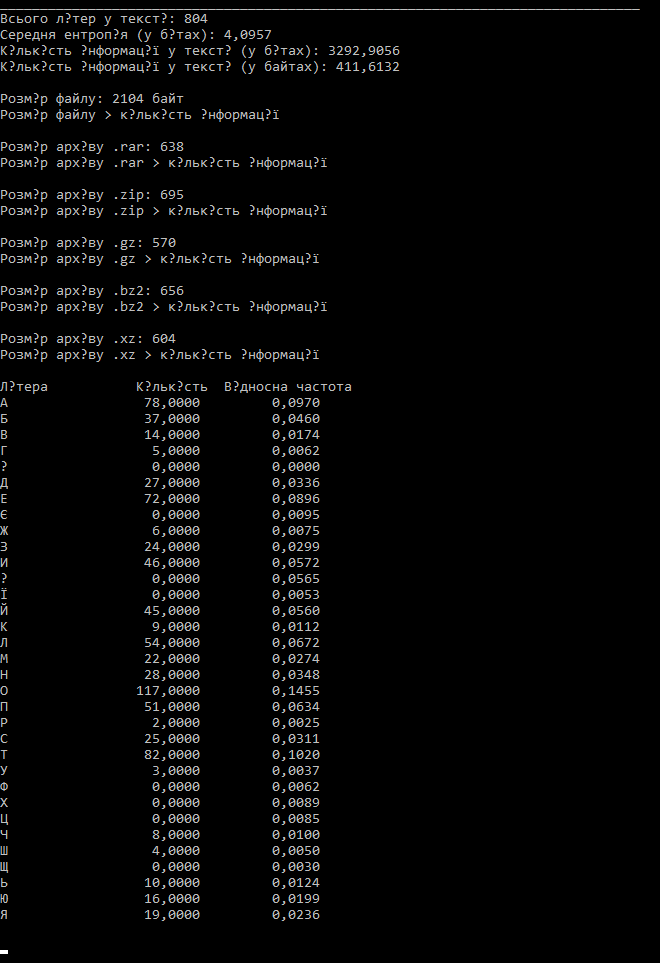
1. Переконайтесь, що тексти, які ви використовуєте є унікальними і не повторюються у ваших колег! Використовуйте наявні електронні засоби зв’язку та документообігу, щоб уникнути дублювання! Вдруге аналіз того самого тексту не зараховується!

[Перевірено](https://docs.google.com/spreadsheets/d/1lv_-aLfkekkrWQPtk8MoTJPQMLCKwFAy083K3bsuEKc/edit#gid%3D0)!

1. Створіть програму (будь-якою зручною для вас мовою), яка в якості вхідних даних приймає текстовий файл, та аналізуючи його вміст:
   1. обраховує частоти (імовірності) появи символів в тексті
   2. обраховує середню ентропію алфавіту для даного тексту
   3. виходячи з ентропії визначає кількість інформації та порівнює її з розмірами файлів
   4. виводить на екран значення частот, ентропії та кількості інформації Приклад роботи створеної [програми](https://github.com/dekom8/Computer-Systems/blob/master/Entropy.cs%20(Sosnov_Lab1_part1)) для обрахунку кількості інформації :

«MAZAFAKA»





1. 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).

За допомогою програми WinRAR та [цього сайту](https://online-converting.com/) файли було стиснено у zip,

rar, gzip, bzip2 та xz.

1. Порівняйте результуючі обсяги архівів з обчисленою кількістю інформації та **наведіть у звіті висновки** щодо кореляції цих величин для обраних вами файлів (яка відмінність, що вийшло більше і чому)

|  |  |  |  |
| --- | --- | --- | --- |
| Файл | REST.txt, Байт | |bpm .txt, Байт | mazafaka.txt, Байт |
| rar | 2430 | 2823 | 638 |
| zip | 2445 | 2829 | 695 |
| gz | 2394 | 2777 | 570 |
| xz | 2284 | 2640 | 656 |
| bz2 | 2073 | 2416 | 604 |
| Оригінал | 7072 | 7455 | 2 104 |
| Кількість інформації | 1772 | 1723 | 411 |

*У всіх випадках найкраще стискав bzip2. Кількість інформації для всіх стиснених файлів менша за їхні фактичні розміри.*

У результаті ідеального стиснення розмір файлу повинен бути рівним кількості інформації. Але у реальності розміри архівованих файлів у більшості випадків дещо більші за кількість інформації.

