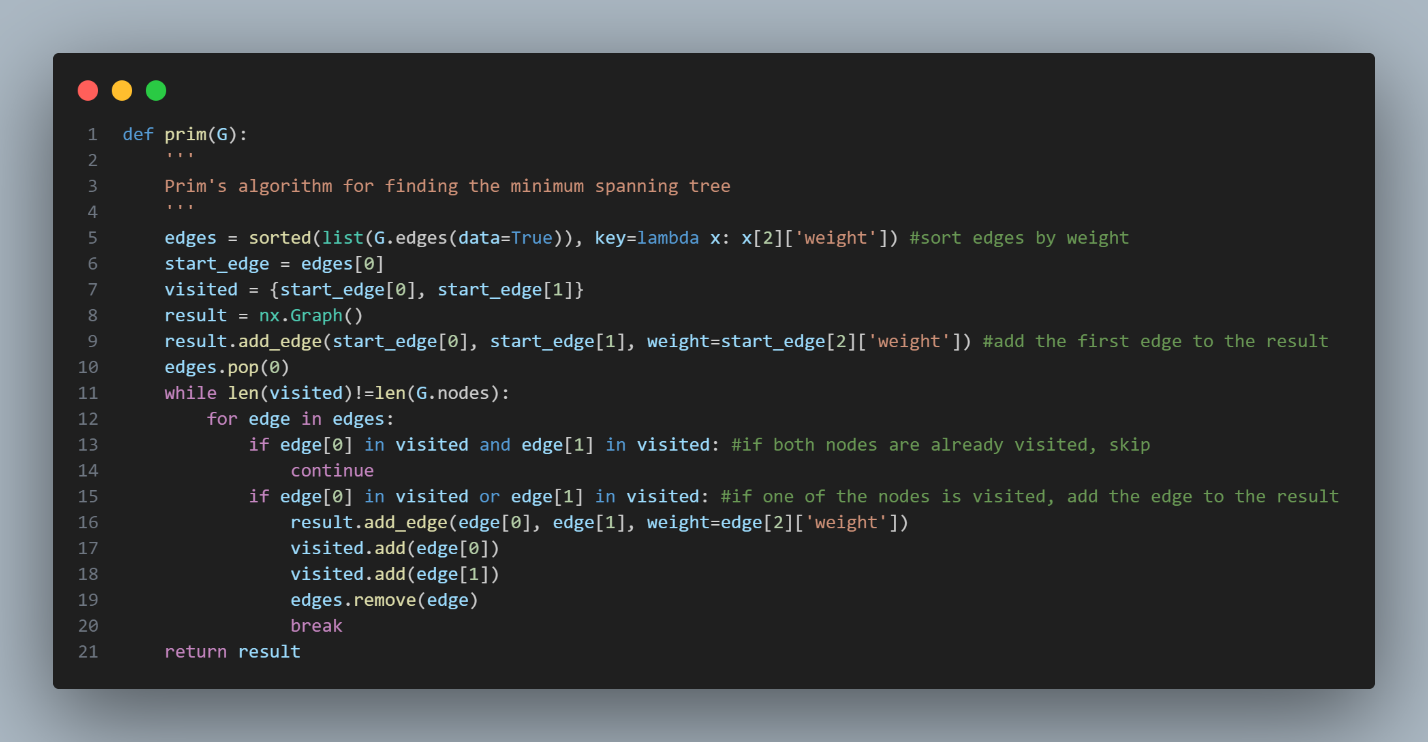
**Дослідження роботи алгоритмів Прима та Крускала**

