Одеський національний політехнічний університет

Інститут комп’ютерних систем

Кафедра системного програмного забезпечення

Лабораторна робота №3.2

з дисципліни «Проектний практикум»

Виконала:

ст.гр. АС-142

Маковій Т.Ю.

Перевірив:

доц. Блажко О. А.

Одеса – 2018 рік

## **Оцінка складності алгоритмів**

На етапі проектування було побудовано алгоритм формування переліку лексем з програмного коду, а також цей алгоритм був оцінений з огляду на алгоритмічну складність. Схема алгоритму надана на рис. 1.

Розглянемо детальніше блоки представленого алгоритму, та опишемо складність кожного з них.

Блок ініціалізації має оцінку складності рівну О(4), бо впродовж роботи алгоритму кількість змінних, що потребують ініціалізації, постійно і буде дорівнювати на етапі реалізації не більше ніж 4. Тому можна вважати, що складність цього блоку є константною і дорівнює О(1). Аналогічну оцінку можна дати іншим блокам, що відображають процеси, а також блокам, що додають лексему до переліку лексем, блокам, відповідним за прирівнювання.

Найскладнішими блоками алгоритму є блоки циклу while , що відповідають за повне проходженні коду, по реченню та по символу. Складність цих блоків залежить від кількості речень в коді, від кількості символів у кожному реченні, тому вона може бути оцінена як О(n) для while для речень, О(n^2) для while для символів, що означає лінійну залежність - чим більше розмір списку сутностей, тим довше буде працювати алгоритм.

Також складними вважаються блоки, що відповідають іншим функціям модулю. Так блок «Перевірка на приналежність tempLexem до keyWords» перебирає та порівнює значення переліку keyWords та tempLexem і його складність може змінюватись в залежності від кількості ключових слів, місця відповідного ключового слова у переліку. Максимальна складність - О(n).

Умовні блоки не мають залежності часу виконання від розміру коду, що надходить для перетворення, тому їх складність оцінюється як константа.

Зважаючи на те, що за складність загального алгоритму приймається найбільше значення з виявлених складностей компонентів алгоритму, розглянутий алгоритм формування переліку лексем з програмного коду має складність О(n^2).