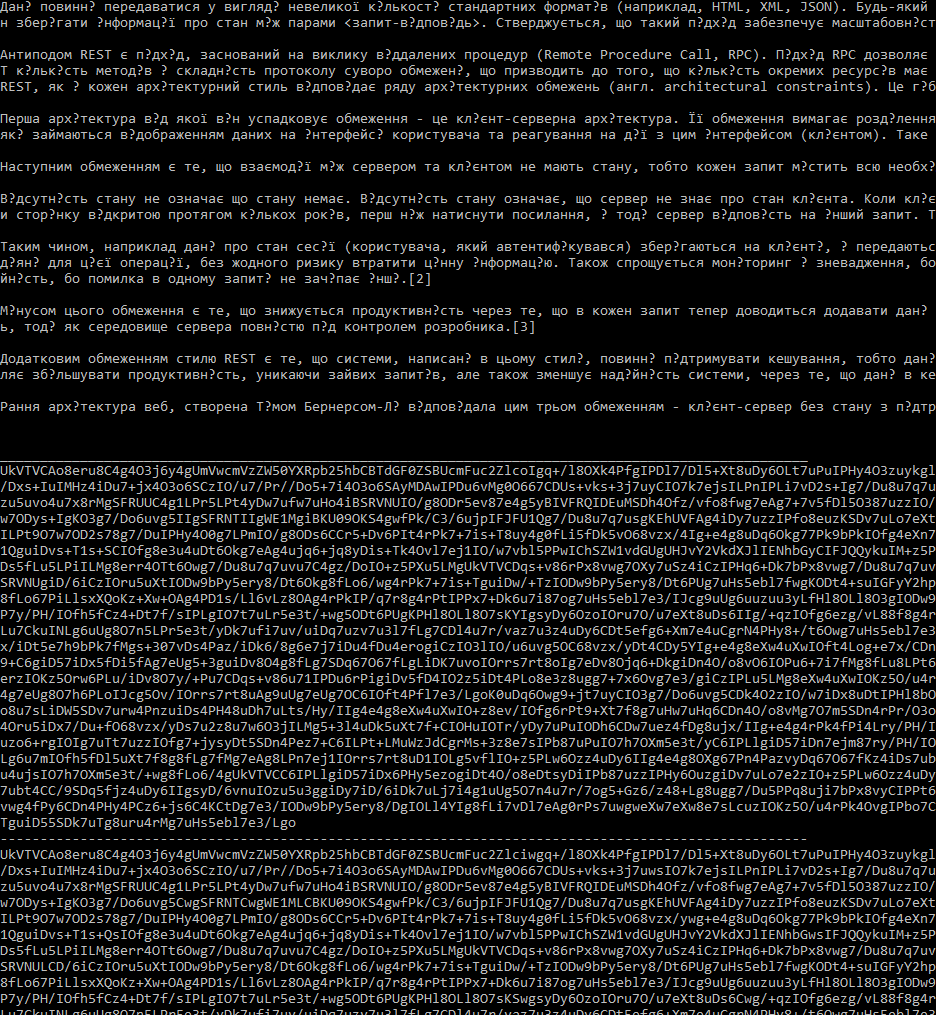
Це відбувається тому, що алгоритми архіваторів побудовані таким чином аби використати повторювані частини тексту. Виходячи з цього, формула розрахунку кількості інформації, використана для програми, не є досконалою, бо вона не враховує передбачення наступного уривку текста.

