**Міністерство освіти і науки України**

**Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського"**

**Факультет інформатики та обчислювальної техніки**

**Кафедра автоматизованих систем обробки інформації**

**і управління**

**Звіт**

з лабораторної роботи № 4 з дисципліни

«Алгоритми та структури даних 2. Структури даних»

**«Спискові структури даних»**

**Виконав(ла)**

(шифр, прізвище, ім'я, по батькові)

*ІП-01 Галько Міла Вячеславівна*

**Перевірив**

(прізвище, ім'я, по батькові)

*Головченко М.М.*

Київ 2021

**Лабораторна робота № 4**

**«Спискові структури даних»**

**Мета роботи** – отримати навички роботи із списковими структурами даних для вирішення типових задач з їх допомогою.

**Завдання**

Розробити алгоритм розв’язання задачі відповідно до варіанту №6. Виконати програмну реалізацію задачі. Не використовувати вбудовані спискові структури даних (контейнери). Зробити висновок по лабораторній роботі.

Варіант №6: Заданий рядок слів, які відокремлюються одне від одного комами. Побудувати список, елементами якого є відповідні слова. Вилучити із списку всі слова заданої користувачем довжини. Надрукувати початковий і змінений списки.

1. Псевдокод:

|  |  |
| --- | --- |
| **Змінна** | **Значення** |
| str | Рядок зі словами відокремлені комами |
| length | Довжина слова (задається користувачем) |
| word | Зчитане слово з рядка str |

MAIN

1. ПОЧАТОК
2. Обрати str
3. ПОКИ !str.empty():
   1. ПОКИ символ str літера:
      1. Перейти до наступного символу str
   2. ЯКЩО поточний символ нульовий, ТО очистити цей символ з str. ІНАКШЕ:
      1. Зчитати слово до поточного символу
      2. Додати слово у List методом Push(word)
      3. Очистити str до поточного символу
4. Надрукувати вміст List
5. Зчитати length
6. Виконати метод List Search\_and\_Delete(length)
7. Вивести кінцевий вміст List
8. КІНЕЦЬ

|  |  |
| --- | --- |
| **Змінна** | **Значення** |
| head/tail | Голова/хвіст List |
| next | Покажчик на наступний елемент List |
| Node | Структура у List, що відповідає за зв’язки між елементами (за доп. покажчиків) |
| current | Елемент List, що буде видалений |
| exHead/exTail | Голова/хвіст, що видаляються |
| newTail | Елемент List, що передує exTail |
| prevItem | Елемент List, що передує current |

**Таблиця змінних методів List**

МЕТОД LIST Push(word)

