Міністерство освіти і науки України

Національний авіаційний університет

Факультет кібербезпеки, комп’ютерної та програмної інженерії

Кафедра інженерії програмного забезпечення

**Лабораторна робота № 1.2**

«Керувальний граф програми»

**Виконано студентом групи:**

ПІ-322, Царук С. О.

**Перевірено:**

Волкогон В.О.

Київ – 2021

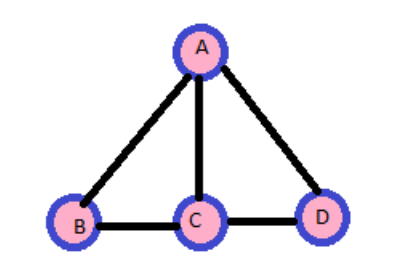
**Мета лабораторної роботи –** ознайомитись з моделлю програми у вигляді графа, отримати практичні навички побудови керувального графа програми, розрахунку дипломатичної складності графа та її застосування при плануванні налагоджування програми.

**Завдання**

1. По заданій суміжній матриці побудувати граф.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **А** | **B** | **C** | **D** |
| **A** |  | 0 | 1 | 1 |
| **B** | 0 |  | 1 | 0 |
| **C** | 1 | 1 |  | 0 |
| **D** | 1 | 0 | 0 |  |

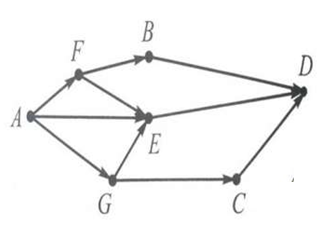
1. По заданому графу побудувати матрицю суміжності.



1. Знайти найкоротший маршрут між пунктами А і D (побудувати граф).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **A** | **B** | **C** | **D** | **E** |
| **A** |  | 2 | 4 | - | 6 |
| **B** | 2 |  | 1 | - | - |
| **C** | 4 | 1 |  | 5 | 1 |
| **D** | - | - | 5 |  | 3 |
| **E** | 6 | - | 1 | 3 |  |

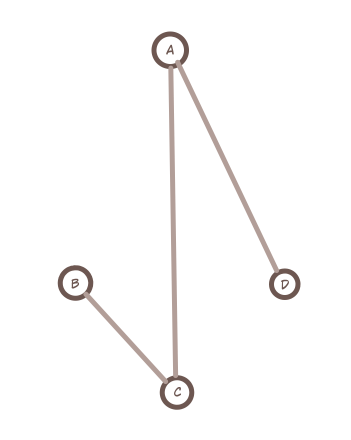
1. На малюнку зображена схема з’єднань, зв’язкових пунктів A,F,G,B,E,C,D. По кожному з’єднанні можливо рухатися тільки в одному напрямку, показано стрілкою. Скільки існує різних шляхів із пункту А в пункт D?



1. Скласти програму рішення задачі, оформити її у вигляді процедури з вхідними та вихідними параметрами (у вигляді UML діаграми).
2. Побудувати управляючий граф створеної програми.
3. Розрахувати цикломатичене число графа для оцінки складності програми.
4. Оцінити кількість тестів необхідних для перевірки розробленої програми (необхідно забезпечити повне покриття шляхів і графа)
5. Протестувати розроблену програму.
6. Уявити управляючий граф програми у вигляді матриці суміжності і матриці інцеденцій

**Хід роботи**

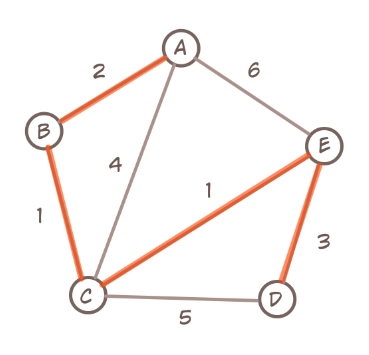
1. Результат:



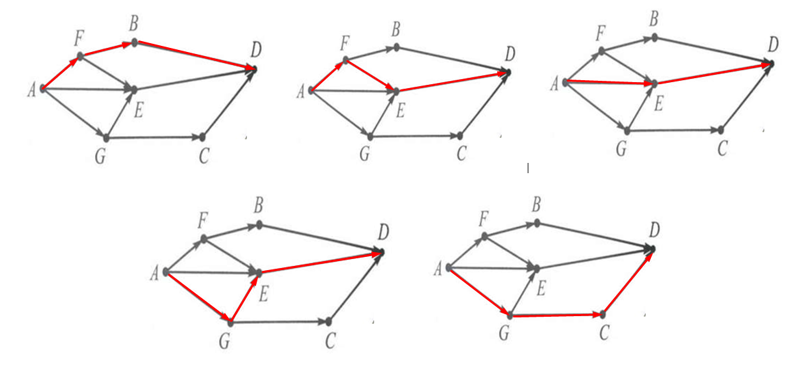
1. Результат:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **A** | **B** | **C** | **D** |
| **A** |  | 1 | 1 | 1 |
| **B** | 1 |  | 1 | 0 |
| **C** | 1 | 1 |  | 1 |
| **D** | 1 | 0 | 1 |  |

1. Результат:



1. Результат: 5.



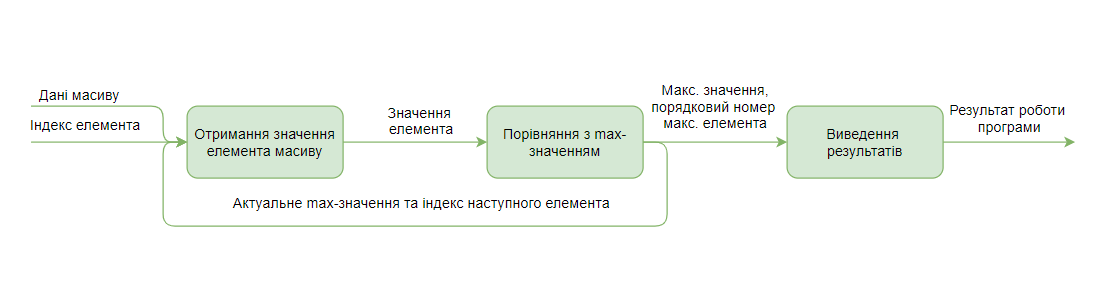
1. Обрана задача:

2) Дано n цілих чисел. Знайти серед них максимальне та його порядковий номер. Якщо є декілька таких чисел, то визначте порядковий номер першого такого числа.

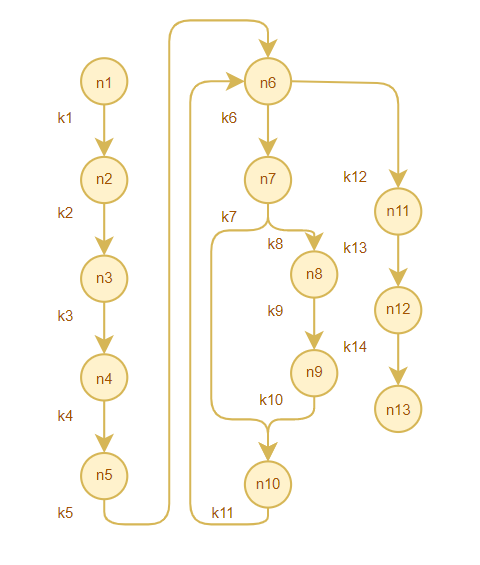
Код (завдання виконано на мові Java):

package com.company;  
  
public class Main {  
  
 public static void main(String[] args) {  
 int[] arr = new int[] {1, -2, 3, 5, 3, 11, -18, 11, 4, 6};  
 int l = arr.length;  
 int max = arr[0];  
 int index = 1;  
  
 System.*out*.print("Масив: " + arr[0] + " ");  
  
 for(int i = 1; i < l; i++)  
 {  
 if(arr[i] > max)  
 {  
 max = arr[i];  
 index = i + 1;  
 }  
 System.*out*.print(arr[i] + " ");  
 }  
  
 System.*out*.println();  
 System.*out*.println("Максимальне число: " + max);  
 System.*out*.println("Порядковий номер: " + index);  
 }  
}

Діаграма:

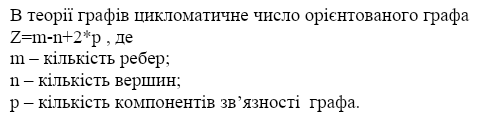


1. Керуючий граф.