## Дослідження способів кодування інформації на прикладі Base64

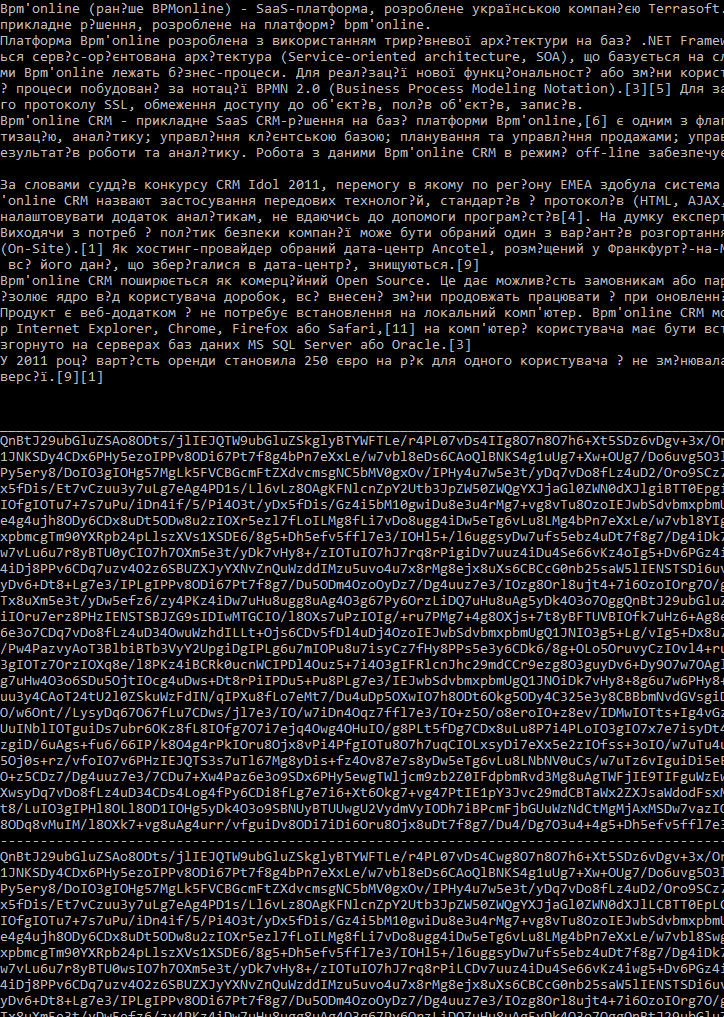
1. Ознайомтесь зі стандартом [RFC4648](https://tools.ietf.org/html/rfc4648)
2. Для практичного засвоєння методу кодування, створіть програму, що кодує довільний файл в Base64 (шляхом реалізації алгоритму вручну, а не виклику бібліотечної функції)

Верхнє кодування - моя програма. Знизу – використовуючи вже існуючу функцію у Visual Studio.