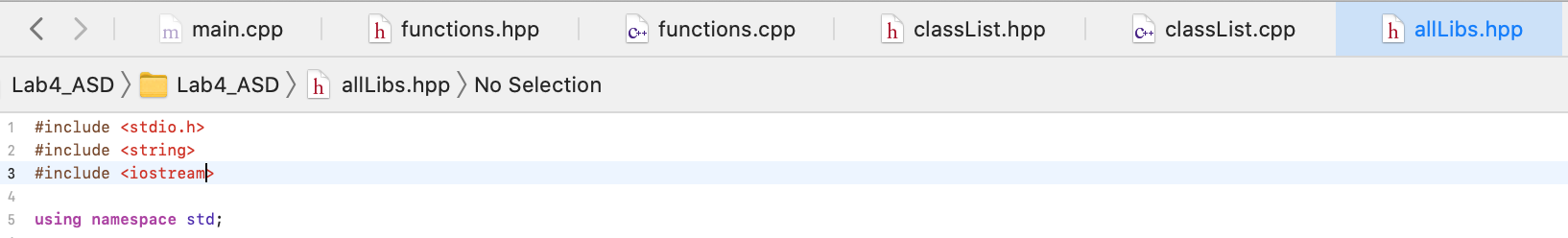
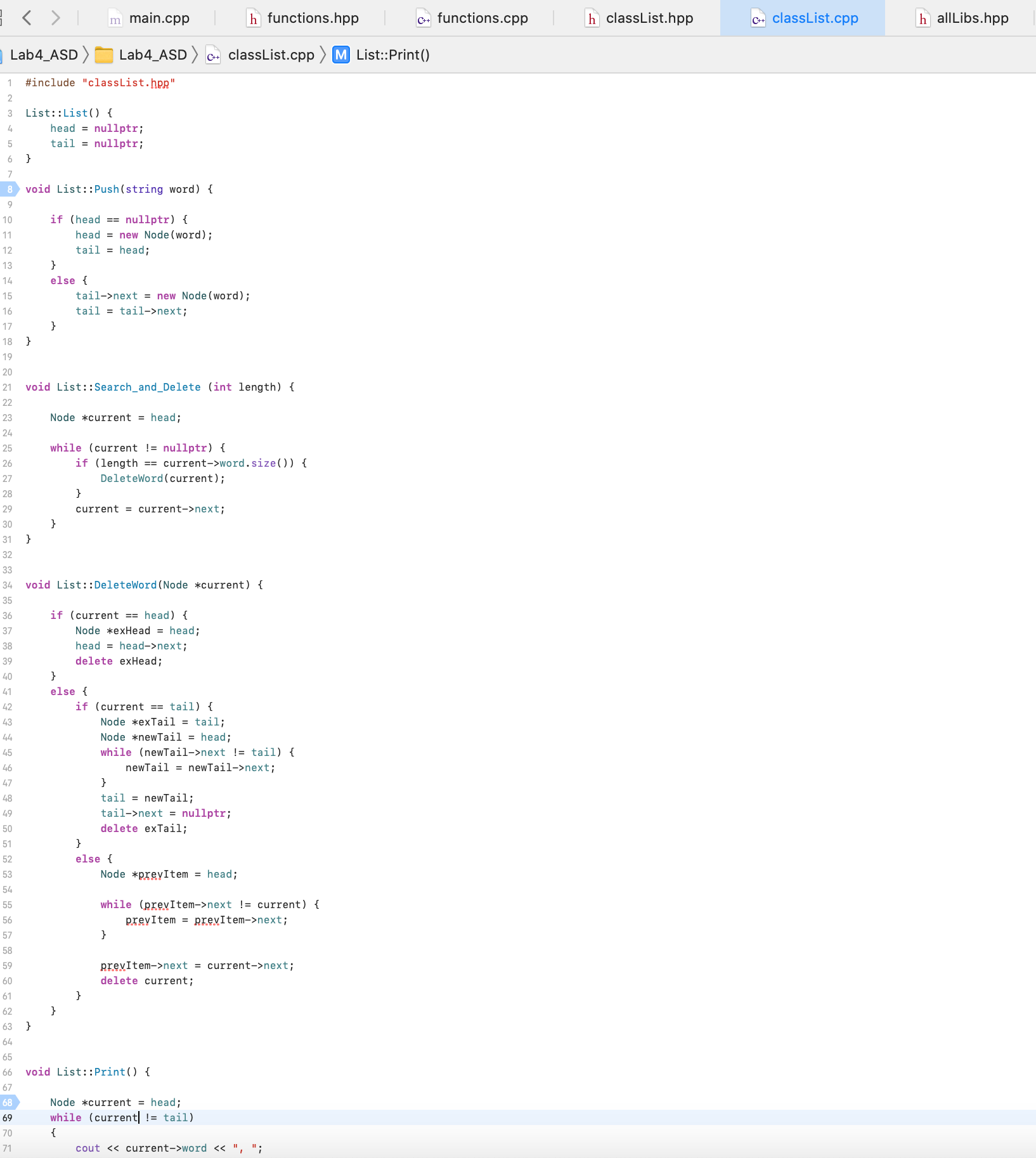
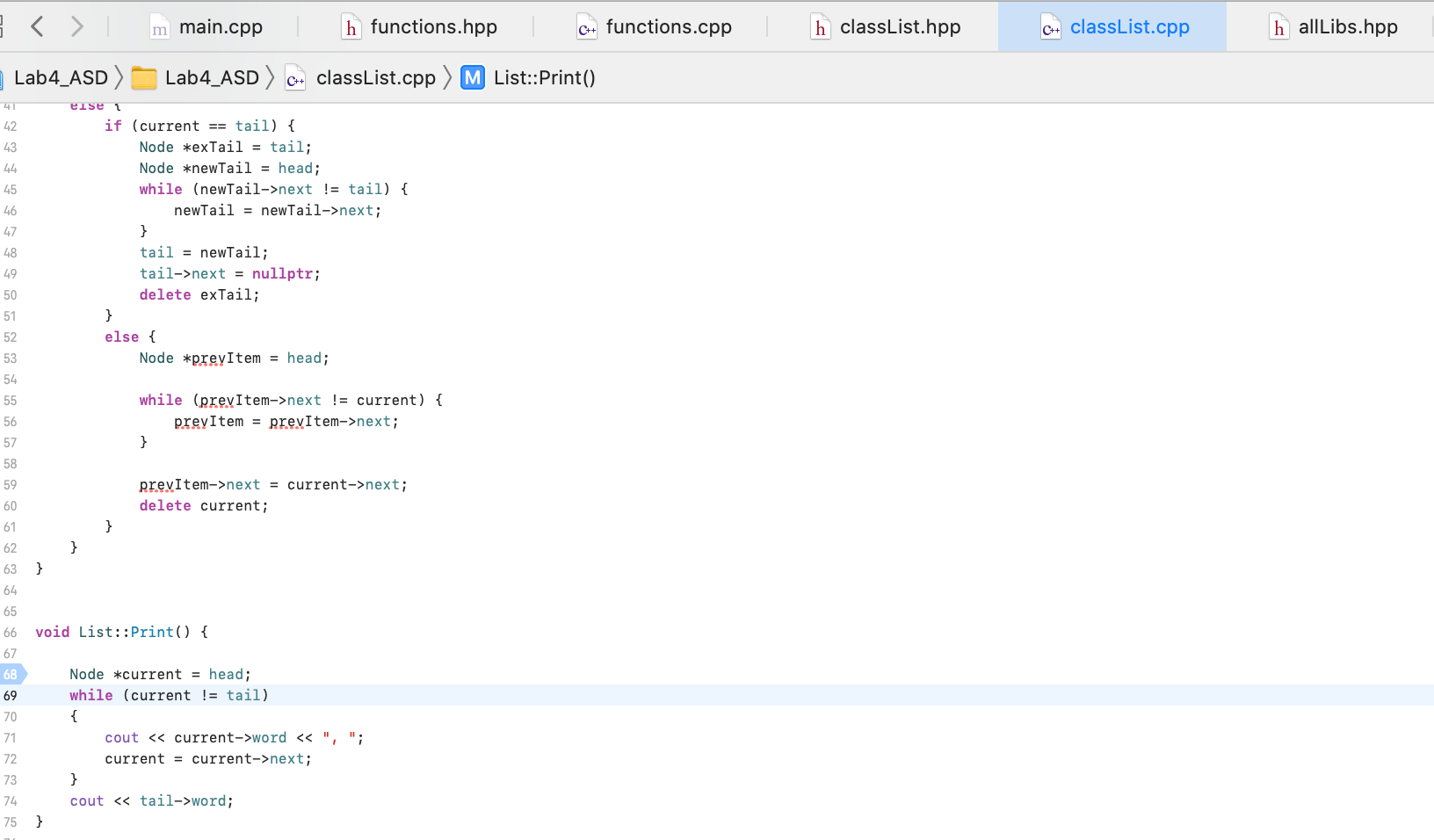
