МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ІВАНА ФРАНКА

Паралельні та розподілені обчислення

ЛАБОРАТОРНА РОБОТА №7

«Алгоритм Прима»

Виконала:

студентка групи ПМі-31

Дудчак Валентина Юріївна

Львів 2024

**Тема**: Розпаралелення алгоритму Прима

**Мета**: Написати програми розв’язування алготму Прима (послідовний та паралельний алгоритми). Для зваженого зв'язного неорієнтованого графа G(V,F), використовуючи алгоритм Прима, з довільно заданої вершини **а** побудувати мінімальне кісткове дерево.

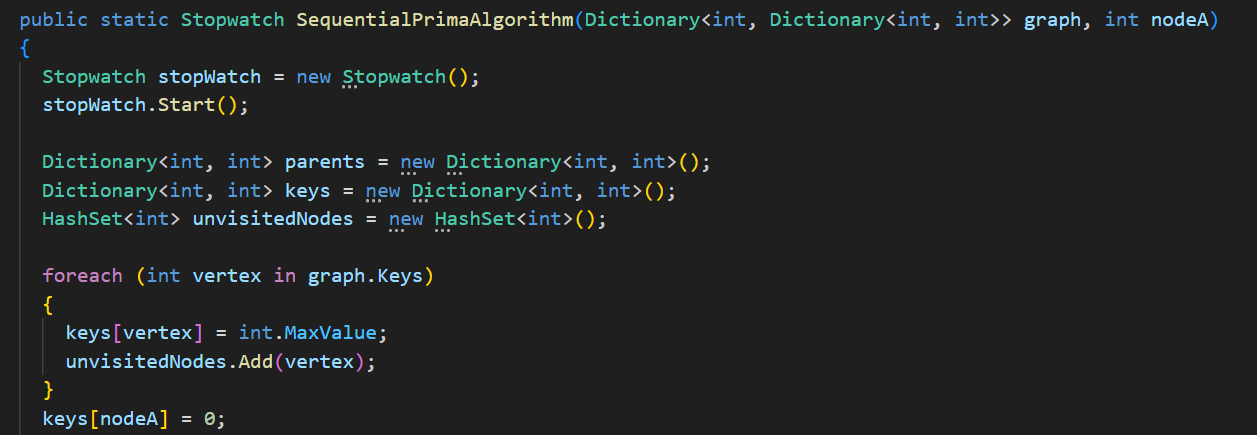
**Хід роботи:**

Для задачі я подаю граф у вигляді словника, де кожна вершина має словник зі значеннями кожної вершини, з якою вона з’днана, та вагою ребра між ними. Так як граф неорієнтований, вага ребер [i, j] = [j, i]. Вершин, між якими відсутні ребра, та петель немає в словнику.

**Послідовний алгоритм**

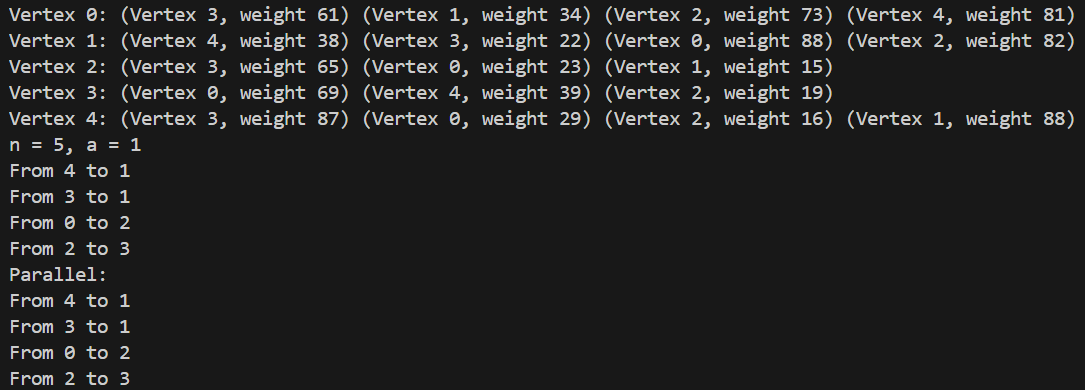
Послідовний алгоритм Дейкстри реалізовано у методі SequentialPrimaAlgorithm.

