

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
«Алгоритми та структури даних-1.
Основи алгоритмізації»

«Дослідження складних
циклічних алгоритмів»

Варіант 3

Виконав студент ПІ-15, Борисик Владислав Тарасович
(шифр, прізвище, ім'я, по батькові)

Перевірила Вєчерковська Анастасія Сергіївна
(прізвище, ім'я, по батькові)

Лабораторна робота №5

Дослідження складних циклічних алгоритмів

Мета – дослідити особливості роботи складних циклів та набути практичних навичок їх використання під час складання програмних специфікацій.

Варіант 3

Задача

Дано натуральне число n . Серед чисел $1, \dots, n$ знайти такі, запис яких співпадає з останніми цифрами запису їх квадрату. Наприклад, 6 ($6^2 = 36$), 25 ($25^2 = 625$) і т.д.

Постановка задачі

За умовою задачі потрібно знайти числа, запис яких співпадає з останніми цифрами запису їх квадрату.

Результатом розв'язку є значення цих чисел.

Побудова математичної моделі

Складемо таблицю змінних

Змінна	Тип	Ім'я	Призначення
Кількість чисел	Цілий	n	Початкове дане
Лічильник циклу	Цілий	i	Проміжне дане
Кількість цифр у числі	Цілий	numLength	Проміжне дане
Коефіцієнт	Цілий	k	Проміжне дане
Квадрат лічильника циклу	Цілий	square	Проміжне дане
Дільник	Цілий	divisor	Проміжне дане
Умова	Цілий	condition	Результат

Для цілочисельного ділення будемо використовувати оператор $/$.

Для остачі від ділення будемо використовувати оператор $\%$.

Для конкатенації рядків будемо використовувати оператор $+$.

Для піднесення числа до степеня будемо використовувати функцію `pow()`.

1) Створюємо змінну n .

- 2) Просимо користувача ввести кількість чисел (n) і присвоюємо це значення змінній n .
- 3) За допомогою арифметичного циклу ітеруємось через числа від 1 до n включно з кроком 1.
- 4) В тілі арифметичного циклу створюємо змінні $numLength$, k і присвоюємо їм значення $numLength = 0$, $k = i$ (i - лічильник циклу). Також, створюємо ітераційний цикл з передумовою.
- 5) В тілі ітераційного циклу з передумовою ділимо коефіцієнт на 10 і додаємо до кількості цифр одиницю поки здійснюється умова $k \geq 1$.
- 6) Після виходу з ітераційного циклу з передумовою створюємо змінні $square$, $divisor$, $condition$ і присвоюємо їм значення
 $square = \text{pow}(i, 2)$,
 $divisor = \text{pow}(10, numLength)$,
 $condition = square \% divisor$.
- 7) За допомогою умовної форми оператора вибору виводимо користувачу число, якщо воно задовільняє умову: $i == condition$.

Розв'язання

Програмні специфікації запишемо у псевдокоді та графічній формі у вигляді блок-схеми.

Крок 1. Визначимо основні дії

Крок 2. Створення змінної n .

Крок 3. Просимо користувача ввести кількість чисел (n) і присвоєння цього значення змінній n .

Крок 4. За допомогою арифметичного циклу ітеруємось через числа від 1 до n включно з кроком 1.

Крок 5. В тілі арифметичного циклу створюємо змінні $numLength$, k і присвоюємо їм значення $numLength = 0$, $k = i$ (i - лічильник циклу).

Крок 6. Також, створюємо ітераційний цикл з передумовою.

Крок 7. В тілі ітераційного циклу з передумовою ділимо коефіцієнт на 10 і

додаємо до кількості цифр одиницю поки здійснюється умова $k \geq 1$.

Крок 8. Після виходу з ітераційного циклу з передумовою створюємо змінні `square`, `divisor`, `condition` і присвоюємо їм значення

`square = pow(i, 2)`, `divisor = pow(10, numLength)`, `condition = square % divisor`.

Крок 9. За допомогою умовної форми оператора вибору виводимо користувачу число, якщо воно задовільняє умову: `i == condition`.

Псевдокод

Крок 1

Початок

Створення змінної `n`.

Введення `n`.

Ітеруємось через числа від 1 до `n` включно з кроком 1.

Створюємо змінні `numLength`, `k` і присвоюємо їм значення `numLength = 0`, `k = i`.

Створюємо ітераційний цикл з передумовою.

Ділимо змінну `k` на 10 і додаємо до `numLength` одиницю поки здійснюється умова $k \geq 1$.

Створюємо змінні `square`, `divisor`, `condition` і присвоюємо їм значення `square = pow(i, 2)`, `divisor = pow(10, numLength)`, `condition = square % divisor`.

Виводимо користувачу число, якщо воно задовільняє умову: `i == condition`.

Кінець

Крок 2

Початок

`n`

Введення `n`.

Ітеруємось через числа від 1 до `n` включно з кроком 1.

Створюємо змінні numLength, k і присвоюємо їм значення numLength = 0, k = i .

Створюємо ітераційний цикл з передумовою.

Ділимо змінну k на 10 і додаємо до numLength одиницю поки здійснюється умова $k \geq 1$.

Створюємо змінні square, divisor, condition і присвоюємо їм значення square = pow(i, 2), divisor = pow(10, numLength), condition = square % divisor.

Виводимо користувачу число, якщо воно задовільняє умову: i == condition.

Кінець

Крок 3

Початок

n

Введення n

Ітеруємось через числа від 1 до n включно з кроком 1.

Створюємо змінні numLength, k і присвоюємо їм значення numLength = 0, k = i .

