**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](https://github.com/GlebGlushko/CompSys)

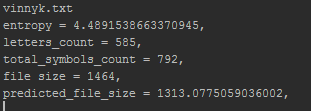
1. Дослідження кількості інформації в тексті

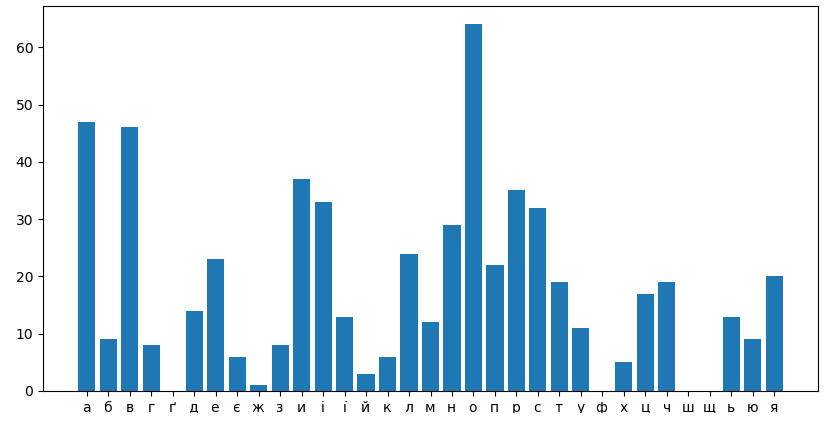
1. Оберіть 3 текстових файла різного тематичного та лінгвістичного спрямування

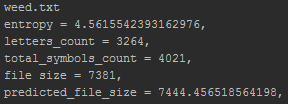
Я вибрав тексти:

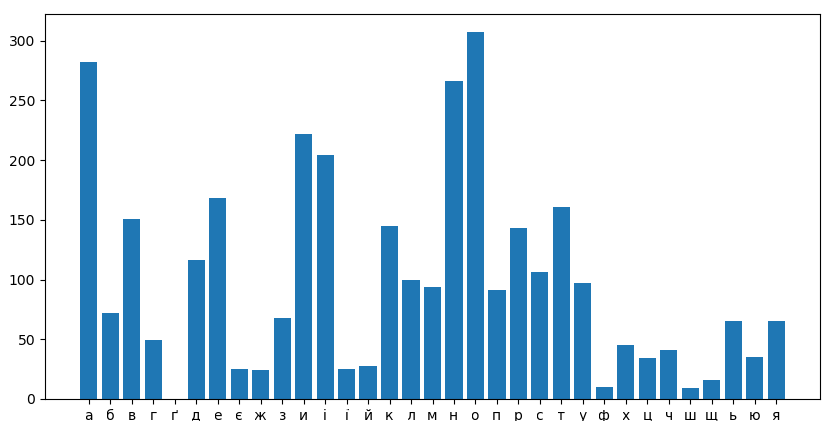
* + 1. Пісня Олега Винника – Вовчиця
    2. Стаття про петицію щодо легазізації медицинського канабісу, та плюси і мінуси канабісу
    3. Докер, основні можливості і процеси

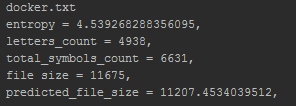
1. Створіть програму (будь-якою зручною для вас мовою), яка в якості вхідних даних приймає текстовий файл, та аналізуючи його вміст:
   1. обраховує частоти (імовірності) появи символів в тексті
   2. обраховує середню ентропію алфавіту для даного тексту
   3. виходячи з ентропії визначає кількість інформації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

