# 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

«Основи програмування-1.

Базові конструкції»

«Організація підпрограм»

Варіант 19

Виконав студент ІП-11 Лисенко Андрій Юрійович

(шифр, прізвище, ім'я, по батькові)

Перевірив Вітковська Ірина Іванівна

( прізвище, ім'я, по батькові)

Київ 2021

**Лабораторна робота№ 6**

**Організація підпрограм**

Мета –набути навичок складання і використання підпрограм користувача

**Індивідуальне завдання**

**Варіант 19**

Два натуралних числа називаються дружніми, якщо кожне з них дорівнює сумі дільників іншого, крім самого цього числа. Надрукувати всі пари дружних чисел, які не перевищують задане число *n.*

**Постановка задачі**

Для розв'язання цієї задачі використаємо підпрограму *Sum\_divisors* з цілочисельним фактичним параметром *k* для знаходженнясуми дільників цього числа. Функція повертатиме цілочисельне значення *S\_div*. В основній функції введемо два арифметичних цикла для знаходження усіх варіантів не впорядкованих пар чисел менших за *n* та надрукуємо їх тільки якщо вони відповідають умові задачі. Також використаємо дві функції типу *void* для введення та виведення данних.

**Побудова математичної моделі**

Складемо таблицю змінних

|  |  |  |  |
| --- | --- | --- | --- |
| Змінна | Тип | Ім’я | Призначення |
| *num1, num2* | int | Лічильники | Зберігати інформацію про ітерацію арифметичного циклу |
| *n* | int | Границя обчислення | Зберігати інформацію про границю обчислення |
| S\_div | int | Сума дільників | Зберігати інформацію про суму дільників числа *k* |

**Псевдокод**

**Основна програма:**

**Початок**

*init();*

Повторити для *num1* від 1 до *n*:

Повторити для *num2* від *num1* до *n*:

**Якщо** *Sum\_divisors(num1) == num2 && Sum\_divisors(num2) == num1* **то:**

*browse();*

**Все якщо**

**Все повторити**

**Все повторити**

**Кінець**

**Підпрограма *init():***

**Початок *init()***

**Введення** *n;*

**Кінець *init()***

**Підпрограма *browse():***

**Початок *browse()***

**Виведення** *num1, num2***;**

**Кінець *browse()***

**Підпрограма *Sum\_divisors(k):***

**Початок *Sum\_divisors(k)***

*S\_div = 0;*

**Повторити для** *div* **від** 1 **до** *k / 2***:**

**Якщо** *k % div == 0* **то:**

*S\_div += div;*

**Все якщо**

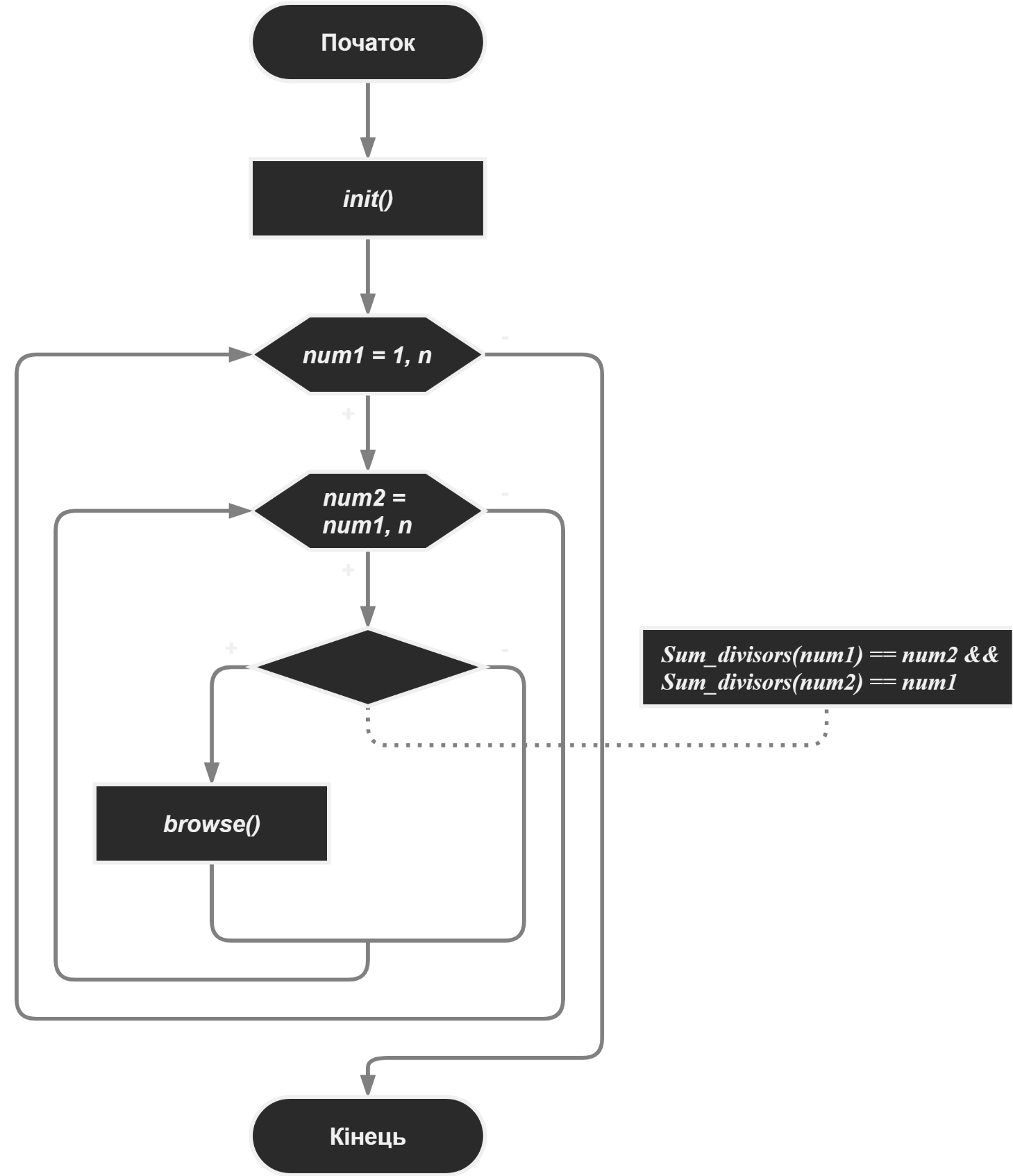
**Все повторити**

**Повернути** *S-div*

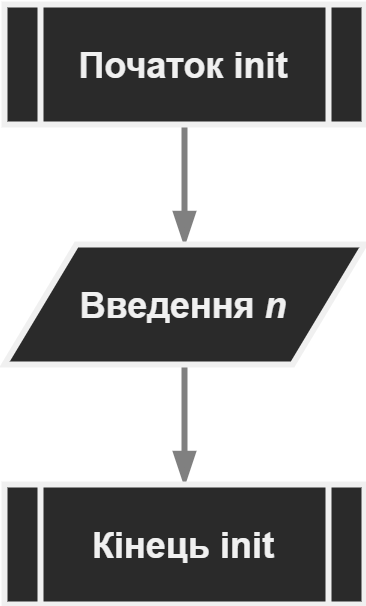
**Кінець *Sum\_divisors(k)***

**Блок-схема**

**Основна програма:**

****

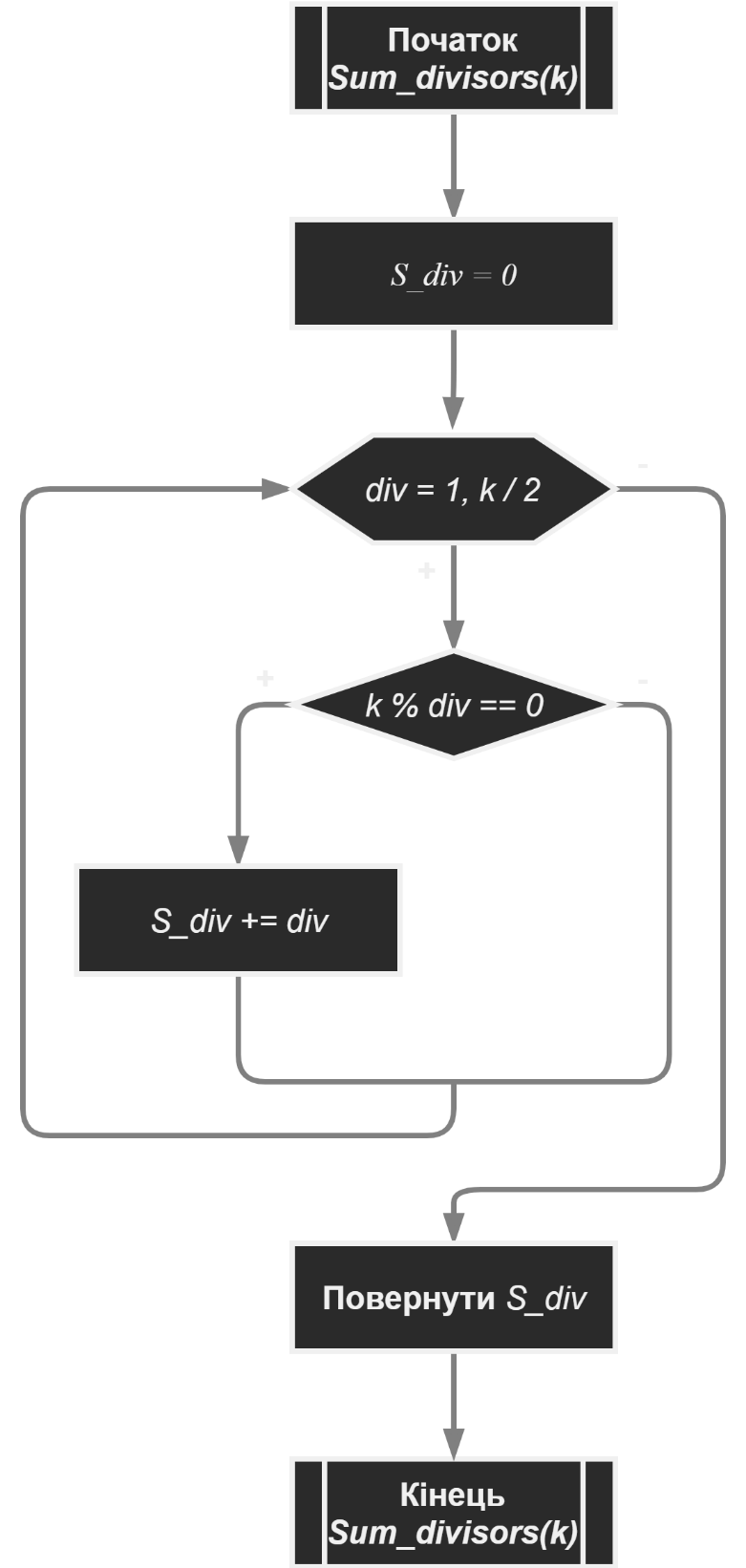
**Підпрограма *init()*:**

****

**Підпрограма *browse()*:**

****

**Підпрограма *Sum\_divisors(k)*:**

****

**Код на С++**

#include <iostream>

using namespace std;

int n;

int num1, num2;

int S\_div;

void init();

void browse();

int Sum\_divisors(int k);

int main() {

init();

for (num1 = 1; num1 <= n; num1++) {

for (num2 = num1; num2 <= n; num2++) {

if ((Sum\_divisors(num1) == num2) && (Sum\_divisors(num2) == num1)) {

