Дискретні структури ІПЗ-21-2

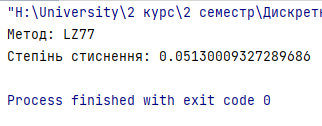
Губарєв Ростислав Вадимович

**Лабораторна робота №9.3**

Створіть програму, яка оцінює стиснення довільного файлу за допомогою словникового алгоритму (LZ77, LZSS, LZ78) із заданими параметрами словника.

|  |
| --- |
| **Код**  def lz77\_compression\_ratio(input\_file, window\_size, lookahead\_buffer):  *# Відкриття вхідного файлу для читання* with open(input\_file, 'rb') as file:  input\_data = file.read()   compressed\_data = []  current\_position = 0   while current\_position < len(input\_data):  *# Шукаємо найдовший префікс у вікні* search\_start = max(0, current\_position - window\_size)  search\_end = current\_position  search\_window = input\_data[search\_start:search\_end]   *# Шукаємо найдовший префікс у майбутньому буфері* match\_length = 0  match\_offset = 0  for i in range(lookahead\_buffer):  current\_window = input\_data[current\_position:current\_position + i]  if current\_window in search\_window and len(current\_window) > match\_length:  match\_length = len(current\_window)  match\_offset = search\_window.index(current\_window)   *# Записуємо таг (зміщення, довжину, наступний символ)* tag = (match\_offset, match\_length, input\_data[current\_position + match\_length:current\_position + match\_length + 1])  compressed\_data.append(tag)   *# Переміщуємо вказівник* current\_position += match\_length + 1   original\_size = len(input\_data)  compressed\_size = len(compressed\_data) \* (2 + 1) *# Розмір тага (зміщення, довжина, наступний символ) у байтах* compression\_ratio = compressed\_size / original\_size   return compression\_ratio   *# Параметри словника* window\_size = 1024 lookahead\_buffer = 64  *# Виклик функції для оцінки стиснення* input\_file = 'input.txt' *# Замініть 'input.txt' на шлях до вашого вхідного файлу* compression\_ratio = lz77\_compression\_ratio(input\_file, window\_size, lookahead\_buffer)  *# Виведення оцінки стиснення* print("Метод: LZ77") print(f"Степінь стиснення: {compression\_ratio}") |

**Результат**

****