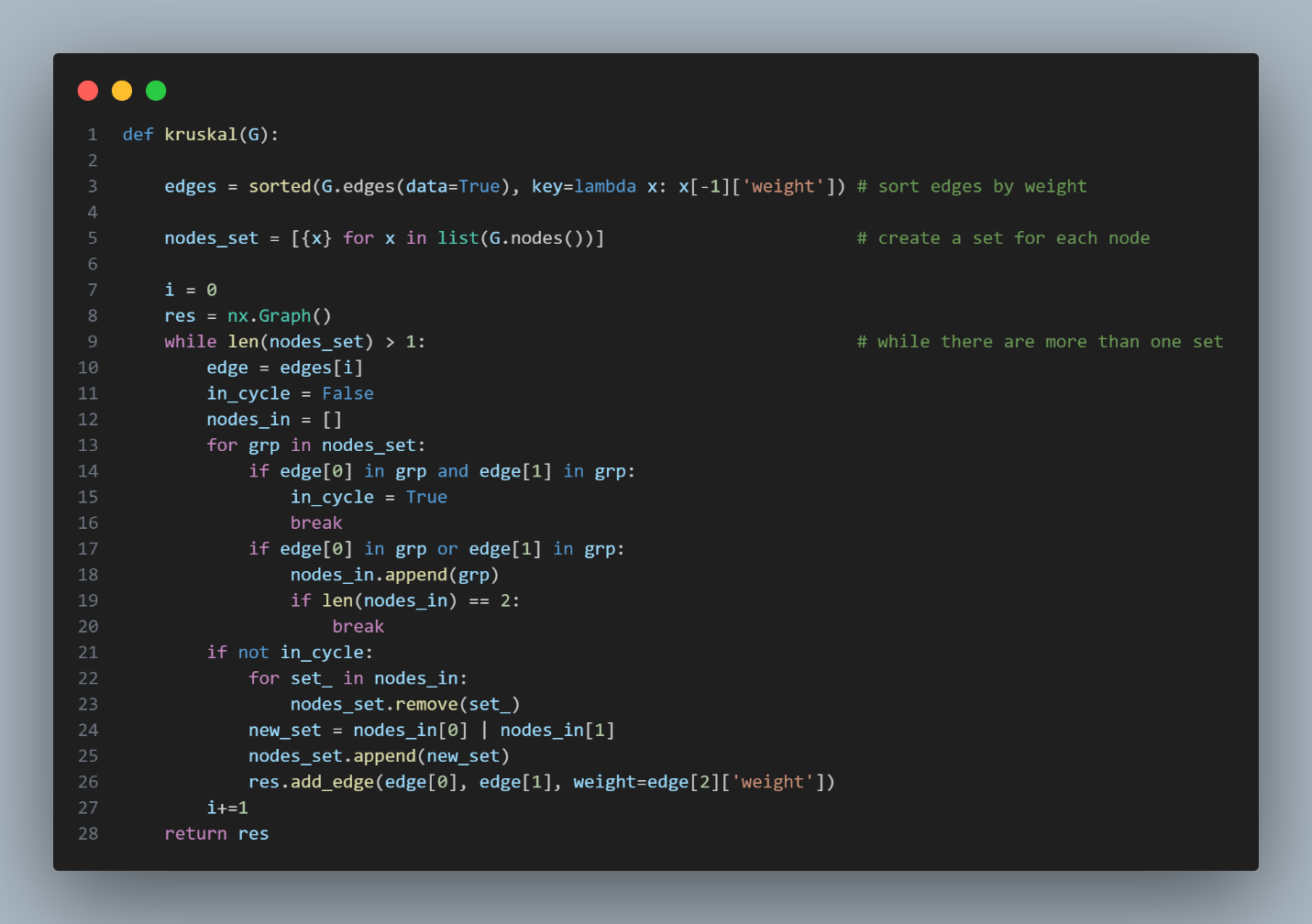
**Розподіл роботи: алгоритм Крускала – Котлярчук Оксана, алгоритм Прима – Тепляков Захарій**

**Алгоритм Прима**

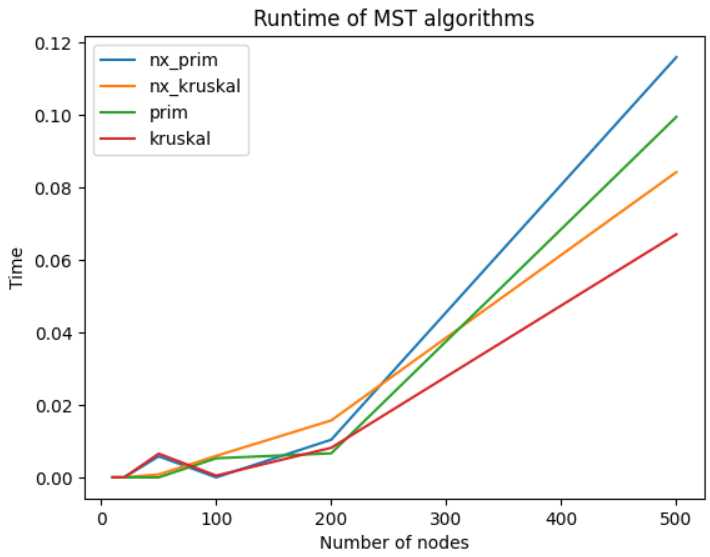
****

**Алгоритм приймає nx.Graph та для початку відсортовує усі його ребра для ефективної роботи. Після цього створюється пустий граф, у який додається вершина з найменшим ребром. Після цього, доки кількість вершин у новоутвореному графі не буде рівній кількості вершин у початковому графі, алгоритм проходить по ребрам та перевіряє, чи були відвідані ці вершини. Якщо дві вершини були відвідані – алгоритм пропускає дане ребро, але якщо лише одна вершина була відвідана, то алгоритм додає це ребро у новий граф, та видаляє дане ребро з початкового списку.**

**Алгоритм Крускала**

****

**Завдання алгоритму - знайти мінімальний каркас у графі. Спочатку імплементований алгоритм сортує всі ребра графу за зростанням ваги і розподіляє вершини по окремих множинах, тоді ітеративно перебирає ребра і додає множини, до яких належать його кінці у окремий список. Тоді обрані множини об'єднуються і додаються в загальний список і алгоритм переходить до наступного ребра. Якщо обидва кінці знаходяться в одній множині - ребро пропускається.**

****

**Як бачимо, алгоритми мають схожий час виконання, адже схоже імплементовані. На більших значеннях алгоритми nx працюють трохи швидше, адже використовують інші бібліотеки. У нашому випадку алгоритм Крускала працює найкраще, і на малих графах (до 500) з такою швидкістю і навіть швидше ніж версія nx.**