

УДК 519.1

Застосування інформаційних технологій при вивченні дискретної математики

Федюк Є. М.¹, к.ф.-м.н., доц. каф. ОМП

Думанський О. І.², к.ф.-м.н., доц. каф. ІТ

Бекас Б. О.², ст. викл. каф. ІТ

Сохан П. Л.¹, к.ф.-м.н., доц. каф. ВМ

¹Національний університет «Львівська політехніка»

(вул. С. Бандери, 12, м. Львів, 79013, Україна)

²Національний лісотехнічний університет України

(вул. ген. Чупринки, 103, м. Львів, 79057, Україна)

Дисципліна “Дискретна математика” є однією з фундаментальних у системі математичної підготовки майбутніх інженерів–програмістів і розрахована на аудиторію, достатньо підготовлену в галузі математики. За сучасних вимог до організації навчального процесу поруч із традиційною формою викладання (проведення лекційних та практичних занять без застосування комп’ютерної техніки) видається логічним і доцільним запровадження лабораторного практикуму з інтенсивним використанням комп’ютерних середовищ.

Метою доповіді є аналіз та синтез шляхів формулювання постановок задач та їх реалізації у середовищі візуальної розробки програм Microsoft Visual Studio за допомогою мови програмування C++.

Це можуть бути задачі, що виникають при вивченні теорії множин (побудова множин відповідно до правил, за якими формуються множини чи випадковим чином, сортування елементів множин, їх статистична обробка, операції над нечіткими множинами з побудовою цих множин та їх функцій належності тощо); відношень (побудова відношень за їх способом подання — у вигляді графів або матриць, дії над чіткими та нечіткими відношеннями та інше); теорії графів (побудова графів, подання графів у вигляді списку ребер чи вершин, подання графів матрицями суміжності, інцидентності та Кірхгофа, обхід графів, пошук з поверненням (бектрекінг), дослідження зважених графів за алгоритмами Дейкстри та Флойда тощо).

Для прикладу наведемо лістинг програми для подання неорієнтованого графа матрицею суміжності та результат роботи програми.

Програма подання неорієнтованого графа матрицею суміжності

Можливий вигляд файла 'Graph.txt' :

3

0 1

0 2

2 1

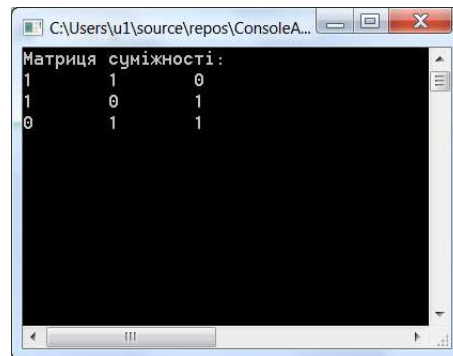
```
\#include "stdafx.h"
```

```
\#include <<fstream>>
```

```

#include <fstream>
using namespace std;
int main()
{ int **pnGraph, i, j, N;
  setlocale(LC_ALL, "UKR");
  //Відкриваємо файл для читання (необхідно вказати правильний шлях)
  ifstream input("D:\\Graph.txt");
  //Читаємо кількість вершин
  input >> N;
  //Виділяємо пам'ять для матриці суміжності
  pnGraph=new int *[N];
  for (i=0; i<N; i++)
    pnGraph[i]=new int [N];
  //Занулюємо всі елементи матриці
  for (i = 0; i<N; i++)
    for (j = 0; j<N; j++)
      pnGraph[i][j]=0;
  //Заповнюємо матрицю даними з файла
  while (!input.eof())
  {input >> i >> j;
   pnGraph[i][j]=1;
   pnGraph[j][i]=1;
   input.close();}
  //Виводимо на консоль матрицю суміжності
  cout << "Матриця суміжності:" << endl;
  for (i = 0; i<N; i++)
    { for (j = 0; j<N; j++)
      { cout << pnGraph[i][j] << "\\t";
        } cout << endl; }
  //Звільняємо пам'ять виділену під матрицю
  delete[] pnGraph;
  return 0;
}

```



```

Матриця суміжності:
1 1 0
1 0 1
0 1 1

```

У процесі вивчення дискретної математики студенти мають отримати необхідні знання з основ і прикладних методів аналізу та синтезу об'єктів і процесів дискретної природи, що може служити апаратом для подальшого розуміння основних методів дослідження, проектування та експлуатації комп'ютерних систем і мереж.

Актуальним питанням є вирішення співвідношення фундаментальної (лекційної) і практичної (лабораторно–практичної) складових у рамках дисципліни. Практична спрямованість (програмне забезпечення лабораторних робіт) забезпечить наскрізний процес для повноцінного розуміння та застосовності середовища Microsoft Visual Studio при вивченні інших дисциплін.