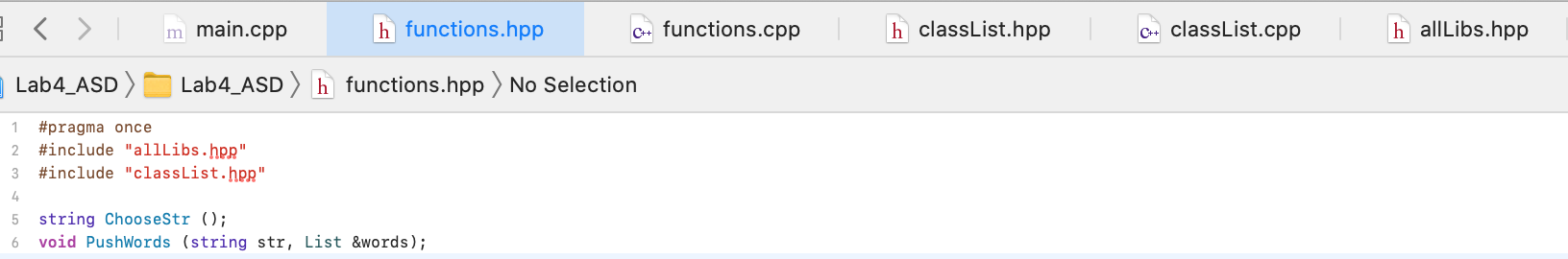
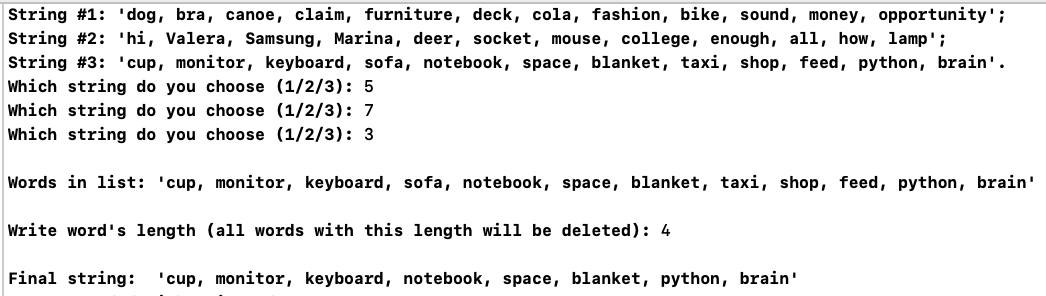
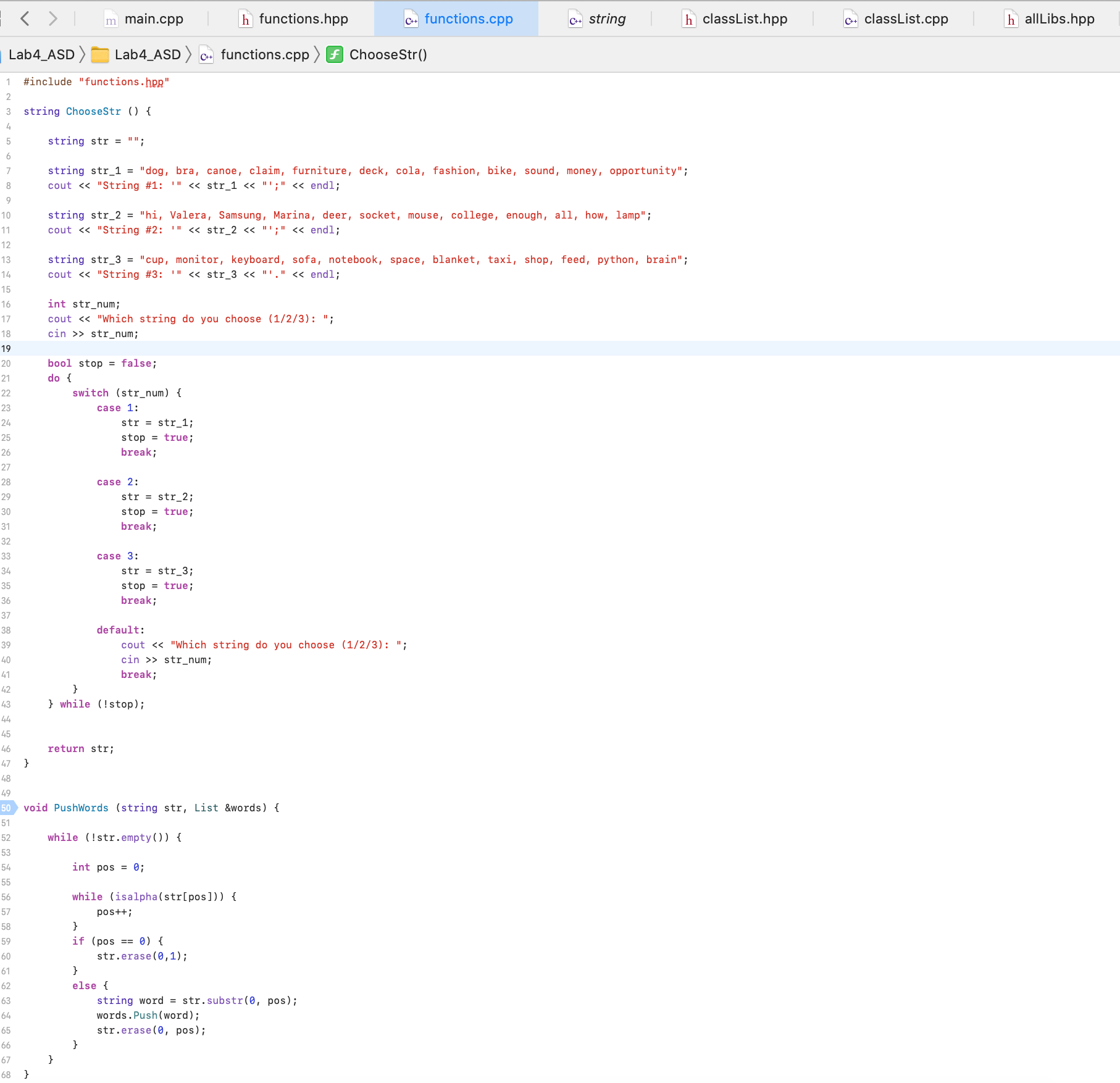
1. ПОЧАТОК
2. ЯКЩО head немає, ТО:
   1. head = new Node(word)
   2. tail = head
3. ІНАКШЕ:
   1. tail->next = new Node(word)
   2. tail = tail->next
4. КІНЕЦЬ

