

Міністерство освіти і науки України
Національний технічний університет України «КПІ ім. Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії
Програмування інтелектуальних інформаційних систем

ЗВІТ
до лабораторних робіт

Виконав
студент

Піць Денис Олександрович
(№ групи, прізвище, ім'я, по батькові)

Прийняв

ас. Очеретяний О. К.
(посада, прізвище, ім'я, по батькові)

Київ 2021

1. Завдання лабораторної роботи

Завдання 1:

Обчислювальна задача тут тривіальна: для текстового файлу ми хочемо відобразити N (наприклад, 25) найчастіших слів і відповідну частоту їх повторення, упорядковано за зменшенням. Слід обов'язково нормалізувати використання великих літер і ігнорувати стоп-слова, як «the», «for» тощо. Щоб все було просто, ми не піклуємося про порядок слів з однаковою частотою повторень. Ця обчислювальна задача відома як *term frequency*.

Завдання 2:

Тепер, нам потрібно виконати задачу, що називається словниковим індексуванням. Для текстового файлу виведіть усі слова в алфавітному порядку разом із номерами сторінок, на яких Ці слова знаходяться. Ігноруйте всі слова, які зустрічаються більше 100 разів. Припустимо, що сторінка являє собою послідовність із 45 рядків.

2. Опис програмного коду

Завдання 1:

Виконую спліт тексту шляхом двох «циклів». Перший шукає початок слова, другий – його кінець. Таким чином отримую масив із слів без будь-яких символів, крім латинського алфавіту. Відразу змінюю першу букву на маленьку. Створюю два масиви, які утворюють пари унікальних слів та кількістю їх повторів за однаковими індексами. Прохожу «циклом» по словам, отриманим з файлу. Якщо серед унікальних слів немає слова з файлу, то додаю його в цей масив, якщо кількість символів більша за три. Якщо таке є, то збільшую кількість повторів слова на 1. Так роблю зі всіма словами з файлу. На виході отримую масив із заповненими ідентичними словами та масив з кількістю повторень цих слів. Після цього виконую сортування масивів за зростанням бульбашковим сортуванням, сортуючи відразу два масиви за порівнянням слова,

використовуючи для цього ще один «цикл». В кінці виконання вивожу в зворотному порядку слова та кількість їх повторення в файлі.

Завдання 2:

Зчитую з файлу текст. Розбиваю його на сторінки, де висота сторінки – lineHeight (кількість переносів рядків). Сторінки утворюють масив типу string із усім їх текстом. Прохожу «циклом» по тексту сторінок. Виконую спліт тексту шляхом двох «циклів». Перший шукає початок слова, другий – його кінець. Таким чином отримую масив із слів без будь-яких символів, крім латинського алфавіту. Створюю два масиви, які утворюють пари унікальних слів та номерами сторінок, на яких вони зустрічаються, де номер сторінки дорівнює їх індексу + 1. Прохожу «циклом» по словам сторінки. Якщо серед унікальних слів немає слова із сторінки, то додаю його в цей унікальний масив, відразу додаючи відповідний номер сторінки до другого масиву. Якщо таке слово є, то просто додаю до сторінок, на який це слово зустрічається, дану сторінку. Так роблю зі всіма словами з файлу. На виході отримую масив із заповненими ідентичними словами та масив із масивів номерів сторінок, на яких це слово зустрічається. Після цього виконую сортування масивів за зростанням бульбашковим сортуванням, сортуючи відразу два масиви за порівнянням слова, використовуючи для цього ще один «цикл». В кінці виконання вивожу слова та номери сторінок, на яких вони з'являються, якщо їх кількість не перевищує 100.

3. Скріншоти роботи програмного застосунку

```
Microsoft Visual Studio Debug Console
amet - 6224
eget - 5068
nunc - 4648
vitae - 4565
enim - 4075
egestas - 3780
pellentesque - 3706
diam - 3615
viverra - 3555
quis - 3532
arcu - 3392
massa - 3284
tellus - 3282
nulla - 3257
mauris - 3223
aliquam - 3181
tincidunt - 3102
morbi - 3077
risus - 3068
faucibus - 2980
neque - 2801
tortor - 2777
nibh - 2757
urna - 2706
purus - 2696

C:\Users\lykos\source\repos\MPP1\MPP1\bin\Debug\net6.0\MPP1.exe (process 16972) exited with code 0.
Press any key to close this window . . .
```