У ньому знаходиться кісткове дерево від заданої вершини **nodeA** за допомогою алгоритму Прима та зберігається у словник. Використано також об’єкт класу Stopwatch для засікання часу:





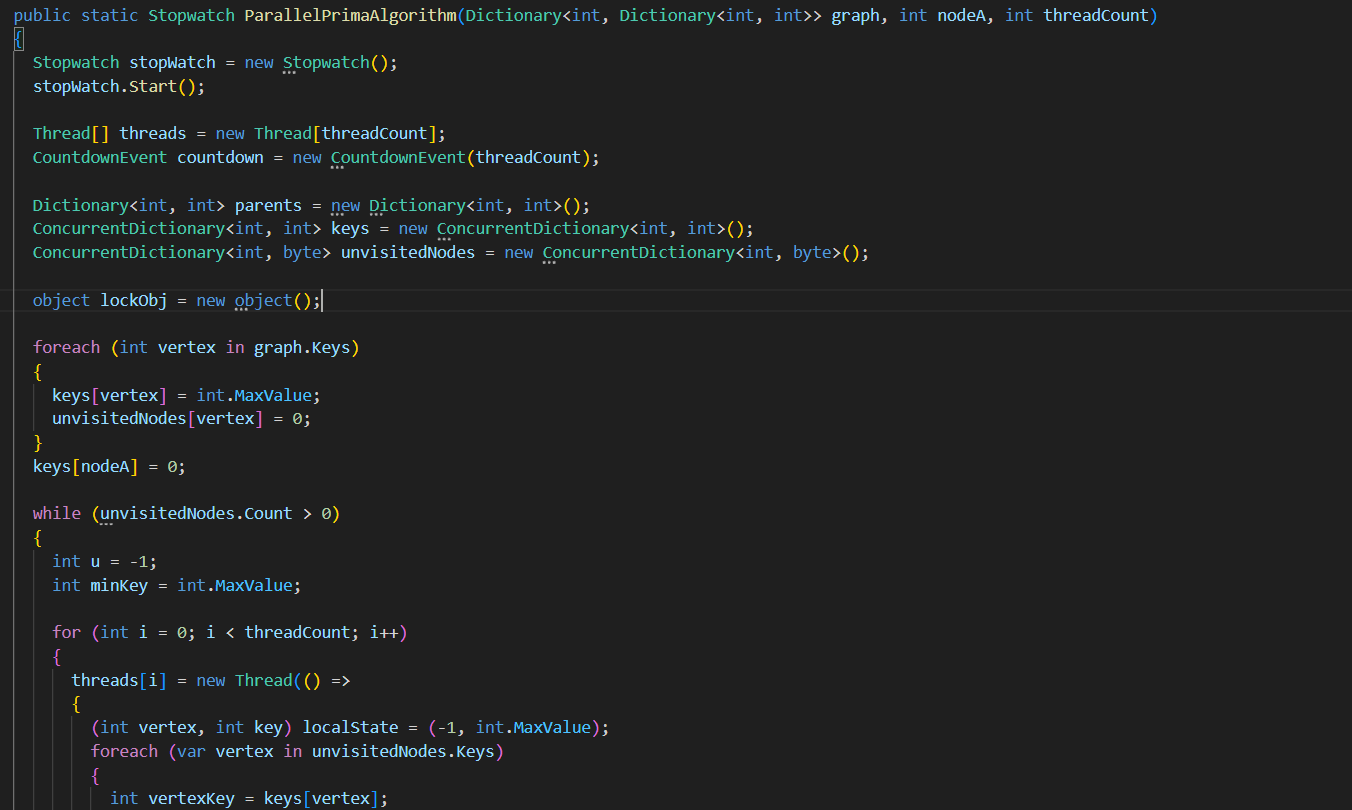
Крім цього, я перевірила, чи результати обчислюються правильно:

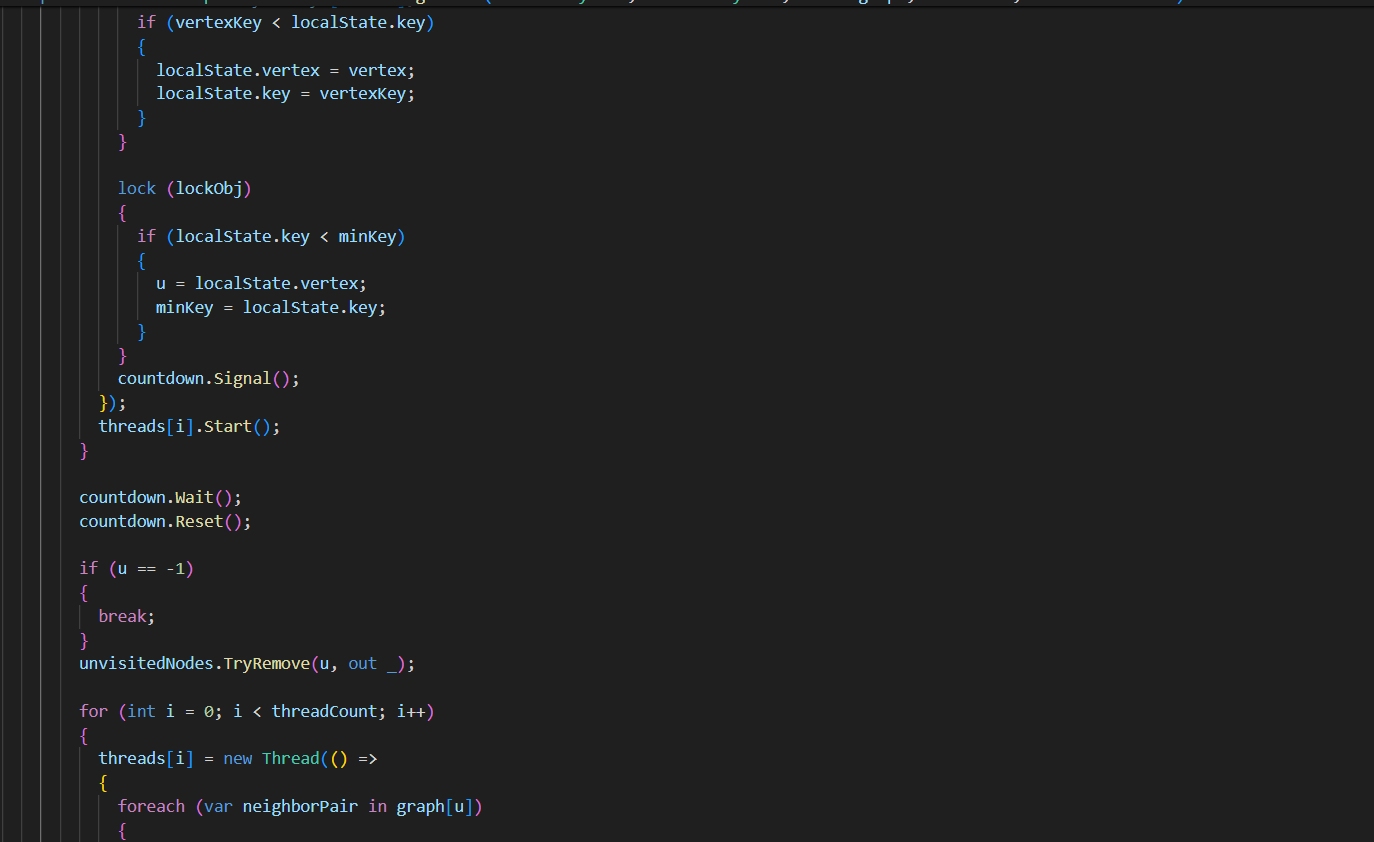


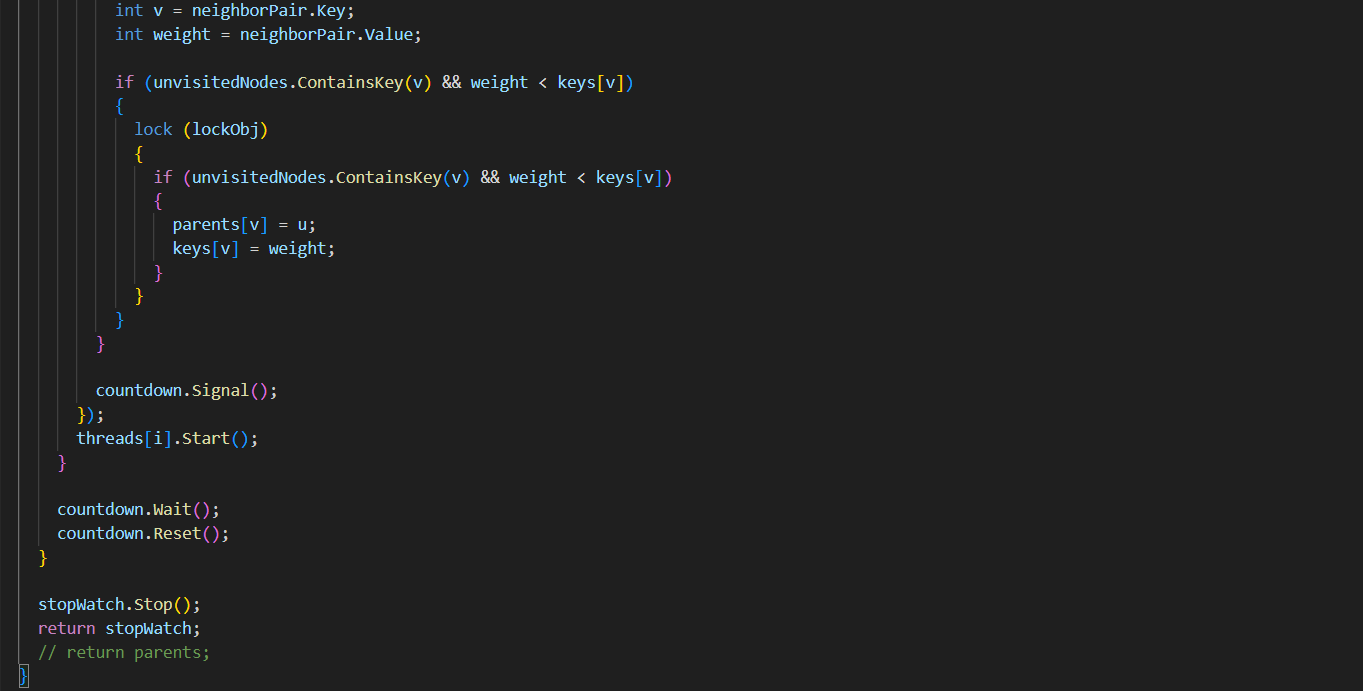
**Паралельний алгоритм**

Паралельний алгоритм Прима реалізовано у методі ParallelPrimAlgorithm.

Для розпаралелення я використовую об’єкти класу Thread та CountdownEvent для сигналу про завершення роботи потоків. Так як потоки залежать одне від одного, при оновленні відстані між вершинами використовується блокування (lock)



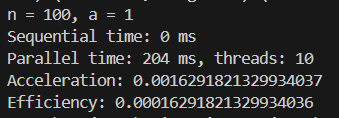


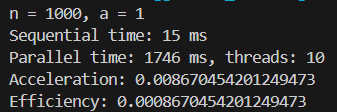


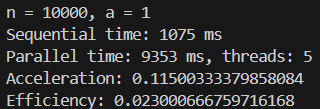
Метод також працює правильно (видно у результатах вище)

**Аналіз результатів**

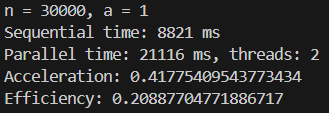
На малих розмірах графа розпаралелення зовсім не ефективне:

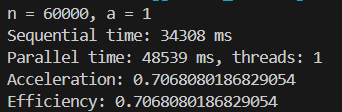




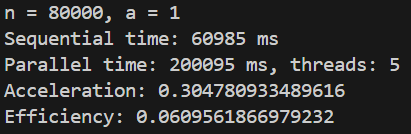


На більших розмірностях та при меншій кіькості потоків розпаралелення ефективніше, але все ще гірше за послідовний алгоритм:





Алгоритм Прима розпаралелювати дуже неефективно, тому бачимо, що навіть при великій розмірності даних та розумній кількості потоків розпаралелення дає поганий результат:



**Висновок:**

Виконуючи лабораторну роботу, я реалізувала послідовний та паралельний варіанти алгоритму Прима для побудови кістякового дерева у зваженому неорієнтованому графі. Для розпаралелення використала класи Thread та CountdownEvent. Порівняння часу виконання показало, що паралельний алгоритм не є ефективним для цієї задачі.