Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського"

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 1 з дисципліни «Алгоритми та структури даних-1. Основи алгоритмізації»

«Дослідження алгоритмів розгалуження»

Варіант 16

Виконав студент ІП-11 Кузьменков Дмитро Олегович

(шифр, прізвище, ім'я, по батькові)

Перевірив

( прізвище, ім'я, по батькові)

Київ 2021

**Лабораторна робота 2**

**Дослідження лінійних алгоритмів**

**Мета** - дослідити подання керувальної дії чергування у вигляді умовної та альтернативної форм та набути практичних навичок їх використання під час складання програмних специфікацій.

# Завдання:

Задані дійсні додатні числа *а*, *b*, *с*, *d*. З’ясувати, чи можна прямокутник із сторонами *а*, *b* розмістити всередині прямокутник із сторонами *с*, *d* так, щоб кожна із сторін одного прямокутника була паралельна або перпендикулярна кожній стороні другого прямокутника.

1. **Постановка задачі:**

Отримавши значення сторін першого прямокутника (**a, b**) та другого прямокутника (**c, d**) знаходимо більшу та меншу сторону кожного прямокутника за допомогою **max()** та **min()** відповідно. Значення більших сторін двох прямокутників порівнюємо між собою, те саме робимо з меншими сторонами. Якщо більша сторона першого прямокутника менша за більшу сторону другого прямокутника та менша сторона першого менша за меншу сторону другого, тоді прямокутник зі сторонами a, b можна розмітити всередині прямокутника зі сторонами c, d. Якщо ж наведена вище умова не виконується, то розмістити перший прямокутник всередині другого неможливо.

# Побудова математичної моделі. Складемо таблицю імен змінних:

|  |  |  |  |
| --- | --- | --- | --- |
| Змінна | Тип | Ім’я | Призначення |
| Сторона першого прямокутника | Раціональне | a | Початкове дане |
| Сторона першого прямокутника | Раціональне | b | Початкове дане |
| Сторона другого прямокутника | Раціональне | c | Початкове дане |
| Сторона другого прямокутника | Раціональне | d | Початкове дане |
| Більша сторона першого прямокутника | Раціональне | Max1 | Проміжне дане |
| Менша сторона першого прямокутника | Раціональне | Min1 | Проміжне дане |
| Більша сторона другого прямокутника | Раціональне | Max2 | Проміжне дане |
| Менша сторона другого прямокутника | Раціональне | Min2 | Проміжне дане |

Таким чином, формулювання задачі зводиться до визначення більших та менших сторін обох прямокутників та їх порівняння.

Крок 1. Знаходимо max(a,b) та min(a,b).

Крок 2. Знаходимо max(c,d) та min(c,d).

Крок 3. Порівнюємо max(a,b) та max(c,d).

Крок 4. Порівнюємо min(a,b) та min(c,d).

1. **Псевдокод:**
2. Знайти max(a, b) та min(a, b)
3. Знайти max(c, d) та min(c, d)
4. Порівняти max(a, b) та max(c, d)
5. Порівняти min(a, b) та min(c, d)

*Крок1*

# Початок

1. Max1:= max(a, b)

Min1:= min(a, b)

1. Знайти max(c, d) та min(c, d)
2. Порівняти max(a, b) та max(c, d)
3. Порівняти min(a, b) та min(c, d)

**Кінець**

*Крок2*

# Початок

1. Max1: = max(a, b)

Min1:= min(a, b)

1. Max2: = max(c, d)

Min2:= min(c, d)

1. Порівняти max(a, b) та max(c, d)
2. Порівняти min(a, b) та min(c, d)

**Кінець**

*Крок3*

**Початок**

1. Max1: = max(a, b)

Min1:= min(a, b)

1. Max2: = max(c, d)

Min2:= min(c, d)

1. **Якщо** Max1 <= Max2

**то** переходимо до наступного етапу

**інакше** прямокутник не можна розмістити

1. Порівняти min(a, b) та min(c, d)

**Кінець**

*Крок4*

**Початок**

1. Max1: = max(a, b)

Min1:= min(a, b)

1. Max2: = max(c, d)

Min2:= min(c, d)