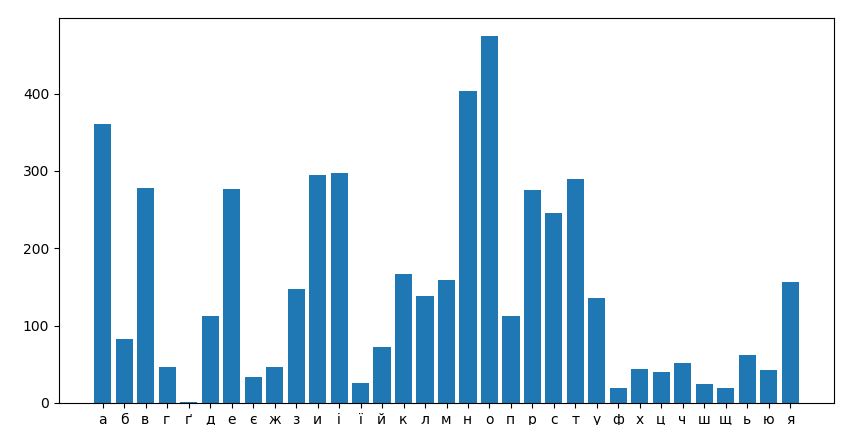




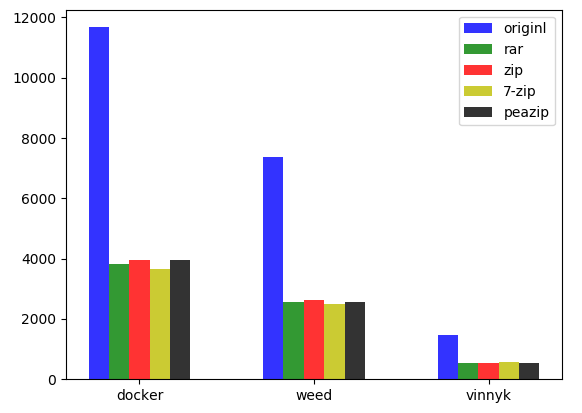


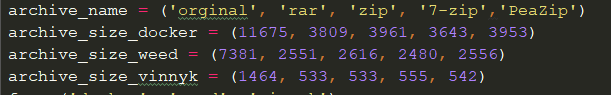






1. Проведіть стиснення кожного вхідного файлу за допомогою 5 різних алгоритмів стиснення (zip, rar, gzip, bzip2, xz, або будь-які інші на ваш вибір, можна використовувати готові програмні засоби для стиснення).