Рис 3.1 – результат виконання першого завдання

```
Microsoft Visual Studio Debug Console
Accumsan - 1, 2, 3, 5, 6, 7, 8, 9, 10, 12, 13, 16, 17, 19, 20, 21, 23, 24, 26, 28, 29, 31, 32, 34, 35, 37, 40, 41, 43, 44, 46, 47, 48,
49, 50, 51, 52, 55, 57, 58, 59, 60, 61, 63, 66, 67, 68, 69, 71, 72, 73, 75, 76, 77, 78, 80, 82, 83, 87, 88, 89, 90, 91, 93, 94, 95, 97,
98, 99, 100, 101, 104, 105, 106, 107, 108, 109, 110, 111, 113, 114, 115, 117, 118, 119, 120, 121, 122, 123, 125, 126
Aenean - 2, 3, 4, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 20, 22, 25, 26, 27, 28, 29, 30, 31, 32, 34, 35, 37, 38, 39, 41, 42, 43, 45, 47,
48, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 67, 68, 71, 72, 74, 76, 77, 78, 79, 80, 82, 83, 84, 85, 86, 88, 89,
90, 91, 92, 93, 94, 95, 96, 97, 98, 101, 102, 103, 105, 107, 108, 110, 111, 112, 113, 114, 115, 117, 118, 119, 121, 124, 126
Ante - 1, 3, 6, 11, 13, 15, 18, 24, 27, 30, 32, 35, 36, 39, 41, 42, 45, 48, 52, 59, 63, 64, 65, 67, 68, 69, 71, 73, 75, 77, 78, 79, 81,
84, 86, 88, 89, 91, 92, 93, 94, 95, 96, 97, 100, 101, 106, 109, 110, 111, 114, 124, 125, 126
Auctor - 2, 6, 7, 8, 9, 11, 12, 13, 14, 15, 16, 18, 19, 20, 21, 22, 23, 25, 27, 29, 30, 31, 32, 33, 34, 35, 37, 38, 39, 42, 43, 44, 46,
47, 48, 50, 51, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 65, 66, 67, 68, 69, 70, 72, 73, 75, 77, 78, 79, 80, 81, 84, 86, 87, 88, 91,
92, 94, 95, 97, 98, 99, 101, 102, 103, 105, 106, 108, 109, 110, 111, 112, 113, 115, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126
Condimentum - 2, 3, 5, 7, 8, 9, 10, 12, 14, 15, 16, 17, 19, 20, 23, 24, 28, 29, 30, 31, 33, 35, 36, 37, 38, 39, 42, 43, 44, 45, 46, 47,
48, 49, 50, 52, 55, 56, 57, 59, 61, 62, 63, 64, 66, 68, 69, 70, 73, 74, 76, 77, 81, 82, 85, 88, 89, 90, 91, 92, 93, 94, 95, 96, 101, 1
04, 105, 106, 107, 108, 109, 110, 111, 113, 114, 115, 117, 120, 122, 123, 124, 125
Congue - 4, 5, 7, 8, 9, 11, 13, 17, 19, 20, 21, 23, 24, 25, 26, 27, 28, 29, 30, 32, 33, 35, 38, 39, 40, 41, 42, 43, 44, 46, 47, 49, 50,
52, 55, 56, 57, 58, 60, 62, 65, 66, 68, 69, 70, 71, 73, 74, 75, 76, 79, 80, 82, 83, 84, 85, 86, 87, 88, 89, 93, 94, 95, 96, 98, 100, 1
01, 103, 105, 106, 107, 109, 111, 112, 113, 116, 117, 118, 119, 120, 121, 122, 123, 124, 126
Cum - 1, 5, 6, 10, 13, 16, 21, 27, 37, 41, 43, 45, 46, 47, 52, 56, 70, 71, 78, 79, 82, 84, 89, 94, 95, 98, 102, 114, 116, 118, 124
Curabitur - 3, 5, 7, 13, 16, 17, 18, 19, 21, 22, 30, 32, 33, 34, 35, 36, 37, 39, 43, 46, 47, 50, 51, 52, 53, 54, 60, 63, 69, 73, 75, 77
, 79, 85, 86, 89, 91, 93, 95, 96, 98, 99, 102, 104, 106, 107, 109, 110, 114, 115, 117, 121, 125, 126
Dapibus - 7, 8, 11, 12, 15, 18, 21, 24, 25, 27, 35, 38, 41, 45, 53, 54, 55, 57, 64, 67, 69, 83, 87, 91, 94, 96, 99, 106, 110, 112, 114,
121
Dictum - 1, 2, 3, 4, 6, 8, 9, 10, 11, 12, 13, 14, 16, 17, 18, 19, 20, 22, 24, 25, 26, 27, 28, 29, 30, 31, 32, 34, 35, 36, 39, 40, 41, 4
3, 44, 45, 47, 48, 49, 51, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 65, 66, 67, 69, 70, 71, 72, 75, 76, 77, 78, 79, 80, 81, 84, 85,
86, 87, 88, 89, 90, 91, 94, 95, 96, 99, 100, 102, 103, 105, 106, 107, 108, 110, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124,
125
Dictumst - 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 16, 17, 18, 20, 27, 30, 31, 32, 35, 39, 40, 41, 42, 44, 49, 50, 51, 55, 56, 57, 58, 59,
64, 65, 72, 73, 74, 76, 79, 81, 83, 84, 85, 89, 94, 95, 97, 99, 100, 103, 107, 109, 113, 116, 123
Dis - 2, 19, 26, 34, 35, 39, 43, 44, 45, 51, 55, 58, 59, 66, 67, 74, 76, 78, 94, 100, 111, 112, 113, 115, 121, 126
Eleifend - 1, 3, 4, 5, 7, 12, 16, 19, 26, 27, 28, 29, 31, 32, 33, 36, 37, 38, 39, 41, 44, 45, 47, 53, 54, 55, 57, 58, 59, 60, 61, 62, 6
5, 66, 67, 68, 69, 71, 72, 73, 75, 79, 84, 85, 86, 88, 90, 92, 93, 94, 96, 98, 99, 100, 101, 102, 104, 105, 106, 108, 109, 110, 111, 11
4, 115, 116, 118, 119, 120, 121, 122, 123, 124, 125
Erat - 2, 3, 4, 5, 6, 7, 8, 10, 12, 13, 14, 17, 18, 19, 20, 23, 25, 26, 27, 28, 29, 31, 32, 33, 34, 35, 36, 37, 38, 40, 46, 47, 50, 51,
```

Рис. 3.2 – результат виконання другого завдання