

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 № 5 з дисципліни  
«Основи програмування-1.  
Базові конструкції»

«Організація циклічних процесів.

Складні цикли»

Варіант 28

Виконав студент ПІ-11 Сідак Кирил Ігорович  
(шифр, прізвище, ім'я, по батькові)

Перевірив \_\_\_\_\_  
( прізвище, ім'я, по батькові)

Київ 2021

**Мета** – вивчити особливості організації складних циклів.

## Варіант 28

Отримати всі піфагорові трійки натуральних чисел, кожне з яких не перевищує  $n$ , тобто всі такі трійки натуральних чисел  $a, b, c$ , що  $a^2 + b^2 = c^2$  ( $a \leq n, b \leq n, c \leq n$ ).

### Постановка задачі:

Для кожного натурального числа  $a$  та  $b$ , кожне з яких не більше заданого числа  $n$ , треба знайти таке число  $c$  ( $c$  не більше  $n$ ), що  $a^2 + b^2 = c^2$ . Тобто, якщо  $c = \sqrt{a^2 + b^2}$  – ціле число, то  $a, b, c$  – одна з шуканих піфагорових трійок чисел.

### Програма на C++:

/\*Варіант

28

```
Отримати всі піфагорові трійки натуральних чисел, кожне з яких не перевищує  $n$ , тобто всі такі
трійки натуральних чисел  $a, b, c$ , що  $a^2 + b^2 = c^2$  ( $a \leq n, b \leq n, c \leq n$ ). */
#include <iostream>
using namespace std;
#include <cmath>
int main() {
    int n;
    double c;
    cout << "Enter natural number n: ";
    cin >> n; // Введення натурального числа n
    for(int a = 1; a <= n; a++) // Цикл для числа a, де ітерація йде від 1 до n включно
    {
        for(int b = a + 1; b <= n; b++) // Вкладений цикл для числа b, де ітерація йде від a+1 до n включно
        {
            c = sqrt(a * a + b * b);
            if(c == int(c) && (c <= n))
            {
                cout << "a = " << a << " b = " << b << " c = " << c << "\n"; /* Якщо  $a^2 + b^2 = c^2$ ,
                тобто корінь з  $a^2 + b^2$  є цілим числом, то виведемо ці значення */
            }
        }
    }
```

```

    }
}
return 0;
}

```

## Програма на Python:

```

# Варіант 28

# Отримати всі піфагорові трійки натуральних чисел, кожне з яких не перевищує n, тобто всі такі
# трійки натуральних чисел a, b, c, що  $a^2 + b^2 = c^2$  ( $a \leq n$ ,  $b \leq n$ ,  $c \leq n$ ).

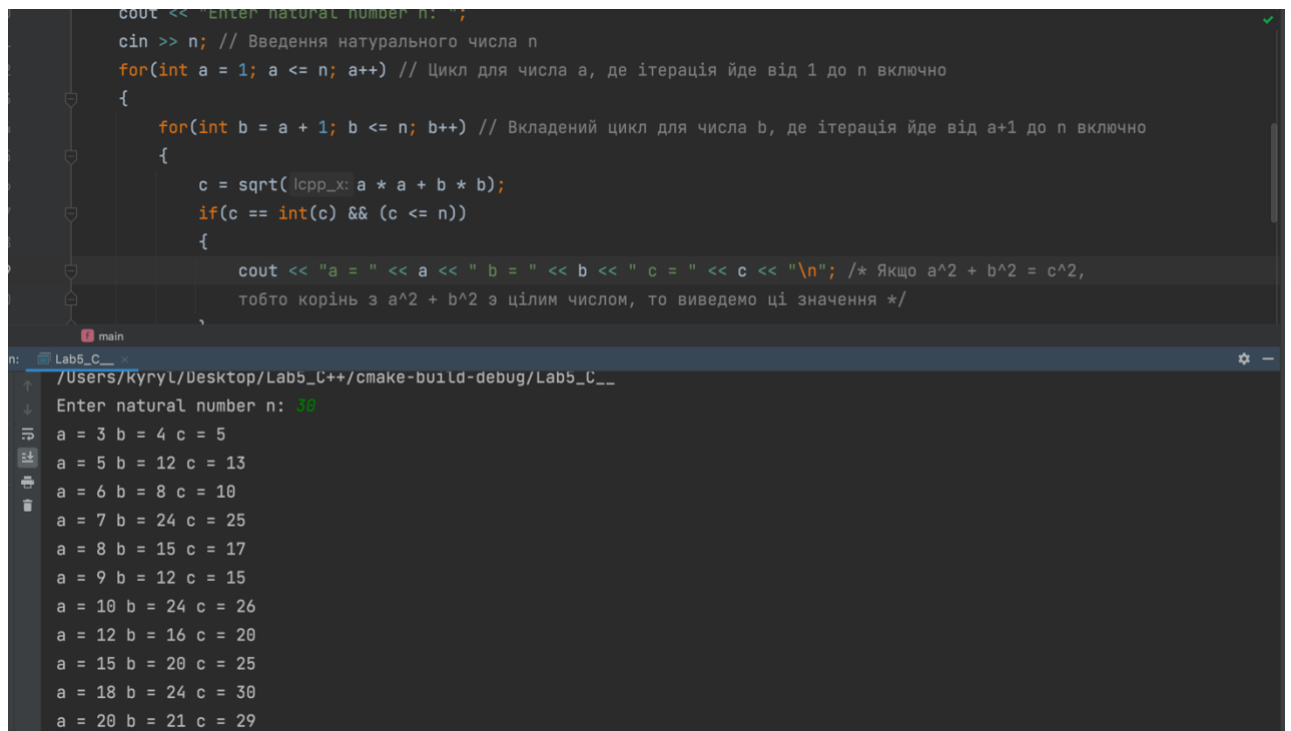
import math
import time

n = int(input("Enter natural number n: ")) # Введення натурального числа n

for a in range(1, n+1): # Цикл для числа a, де ітерація йде від 1 до n включно
    for b in range(a+1, n+1): # Вкладений цикл для числа b, де ітерація йде від a+1 до n включно
        c = math.sqrt(a ** 2 + b ** 2)
        if c == int(c) and c <= n: # Якщо  $a^2 + b^2 = c^2$ , тобто корінь з  $a^2 + b^2$  є цілим числом,
            # то виведемо ці значення
            c = int(c)
            print(f'a = {a}, b = {b}, c = {c}')