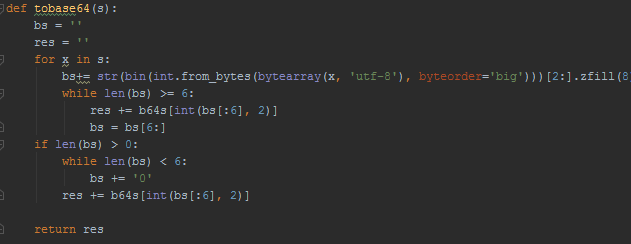




2. Дослідження способів кодування інформації на прикладі Base64

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

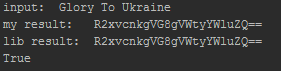
Мій алгоритм

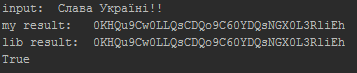


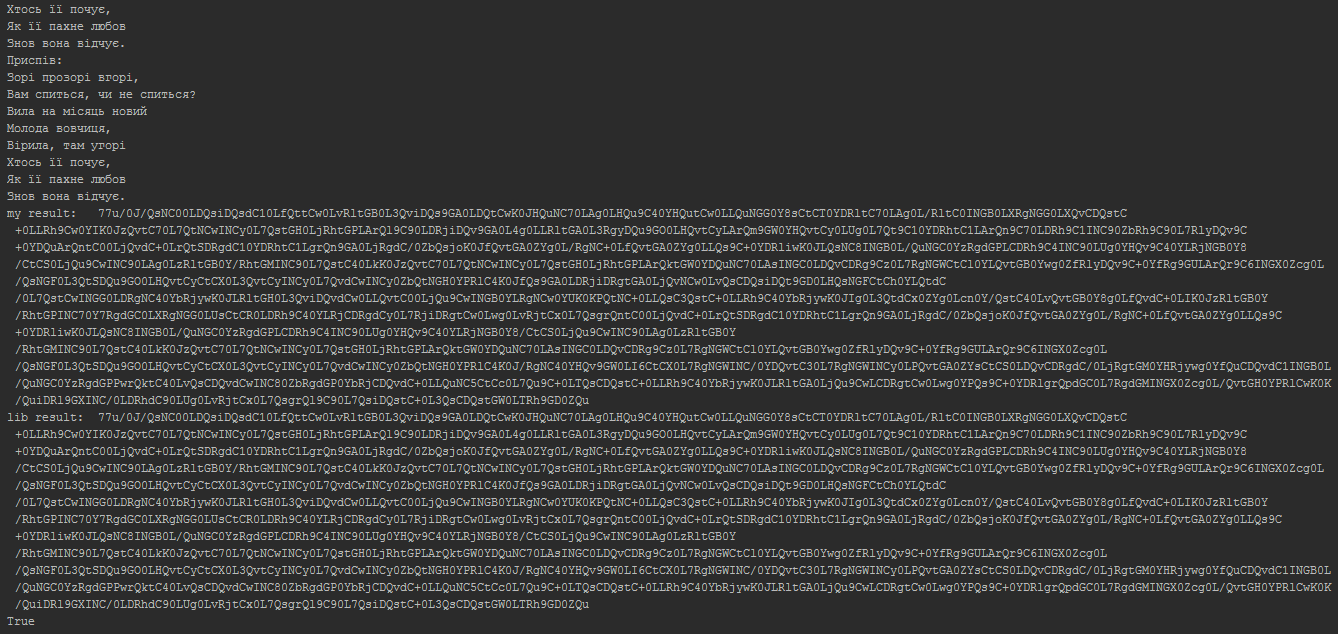
Використання бібліотеки **base64**



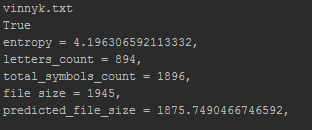
* 1. перевірте коректність роботи програми, порівнявши результат з існуючими програмними засобами