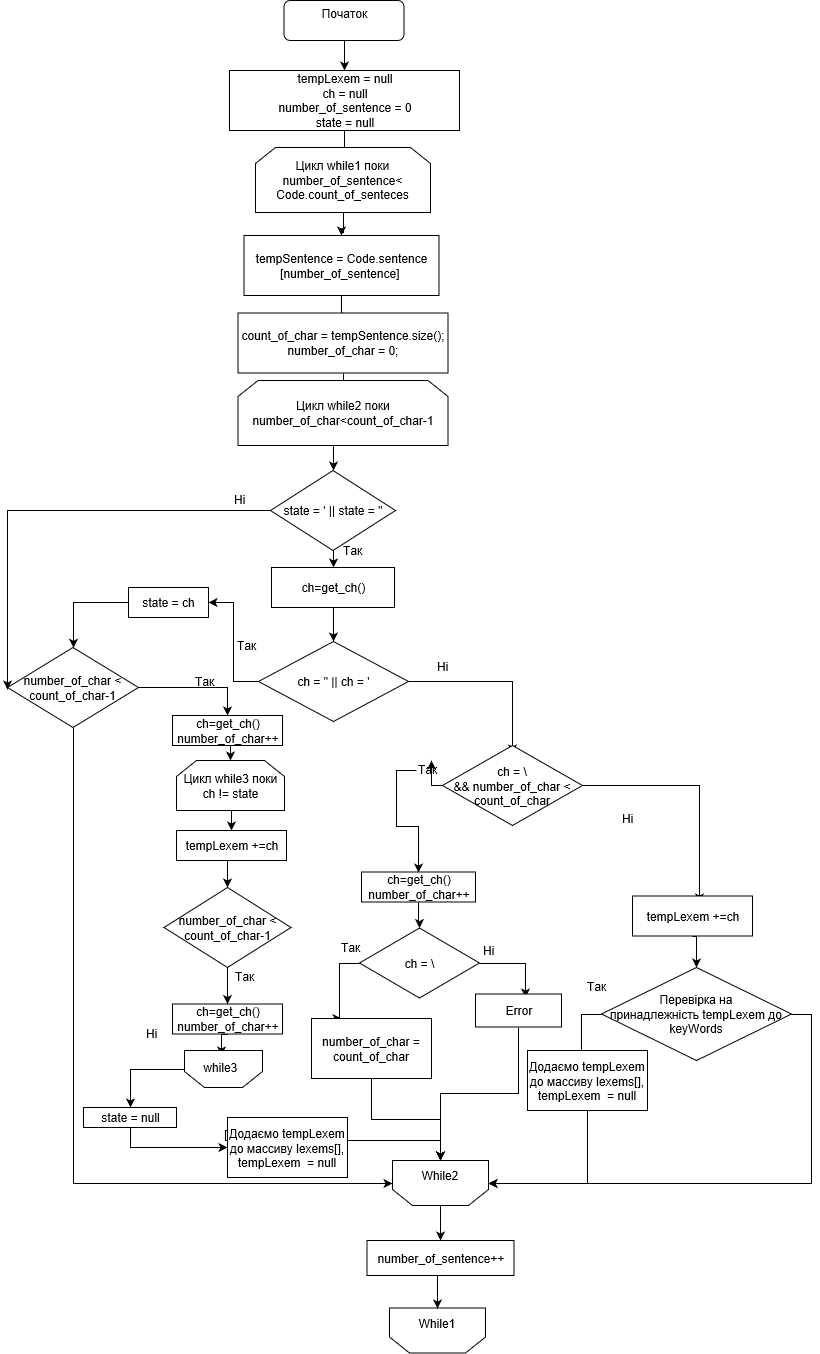


Рисунок 1 - Схема алгоритму розбору вхідного программного коду

## **Проектування інтерфейсу користувача**

На цьому етапі було створено макети форм графічного інтерфейсу користувача.

Майже весь простір вікна мають займати поля введення даних, бо майже вся взаємодія націлена на отримання інформації, але й значну частку займає поле виведення, що функціонує, якщо користувач обирає перетворення у інший код, а не у граф. Функціонал програми міститься в кнопці «Преобразовать», що отримує данні, з заповнених полів, та використовує їх для розв’язку задачі. Поля обирання шляху до файлу з кодом та вставлення коду напряму є взаємовиключними і активуються за допомогою radio button, також взаємовиключними є поля обирання вихідного коду та графу. Якщо обрано поле «Граф», то поле, де повинен виводитись вихідний код - дезактивовано. Поля вибору вхідного і вихідного коду виконані за допомогою ComboBox, що дозволяє обрати один з наведених варіантів. Підсумовуючи все вище зазначене представимо макет графічного інтерфейсу на рис. 2.

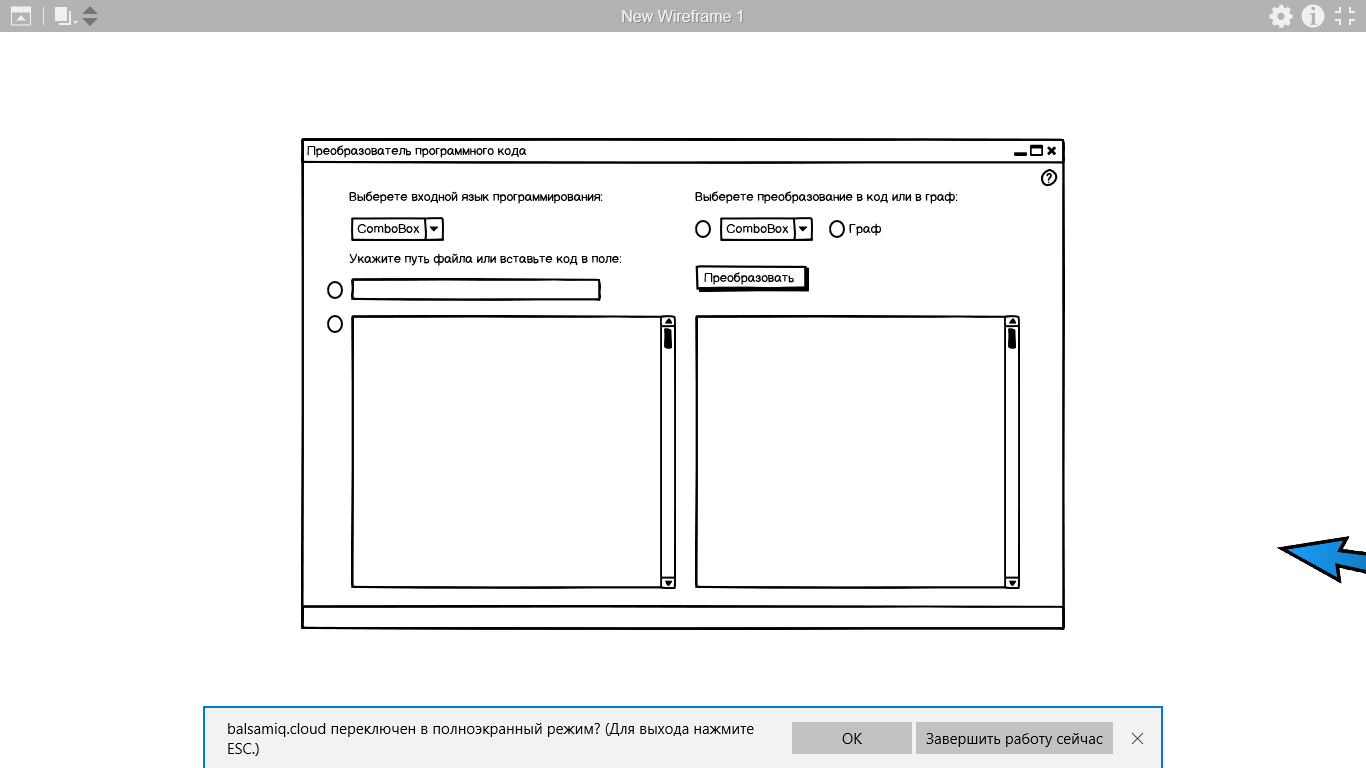


Рисунок 2 - Макет головного вікна програми