browse();

}

}

}

system("pause>0");

}

void init() {

cout << "Enter the value of n: ";

cin >> n;

}

void browse() {

cout << "Numbers " << num1 << " and " << num2 << " are friendly" << endl;

}

int Sum\_divisors(int k) {

S\_div = 0;

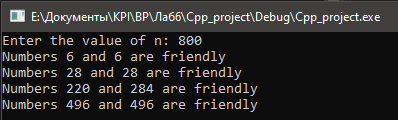
for (int div = 1; div <= k / 2; div++) {

if (k % div == 0) S\_div += div;

}

return S\_div;

}

****

**Код на Python**

def init():

n = int(input("Enter the value of n: "))

def browse():

print(f"Numbers {num1} and {num2} are friendly")

def Sum\_divisors(k):

S\_div = 0

for div in range(1, int(k / 2 + 1)):

if (k % div == 0): S\_div += div

return S\_div

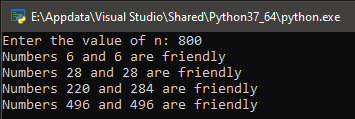
init()

for num1 in range(1, n):

for num2 in range(num1, n):

if ((Sum\_divisors(num1) == num2) and (Sum\_divisors(num2) == num1)): browse();

input()

****

**Висновок**

Ми використали підпрограму *Sum\_divisors* для знаходження суми дільників числа *k* та використали повернене значення *S\_div* для перевірки умови чи є два числа дружніми. Також ми використали програми *init()* та *browse()* для введення та виведення даних. Отже, ми набули навичок складання і використання підпрограм користувача.