1. **Якщо** Max1 <= Max2

**то** переходимо до наступного етапу

**інакше** прямокутник неможливо розмістити

1. **Якщо** Min1 <= Min2

**то** прямокутник можна розмістити

**інакше** прямокутник не можна розмістити

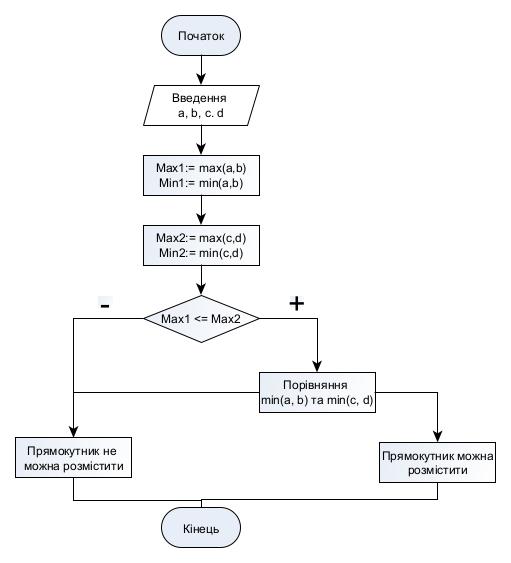
**Кінець**

**4. Блок-схема**

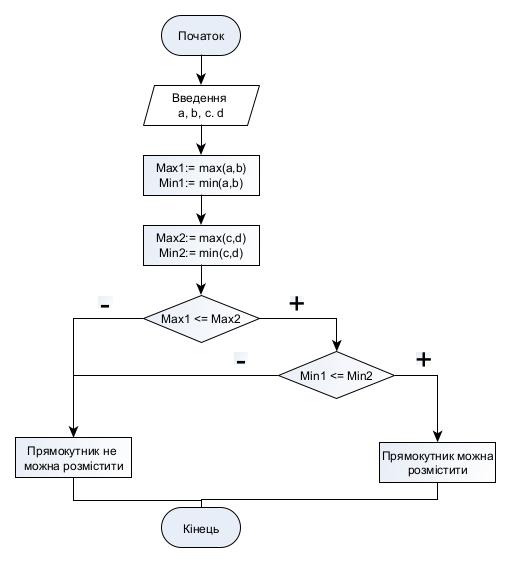
*Крок1 Крок2*

|  |  |  |
| --- | --- | --- |
|  |  |  |

*Крок3*



*Крок4*

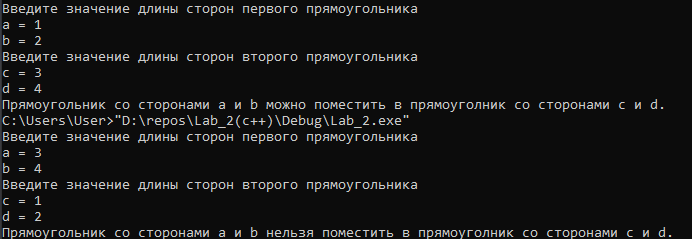


**5. Випробування алгоритму:**

|  |  |
| --- | --- |
| **Блок** | **Дія** |
|  | Початок |
| **1** | Введення a = 1, b = 2, c = 3, d = 4 |
| **2** | Max1: = 2 |
| **3** | Min1: = 1 |
| **4** | Max2: = 4 |
| **5** | Min2:= 3 |
| **6** | 2 < 4 |
| **7** | 1 < 3 |
| **8** | “Прямокутник можна розмістити” |
|  | Кінець |

|  |  |
| --- | --- |
| **Блок** | **Дія** |
|  | Початок |
| **1** | Введення a = 3, b = 4, c = 1, d = 2 |
| **2** | Max1: = 4 |
| **3** | Min1: = 3 |
| **4** | Max2: = 2 |
| **5** | Min2: = 1 |
| **6** | 4 > 2 |
| **7** | 3 > 1 |
| **8** | “Прямокутник неможливо вмістити” |
|  | Кінець |

**6. Результат:**



**Висновок:** Виконуючи лабораторну роботу, я дослідив лінійні програмні специфікації для подання перетворювальних операторів та операторів суперпозиції, набув практичних навичок їх використання під час складання лінійних програмних специфікацій. Створив алгоритм для дослідження можливості розміщення одного прямокутника всередині іншого шляхом знаходження більших та менших сторін цих прямокутників та їх порівняння.