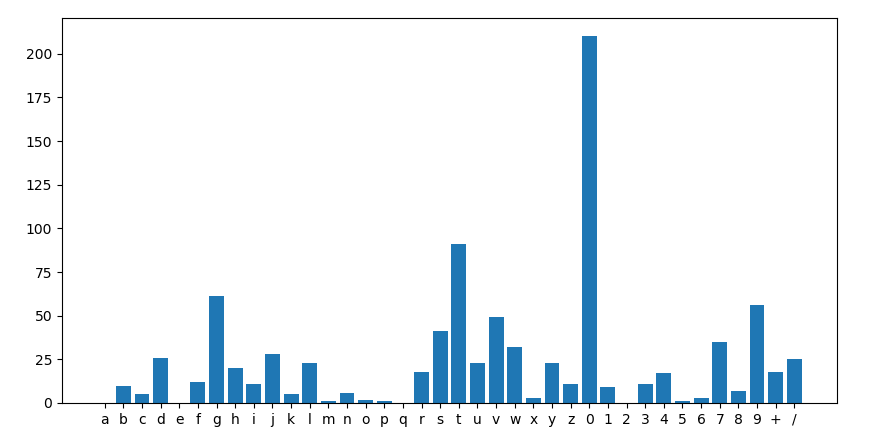


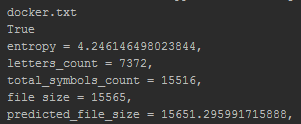


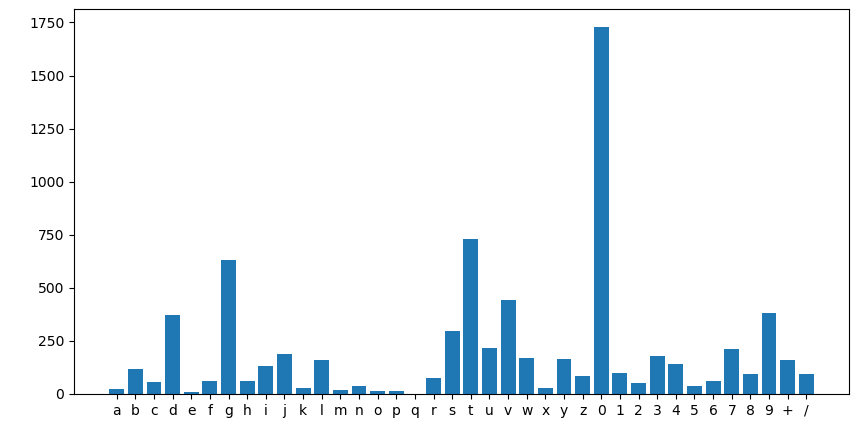


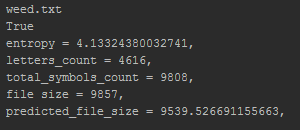
1. Закодуйте в Base64 обрані вами текстові файли

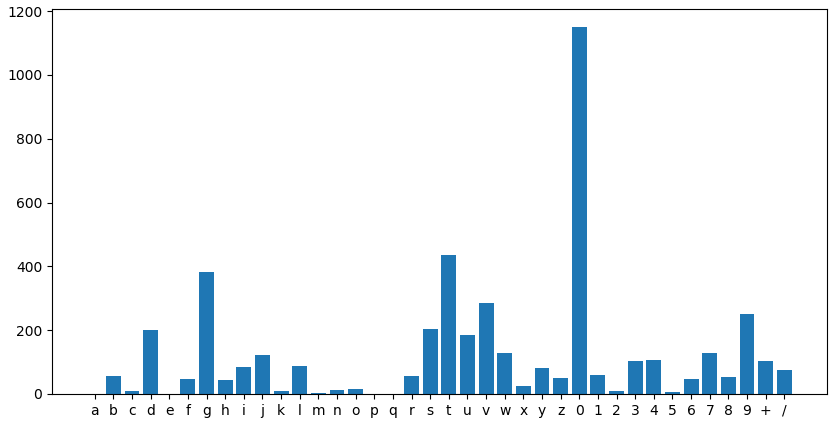












По-перше, я змінив алфавіт на англійський, по-друге, додав до алфавіту символи та цифри, які я використовув у кодуванні в **base64**, тому що, якщо рахувати лише символи, які є літерами, то виходить неправильне значення ентропії і неправильний результат ентропії, тому что текст складається с великої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

Можна побачити, що цифра 0 дуже часто використовується

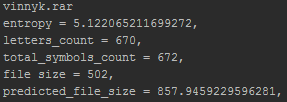
* 1. Порівняйте отримане значення з кількістю інформації вихідного файлу
  2. Зробіть висновки з отриманого результату

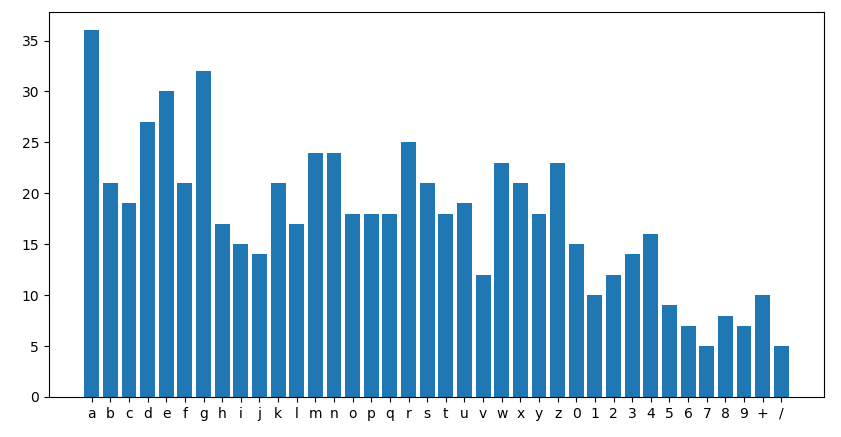
У закодованому вигляді, розміри файлів збільшились у 1.33 рази, тому що тепер для 1 байт замінюєтсья на символи, розміри яких теж 1 байт, але 1символ кодує лише 6 біт.