МЕТОД LIST Search\_and\_Delete(length)

1. ПОЧАТОК
2. ПОКИ не дійдемо до кінця List:
   1. ЯКЩО length == довжині слова current List, ТО виконати метод List DeleteWord(current)
   2. Перейти до наступного елемента List
3. КІНЕЦЬ

МЕТОД LIST Delete(current)

1. ПОЧАТОК
2. ЯКЩО current == head, ТО:
   1. Node \*exHead = head
   2. head = head->next
   3. delete exHead
3. ІНАКШЕ:
   1. ЯКЩО current == tail, ТО:
      1. Node \*exTail = tail
      2. Node \*newTail = head
      3. ПОКИ newTail->next != tail:
         1. newTail = newTail->next
      4. tail = newTail
      5. tail->next = nullptr
      6. delete exTail
   2. ІНАКШЕ:
      1. Node \*prevItem = head
      2. ПОКИ prevItem ->next != current:
         1. prevItem = prevItem ->next
      3. prevItem->next = current->next
      4. delete current
4. КІНЕЦЬ
5. С++:

3) Висновок:

У даній лабораторній роботі я отримала навички у використанні спискових структур. Для вирішення завдання був використаний однозв’язний список List. Додавання елементів у якому виконується за О(1), пошук за O(n) та видалення елемента за О(n).

Оскільки за задачею потрібно продивитися усі слова (для видалення елементів), то в сумі пошук із видаленням виконується за O(n2).

Підсумовуючи, під час виконання лабораторній роботі були розроблені:

1. Алгоритм виконання задачі (Варіант №6);

2. Програма, що спроектована на основі цього алгоритму.

Щодо програми:

1. Рядок обирається 1 з 3-х запропонованих ;

2. Виводяться: запропоновані рядки; вміст List початковий; вміст List кінцевий.