```

## Результат на C++:



```

cout << "Enter natural number n: ";
cin >> n; // Введення натурального числа n
for(int a = 1; a <= n; a++) // Цикл для числа a, де ітерація йде від 1 до n включно
{
    for(int b = a + 1; b <= n; b++) // Вкладений цикл для числа b, де ітерація йде від a+1 до n включно
    {
        c = sqrt((long) a * a + b * b);
        if(c == int(c) && (c <= n))
        {
            cout << "a = " << a << " b = " << b << " c = " << c << "\n"; /* Якщо  $a^2 + b^2 = c^2$ ,
            тобто корінь з  $a^2 + b^2$  є цілим числом, то виведемо ці значення */
        }
    }
}

```

main

```

/Users/kyryl/Desktop/Lab5_C++/cmake-build-debug/Lab5_C_
Enter natural number n: 30
a = 3 b = 4 c = 5
a = 5 b = 12 c = 13
a = 6 b = 8 c = 10
a = 7 b = 24 c = 25
a = 8 b = 15 c = 17
a = 9 b = 12 c = 15
a = 10 b = 24 c = 26
a = 12 b = 16 c = 20
a = 15 b = 20 c = 25
a = 18 b = 24 c = 30
a = 20 b = 21 c = 29

```

```
15 c = sqrt(lcpp_x: a * a + b * b);
16 if(c == int(c) && (c <= n))
17
2 # Отримати всі піфагорові трійки натуральних чисел, кожне з яких не перевищує n, тобто всі такі
3 # трійки натуральних чисел a, b, c, що a^2 + b^2 = c^2 (a<=n, b<=n, c<=n).
4 import math
5 import time
6 n = int(input("Enter natural number n: ")) # Введення натурального числа n
7 for a in range(1, n+1): # Цикл для числа a, де ітерація йде від 1 до n включно
8     for b in range(a+1, n+1): # Вкладений цикл для числа b, де ітерація йде від a+1 до n включно
9         c = math.sqrt(a ** 2 + b ** 2)
10        if c == int(c) and c <= n: # Якщо a^2 + b^2 = c^2, тобто корінь з a^2 + b^2 з цілим числом,
11            # то виведемо ці значення
12            c = int(c)
13            print(f'a = {a}, b = {b}, c = {c}')
14
15 for a in range(1, n+1)    for b in range(a+1, n+1)    if c == int(c) and c <= n
Run: main
Enter natural number n: 30
a = 3, b = 4, c = 5
a = 5, b = 12, c = 13
a = 6, b = 8, c = 10
a = 7, b = 24, c = 25
a = 8, b = 15, c = 17
a = 9, b = 12, c = 15
a = 10, b = 24, c = 26
a = 12, b = 16, c = 20
a = 15, b = 20, c = 25
a = 18, b = 24, c = 30
a = 20, b = 21, c = 29
Process finished with exit code 0
```

## Результат на Python:

```
2 # Отримати всі піфагорові трійки натуральних чисел, кожне з яких не перевищує n, тобто всі такі
3 # трійки натуральних чисел a, b, c, що a^2 + b^2 = c^2 (a<=n, b<=n, c<=n).
4 import math
5 import time
6 n = int(input("Enter natural number n: ")) # Введення натурального числа n
7 for a in range(1, n+1): # Цикл для числа a, де ітерація йде від 1 до n включно
8     for b in range(a+1, n+1): # Вкладений цикл для числа b, де ітерація йде від a+1 до n включно
9         c = math.sqrt(a ** 2 + b ** 2)
10        if c == int(c) and c <= n: # Якщо a^2 + b^2 = c^2, тобто корінь з a^2 + b^2 з цілим числом,
11            # то виведемо ці значення
12            c = int(c)
13            print(f'a = {a}, b = {b}, c = {c}')
14
15 for a in range(1, n+1)    for b in range(a+1, n+1)    if c == int(c) and c <= n
Run: main
"/Users/kyryl/Desktop/Кирилл Сидак/Lab5/venv/bin/python" "/Users/kyryl/Desktop/Кирилл Сидак/Lab5/main.py"
Enter natural number n: 25
a = 3, b = 4, c = 5
a = 5, b = 12, c = 13
a = 6, b = 8, c = 10
a = 7, b = 24, c = 25
a = 8, b = 15, c = 17
a = 9, b = 12, c = 15
a = 12, b = 16, c = 20
a = 15, b = 20, c = 25
Process finished with exit code 0
```

## Висновок

Отже, я дослідив особливості організації складних цикл, створивши алгоритм для пошуку піфагорових трійок чисел, використавши вкладений арифметичний цикл.