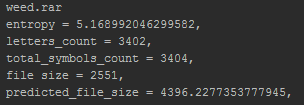
Після кодування, я обрахував ентропію і кількість інформації, і кількість інформації теж пропорційно збільшилась

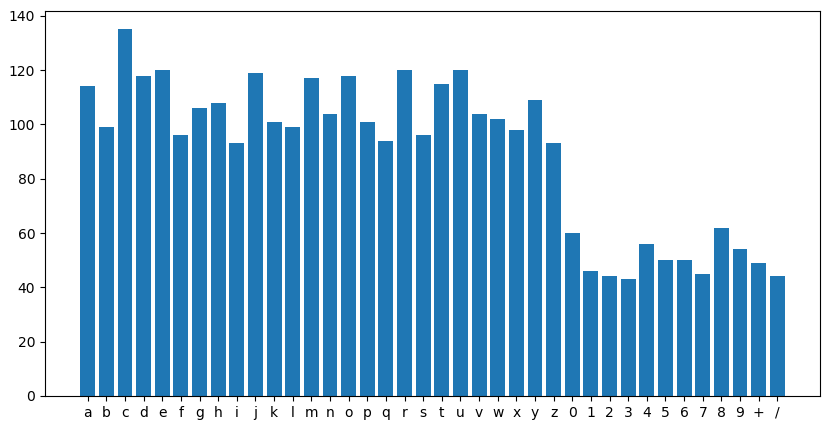
Закодуйте в Base64 стиснені кращим з алгоритмів текстові файли

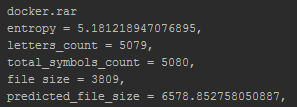
* 1. Обрахуйте кількість інформації в base64-закодованому варіанті стисненого файлу
  2. Порівняйте отримане значення з кількістю інформації вихідного файлу та base64-закодованого файлу

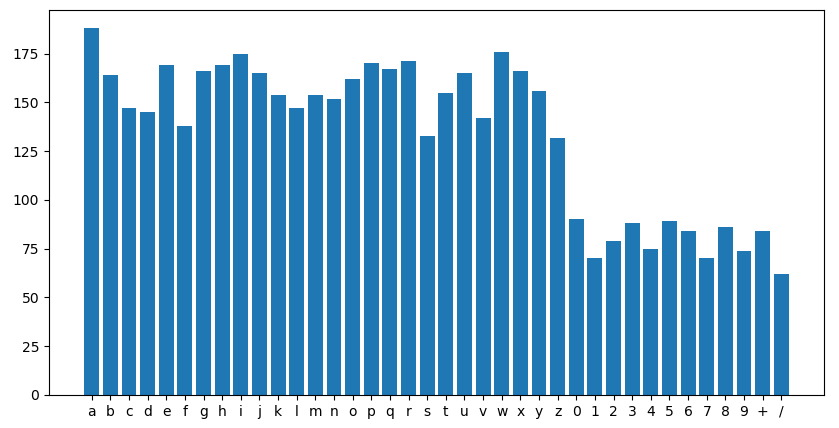












Як можна побачити, що у заархівованому вигляді, у нас більша ентропія і більше спрогнозованої інформації, тому що у заархівованому вигляді, файли намагаються, якнайкраще зберегти інформацію з меншим розміром, отже на один символ(байт) інформації буде більша кількість інформації. Це можна просто пояснити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