

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
«ЛЬВІВСЬКА ПОЛІТЕХНІКА»



Лабораторна робота №5
з курсу «Дискретні моделі в САПР»:

ІЗОМОРФІЗМ ГРАФІВ

Виконав:
Ст.гр.КН-409
Погуляєв В.В.

Мета роботи

Метою лабораторної роботи є вивчення і дослідження основних підходів до встановлення ізоморфізму графів.

Теоретичні відомості

Теорія графів дає простий, доступний і потужний інструмент побудови моделей і рішення задач впорядкування взаємозв'язаних об'єктів. Нині є багато проблем де необхідно дослідити деякі складні системи з допомогою впорядкування їх елементів. До таких проблем відносяться і задачі ідентифікації в електричних схемах, в авіації, в органічній хімії і т.д. Вирішення таких проблем досягається з допомогою встановлення ізоморфізму графів. Два графа $G=(X,U,P)$ і $G'=(X',U',P')$ називаються ізоморфними, якщо між їх вершинами, а також між їхніми ребрами можна встановити взаємно однозначне співвідношення $X \leftrightarrow X'$, $U \leftrightarrow U'$, що зберігає інцидентність, тобто таке, що для всякої пари $(x,u) \in X$ ребра $u \in U$, що з'єднує їх, обов'язково існує пара $(x',u') \in X'$ і ребро $u' \in U'$, що з'єднує їх, і навпаки. Тут P - предикат, інцидентор графа G . Зауважимо, що відношення ізоморфізму графів рефлексивне, симетричне і транзитивне, тобто представляє собою еквівалентність. На даний час існує досить детальна класифікація розроблених методів рішення такого типу задач [1]. Розглядаючи комбінаторно-логічну природу вказаної задачі можна всі роботи в цьому напрямку розділити на дві групи: рішення теоретичної задачі встановлення ізоморфізму простих графів; розробка наближених методів, які найбільш повно враховують обмеження і специфіку задачі з застосуванням характерних ознак об'єкту дослідження.

Лабораторне завдання

Реалізувати програмний код для встановлення ізоморфізму графів.

Код програми

Метод для перевірки ізоморфізму:

```
public boolean isIsomorphic(int[][] graph1, int[][] graph2){
    int n1 = graph1.length;
    int n2 = graph2.length;

    if(n1 != n2){
        return false;
    }

    int[] degree1 = new int[n1];
    int[] degree2 = new int[n2];
    for (int i = 0; i < n1; i++){
        for (int j = 0; j < n1; j++){
            if(graph1[i][j] > 0){
                degree1[i]++;
            }

            if(graph2[i][j] > 0){
                degree2[i]++;
            }
        }
    }
}
```

```
    }  
}  
  
for (int i = 0; i < n1; i++){  
    if(degree1[i] != degree2[i]){  
        return false;  
    }  
}  
return true;  
}
```

Посилання на GitHub –
https://github.com/flippflopp/DM_Pohuliaiev

Аналіз результатів

Аналітичний розв'язок:

Результат виконання програми використовуючи тестові файли I1_1.txt і I1_2.txt:

```
C:\Users\nazar\.jdk\openjdk-18.0.1\bin\java.exe -Djava.class.path=.\lib\graph.jar -jar I1_1.txt I1_2.txt  
These graphs aren't isomorphic
```

Результат виконання програми використовуючи тестові файли I2_2.txt і I2_2.txt:

```
C:\Users\nazar\.jdk\openjdk-18.0.1\bin\java.exe -Djava.class.path=.\lib\graph.jar -jar I2_2.txt I2_2.txt  
These graphs are isomorphic  
Process finished with exit code 0
```

Висновок

В ході виконання лабораторної роботи, вивчив і дослідив основні підходи для встановлення ізоморфізму графів.