# 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 № 6 з дисципліни

«Алгоритми та структури даних-1.

Основи алгоритмізації»

«Дослідження лінійних алгоритмів» Варіант 2

Виконав студент Боровков Іван Ігорович

(шифр, прізвище, ім'я, по батькові)

Перевірив

( прізвище, ім'я, по батькові)

Київ 2021

**Мета:**

Набути навичок складання і використання підпрограм користувача

**Умова задачі:**

Три трикутники задані своїми сторонами А, В, С. За допомогою формули Герона обчислити площі трикутників та визначити, площа якого є найбільшою.

**Постановка задачі:**

Розбити задачу на підзадачі, які реалізуємо в підпрограммах. Такими підзадачами є операції введення, виведення, обчислення площі за допомогою формули Герона та визначення більшої площі.

**Математична модель:**

**Складемо таблицю імен змінних:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Змінна** | **Тип** | **Ім’я** | **Призначення** |
| Сторони трикутників | Дійсний | a1, b1, c1, a2, b2, c2, a3, b3, c3 | Вхідні дані |
| Площі трикутників | Дійсний | square1, square2, square3 | Вихідні дані |
| Найбільша площа | Дійсний | answer | Вихідні дані |
| Півпериметр | Дійсний | p | Допоміжна змінна |
| Перевірка трикутника на існування | Логічний | f | Допоміжна змінна |

**Текст файла проєкту**

**Python**

from math import sqrt

def calculate\_square\_triangle(a, b, c):

p = (a + b + c) / 2

square = sqrt(p \* (p - a) \* (p - b) \* (p - c))

return square

def calculate\_max\_square(square1, square2, square3):

if((square1 > square2) and (square1 > square3)):

max = square1

if((square2 > square3) and (square2 > square1)):

max = square2

if((square3 > square2) and (square3 > square1)):

max = square3

return max

def inputValues():

f = True

while(f):

f = False

a = float(input())

b = float(input())

c = float(input())

if((a >= b + c) or (b >= a + c) or (c >= b + a)):

print("Такого трикутника не існує")

f = True

return a, b, c

def output(square1, square2, square3, answer):

print("Площа першого трикутника: ", square1, "\nПлоща другого трикутника: ", square2, "\nПлоща третього трикутника: ", square3, "\nНайбільша площа трикутника: ", answer)

print("Введіть сторони першого трикутника:")

a1, b1, c1 = inputValues()

print("Введіть сторони другого трикутника:")

a2, b2, c2 = inputValues()

print("Введіть сторони третього трикутника:")

a3, b3, c3 = inputValues()

square1 = calculate\_square\_triangle(a1, b1, c1)

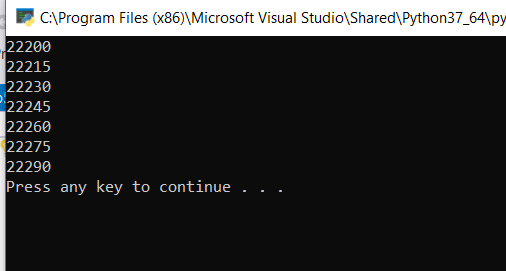
square2 = calculate\_square\_triangle(a2, b2, c2)

square3 = calculate\_square\_triangle(a3, b3, c3)

output(square1, square2, square3, calculate\_max\_square(square1, square2, square3) )

**Копії екранних форм результатів роботи**

Програма виводить користувачу числа виду 222\*\*, що діляться на 15.



**C++**

#include <iostream>

#include <math.h>

using namespace std;

double calculate\_square\_triangle(double a, double b, double c);

double calculate\_max\_square(double square1, double square2, double square3);

void input(double &a, double &b, double &c);

void output(double square1, double square2, double square3, double answer);

int main()

{

setlocale(LC\_ALL, "");

double a1, b1, c1, a2, b2, c2, a3, b3, c3;

cout << "Введiть сторони першого трикутника\n";

input(a1, b1, c1);

cout << "Введiть сторони другого трикутника\n";

input(a2, b2, c2);

cout << "Введiть сторони третього трикутника\n";

input(a3, b3, c3);

//Обчислюємо площі трикутників

double square1 = calculate\_square\_triangle(a1, b1, c1);

double square2 = calculate\_square\_triangle(a2, b2, c2);

double square3 = calculate\_square\_triangle(a3, b3, c3);

//обчислюємо найбільшу площу

output(square1, square2, square3, calculate\_max\_square(square1, square2, square3) );

return 1;

}

double calculate\_square\_triangle(double a, double b, double c)

{

double p = (a + b + c) / 2;

double square = sqrt(p \* (p - a) \* (p - b) \* (p - c));

return square;

}

double calculate\_max\_square(double square1, double square2, double square3)

{

double max;

if ((square1 > square2) && (square1 > square3)) {

max = square1;

}

if ((square2 > square1) && (square2 > square3)) {

max = square2;

}

if ((square3 > square2) && (square3 > square1)) {

max = square3;

}

return max;

}

void input(double &a, double &b, double &c)

{

bool f = true;

while (f)

{

f = false;

cin >> a >> b >> c;

if ((a + b <= c) || (c + b <= a) || (a + c <= b))

{

f = true;

cout << "Такого трикутника не iснує\n";

}

}

}

void output(double square1, double square2, double square3, double answer)

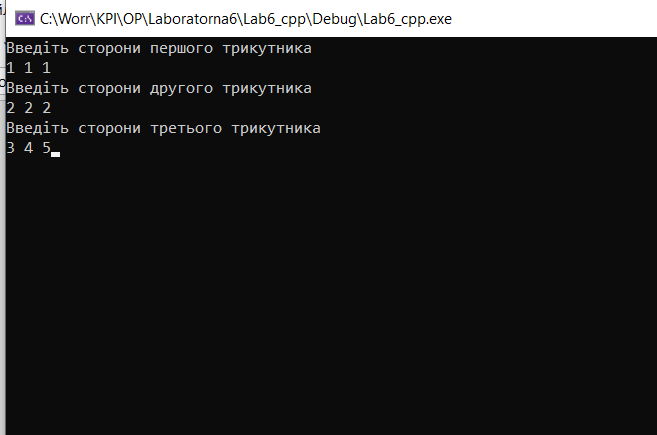
{

cout << "Площа першого трикутника: " << square1 << "\nПлоща другого трикутника: " << square2 << "\nПлоща третього трикутника: " << square3 << "\nНайбiльша площа трикутника: " << answer;

}

**Копії екранних форм результатів роботи**

Програма запиту. в користувача значення сторін трикутників...

... та виводить на екран площі трикутників та найбільшу з них.

**Висновок:** в ході лабораторної роботи я набув навичок складання і використання підпрограм користувача