1. Розрахунок цикломатичного числа.

Формула:



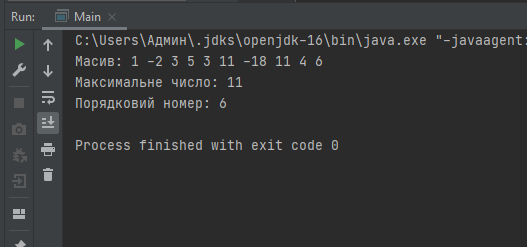
Результат: 13 – 14 + 2\*1 = 1.

1. Оцінка необхідної кількості тестів.

Для проведення тестування з покриттям всіх шляхів графу потрібен 1 тест, за умови, що перше число в масиві не є найбільшим.

1. Тестування.

Результат:



1. Матриці.

Матриця суміжності:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **n1** | **n2** | **n3** | **n4** | **n5** | **n6** | **n7** | **n8** | **n9** | **n10** | **n11** | **n12** | **n13** |
| **n1** | 1 | 1 |  |  |  |  |  |  |  |  |  |  |  |
| **n2** | 1 | 1 | 1 |  |  |  |  |  |  |  |  |  |  |
| **n3** |  | 1 | 1 | 1 |  |  |  |  |  |  |  |  |  |
| **n4** |  |  | 1 | 1 | 1 |  |  |  |  |  |  |  |  |
| **n5** |  |  |  | 1 | 1 | 1 |  |  |  |  |  |  |  |
| **n6** |  |  |  |  | 1 | 1 | 1 |  |  | 1 | 1 |  |  |
| **n7** |  |  |  |  |  | 1 | 1 | 1 |  | 1 |  |  |  |
| **n8** |  |  |  |  |  |  | 1 | 1 | 1 |  |  |  |  |
| **n9** |  |  |  |  |  |  |  | 1 | 1 | 1 |  |  |  |
| **n10** |  |  |  |  |  | 1 | 1 |  | 1 | 1 |  |  |  |
| **n11** |  |  |  |  |  | 1 |  |  |  |  | 1 | 1 |  |
| **n12** |  |  |  |  |  |  |  |  |  |  | 1 | 1 | 1 |
| **n13** |  |  |  |  |  |  |  |  |  |  |  | 1 | 1 |

Матриця інцидентності:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **k1** | **k2** | **k3** | **k4** | **k5** | **k6** | **k7** | **k8** | **k9** | **k10** | **k11** | **k12** | **k13** | **k14** |
| **n1** | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **n2** | -1 | 1 |  |  |  |  |  |  |  |  |  |  |  |  |
| **n3** |  | -1 | 1 |  |  |  |  |  |  |  |  |  |  |  |
| **n4** |  |  | -1 | 1 |  |  |  |  |  |  |  |  |  |  |
| **n5** |  |  |  | -1 | 1 |  |  |  |  |  |  |  |  |  |
| **n6** |  |  |  |  | -1 | 1 |  |  |  |  | -1 | 1 |  |  |
| **n7** |  |  |  |  |  | -1 | 1 | 1 |  |  |  |  |  |  |
| **n8** |  |  |  |  |  |  |  | -1 | 1 |  |  |  |  |  |
| **n9** |  |  |  |  |  |  |  |  | -1 | 1 |  |  |  |  |
| **n10** |  |  |  |  |  |  | -1 |  |  | -1 | 1 |  |  |  |
| **n11** |  |  |  |  |  |  |  |  |  |  |  | -1 | 1 |  |
| **n12** |  |  |  |  |  |  |  |  |  |  |  |  | -1 | 1 |
| **n13** |  |  |  |  |  |  |  |  |  |  |  |  |  | -1 |

**Висновки**

Під час виконання лабораторної роботи я ознайомився з моделлю програми у вигляді графа, отримв практичні навички побудови керувального графа програми, розрахунку дипломатичної складності графа та її застосування при плануванні налагоджування програми, створенні матриць інцидентності та суміжності.