

Функціональні вимоги до системи

Система повинна передати зашифроване повідомлення. Основні користувачі системи: Користувач, Адміністратор.

1. Читання текстових файлів та збереження результатів шифрування:
 - 1.1. Система дозволяє Користувачу задати шлях до файла, який він хоче зашифрувати. Пріоритет - високий.
 - 1.2. Система дозволяє Користувачу надати текстовий файл, який представляє зашифроване повідомлення для подальшого розшифрування. Пріоритет - високий.
 - 1.3. Система дозволяє Користувачу ввести текст власноруч для подальшого шифрування. Пріоритет - високий.
 - 1.4. Система виводить на екран зашифроване повідомлення. Пріоритет - високий.
 - 1.5. Система зберігає файл з графічним зображенням в форматі png. Пріоритет - високий.
 - 1.6. Система зберігає результат в файлі формату txt для зображення точок графіків. Пріоритет - високий.
 - 1.7. Система зберігає готовий текст після розшифрування в файлі формату txt. Пріоритет - високий.
2. Шифрування повідомлень:
 - 2.1. Система дозволяє Користувачу отримати зашифроване повідомлення в форматі зображення чи тексту. Пріоритет - високий.
3. Читання зображення з картинки:
 - 3.1. Система дозволяє Користувачу розшифрувати повідомлення з png зображення. Пріоритет - низький.
 - 3.2. Система дозволяє Користувачу розшифрувати повідомлення з текстового файлу точок. Пріоритет - середній.
4. Додавання даних, внесення змін чи їх збереження:
 - 4.1. Система дозволяє Адміністратору обрати нові значення ключів для utf символів. Пріоритет - середній.
 - 4.2. Система дозволяє Адміністратору обрати нові значення ключів для побудови окремих слів та їх взаєморозміщення. Пріоритет - середній.
 - 4.3. Система дозволяє Адміністратору бачити останні повідомлення, які були зашифровані, чи зображення, яка подавали на вході, у порядку їх використання. Пріоритет - середній.
 - 4.4. Система дозволяє Адміністратору бачити та зберігати списки ключів для символів та для розміщення.

Нефункціональні вимоги до системи

1. Вимоги до продуктивності
 - 1.1. Створення зображення і текстового файлу з точками повинно виконуватись не більше 3 секунд.
 - 1.2. Створення текстового файлу з розшифрованим повідомлення має виконуватись не більше 3 секунд.
2. Вимоги до безпеки
 - 2.1. Тільки Адміністратор може міняти ключі шифрування.
 - 2.2. Лише маючи обидва ключі можна розшифрувати повідомлення.
 - 2.3. Тільки Адміністратор може зберігати ключі та шифровані повідомлення.
3. Вимоги до точності
 - 3.1. Розбираючи фотографію зашифрованого зображення, похибка кута, має бути достатньою, щоб розібрати слово
 - 3.2. Максимальна кількість похибок на слово: 10% слова, щоб слово можна було доповнити зі словника.
4. Вимоги до програмного забезпечення
 - 4.1. Система повинна бути розроблена мовою програмування Python.
 - 4.2. Система повинна бути готова до розширення можливостей.
 - 4.3. Система повинна працювати на всіх основних платформах (Windows, Linux, macOS).

Опис даних

Вхідні дані:

Так як дана програма не отримує з інтернету ніякі дані, а лише працює локально, то вона обробляє txt файл. Даний текст може бути довільної форми і містити будь-які utf символи.

Дані, з якими працює програма:

Ключі являють собою два текстові файли:

1. Один з них, є просто рядком-кортежем, який містить двовимірні координати та кут альфа. Можливо, для подальшого ускладнення програми, але не в рамках циклу домашніх завдань, цей файл буде складати з багатьох кортежів, і кожна наступна залежність слів буде вираховуватись окремо.
2. Інший файл містить таблицю, де кожному елементу utf символів відповідає координата. Ця таблиця буде доволі часто змінюватись, щоб розгадати шифр було важче. Поза рамками циклу домашніх робіт буде виконана реалізація іншої програми, яка буде відслідковувати тексти, що будуть

шифруватися, і на основі їх за певним принципом буде змінювати таблицю ключів літер, що буде виглядати майже рандомно.

Вихідні дані:

Вихідні дані можуть мати два різні види:

1. Png зображення, яке містить окремі ламані, кожна з яких представляє одне слово і пов'язане з наступною ламаною.
2. Txt або csv файл, який буде містити кортежі загальних координат на площині. Так як ці координати, мають загальний принцип розміщення і ніяк не пов'язані між собою, то розгадати за ними без ключів нічого неможливо, адже роль грає кут нахилу, завдяки якому одна й та сама точка в просторі може задавати різні символи.

Опис можливостей модулів, бібліотек, що будуть використовуватись

Matplotlib

Дана бібліотека потрібна для зображення доволі складних рисунків в Python. Для даного проекту ця бібліотека необхідна, щоб зображувати слова у вигляді графічних символів-ієрогліфів. Тобто, цей модуль буде найбільше використовуватись в даній програмі.

Функції, класи, методи:

Figure - найвищий клас даної бібліотеки, що визначає весь малюнок в цілому.

Axes - область, в якій відбувається малювання.

Axis - координатна вісь.

Artists - елементи малюнка, що будуть основними структурами в даній програмі.

Всі об'єкти класу Artists діляться на два типи: primitives та containers. Нас більше буде цікавити перший тип, бо він містить такі прості об'єкти, як: лінії, крапки, прямокутники, текст, зображення і тд. Другий тип містить функціонал для роботи з областями, куди вносяться примітиви.

figure() - метод, який створює основну фігуру, де будуть потім будуватись осі та точки, що задають ламані.

show() - метод, що показує малюнки, що були створені.

savefig() - зберігає даний рисунок і зображення необхідного типу. Для того, щоб якість не погіршувалась, для цієї програми буде використано png.

scatter() - точкове подання графіка. Річ, яка буде максимально використовуватись.

plot() - ламана лінія, що також буде часто використовуватись.

Numpy

Ця бібліотека потрібна для роботи з масивами даних. Так, як вона написана ще на фортрані, масиви набагато оптимізованіші, ніж пайтонівські списки. Скоріше за все, цей модуль буде потрібен в ході виконання системи, але поки я не уявляю, як саме і що для цього буде потрібне, тому не візьмуся зараз описувати необхідні інструменти, а зроблю це, коли буду достовірно знати, що саме в конкретній ситуації треба.

OpenCV

Ця бібліотека потрібна для розпізнавання графічних зображень комп'ютером. В даній роботі це необхідно для розшифрування повідомлень, тобто для перетворення графіка в систему точок та потім в текст.

Функції, класи, методи:

imread(path) - основна функція, яка дозволяє відкрити зображення та почати з ним роботу.

findContours() - знаходить контури вибраного об'єкту.