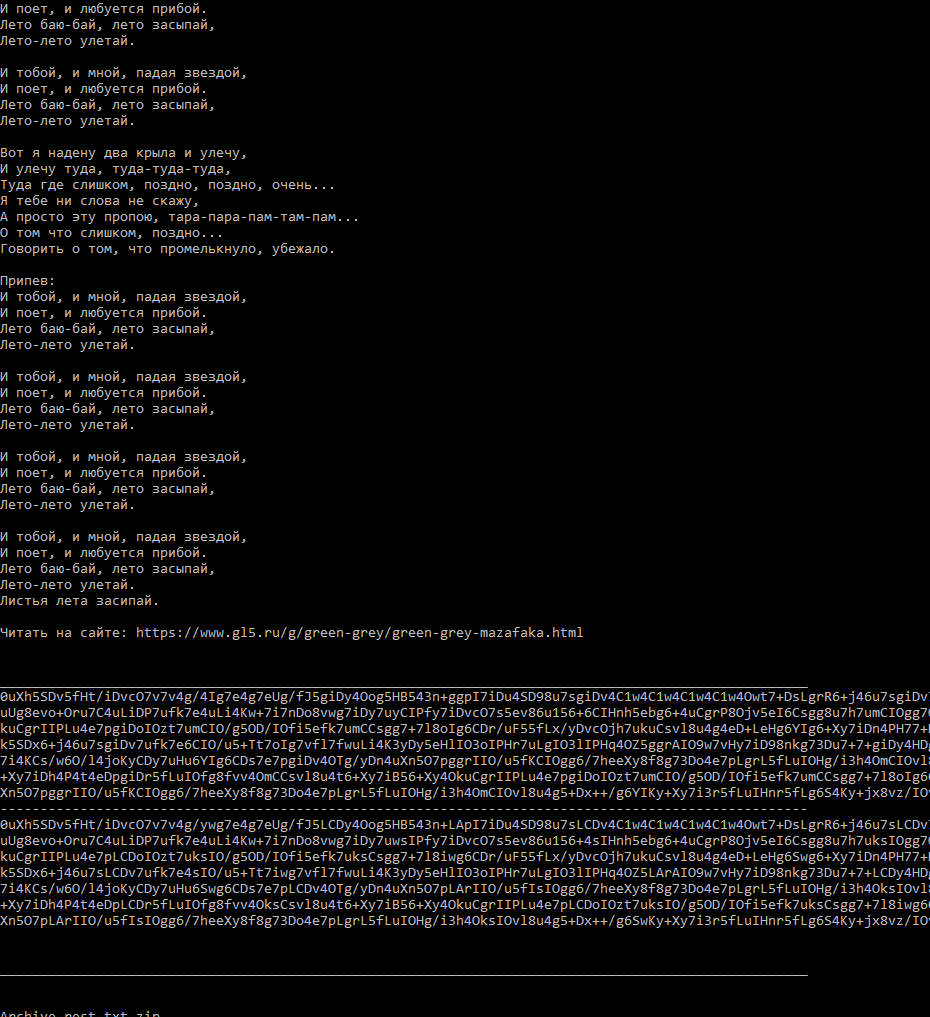
«Rest»



BPM



Mazafaka



1. Закодуйте в Base64 обрані вами текстові файли
   1. Обрахуйте кількість інформації в base64-закодованому варіанті файлу
   2. Порівняйте отримане значення з кількістю інформації вихідного файлу

|  |  |  |
| --- | --- | --- |
| **Файл** | **Кількість інф. у вихідному файлі** | **Кількість інф. у base64** |
| rest.txt | 1772 | 2923.8 |
| bpm.txt | 1723 | 2842.95 |
| mazafaka.txt | 411 | 678.15 |

* 1. Зробіть висновки з отриманого результату

Кількість інформації у Base64 закодованому файлі у середньому у 1.65 рази більше ніж кількість інформації у вихідному файлі.

1. Закодуйте в Base64 стиснені кращим з алгоритмів текстові файли
2. Обрахуйте кількість інформації в base64-закодованому варіанті стисненого файлу
3. Порівняйте отримане значення з кількістю інформації вихідного файлу та base64-

акодованого файлу

|  |  |  |
| --- | --- | --- |
| **Файл** | **Кількість інф. у вихідному файлі** | **Кількість інф. у base64**  **стисненого файлу zip** |
| rest.txt | 2445 | 3251.85 |
| bpm.txt | 2829 | 3762.57 |
| mazafaka.txt | 695 | 924.35 |

1. Зробіть висновки з отриманого результату

Кількість інформації у Base64 стисненого файлу у bz2 у середньому у 1.33 рази більше, ніж кількість інформації у вихідному файлі

**Висновок**

Протягом виконання даної лаб. роботи я дізнався, що таке ентропія інформації та про пов’язані з нею поняття. Теоретичних знань вже майже не залишилися, практичні знання виконав з болем. Також порівняв алгоритми стиснення – з усіх bz2 виявився найкращим. Також працював з алгоритмом кодування Base64, читав про його переваги та недоліки.

[Github](https://github.com/Lagranje/ComputerSystems)