Створюємо ітераційний цикл з передумовою.

Ділимо змінну k на 10 і додаємо до numLength одиницю поки здійснюється умова $k \geq 1$.

Створюємо змінні square, divisor, condition і присвоюємо їм значення square = pow(i, 2), divisor = pow(10, numLength), condition = square % divisor.

Виводимо користувачу число, якщо воно задовільняє умову: i == condition.

Кінець

Крок 4

Початок

n

Введення n

повторити

для i від 1 до n

Створюємо змінні numLength, k і присвоюємо їм значення numLength = 0,
k = i .

Створюємо ітераційний цикл з передумовою.

Ділимо змінну k на 10 і додаємо до numLength одиницю поки
здійснюється умова $k \geq 1$.

Створюємо змінні square, divisor, condition і присвоюємо їм значення
 $\text{square} = \text{pow}(i, 2)$, $\text{divisor} = \text{pow}(10, \text{numLength})$, $\text{condition} = \text{square} \% \text{divisor}$.

Виводимо користувачу число, якщо воно задовільняє умову: i
 $== \text{condition}$.

все повторити

Кінець

Крок 5

Початок

n

Введення n

повторити

для i від 1 до n

numLength := 0

k := i

Створюємо ітераційний цикл з передумовою.

Ділимо змінну k на 10 і додаємо до numLength одиницю поки
здійснюється умова $k \geq 1$.

Створюємо змінні square, divisor, condition і присвоюємо їм значення
 $\text{square} = \text{pow}(i, 2)$, $\text{divisor} = \text{pow}(10, \text{numLength})$, $\text{condition} = \text{square} \% \text{divisor}$.

Виводимо користувачу число, якщо воно задовільняє умову: i
 $== \text{condition}$.

все повторити

Кінець

Крок 6

Початок

n

Введення n

повторити

для i від 1 до n

numLength := 0

k := i

повторити

поки k >= 1

Ділимо змінну k на 10 і додаємо до numLength одиницю поки здійснюється умова k >= 1.

все повторити

Створюємо змінні square, divisor, condition і присвоюємо їм значення

square = pow(i, 2), divisor = pow(10, numLength), condition = square % divisor.

Виводимо користувачу число, якщо воно задовільняє умову: i

== condition.

все повторити

Кінець

Крок 7

Початок

n

Введення n

повторити

для i від 1 до n

numLength := 0

k := i

повторити

поки $k \geq 1$

$k := k / 10$

$numLength := numLength + 1$

все повторити

Створюємо змінні square, divisor, condition і присвоюємо їм значення
 $square = pow(i, 2)$, $divisor = pow(10, numLength)$, $condition = square \%$
 $divisor$.

Виводимо користувачу число, якщо воно задовільняє умову: i
 $== condition$.

все повторити

Кінець

Крок 8

Початок

n

Введення n

повторити

для i від 1 до n

$numLength := 0$

$k := i$

повторити

поки $k \geq 1$

$k := k / 10$

$numLength := numLength + 1$

все повторити

$square := pow(i, 2)$

$divisor := pow(10, numLength)$

$condition := square \% divisor$.

Виводимо користувачу число, якщо воно задовільняє умову: i
 $== condition$.

все повторити

Кінець

Крок 9

Початок

n

Введення n

повторити

для i від 1 до n

numLength := 0

k := i

повторити

поки k >= 1

k := k / 10

numLength := numLength + 1

все повторити

square := pow(i, 2)

divisor := pow(10, numLength)

condition := square % divisor.

якщо i == condition

то

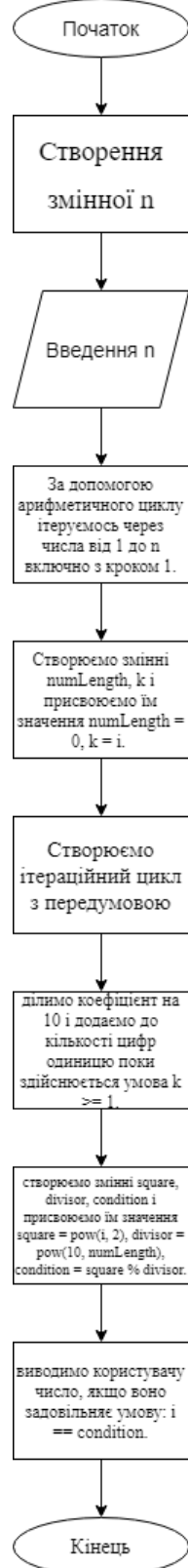
Вивід: i + " = " + square

все повторити

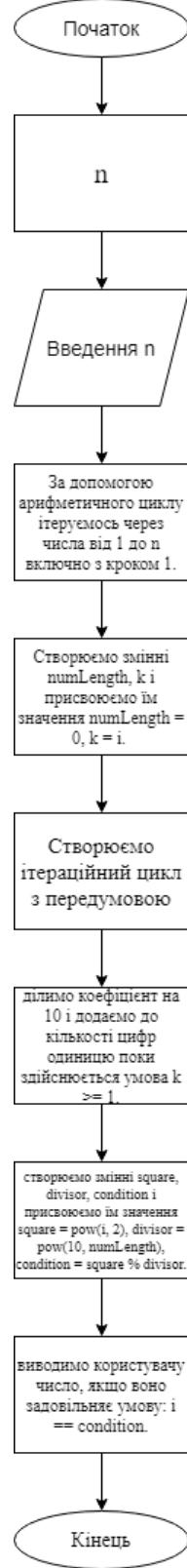
Кінець

Блок-схема алгоритму

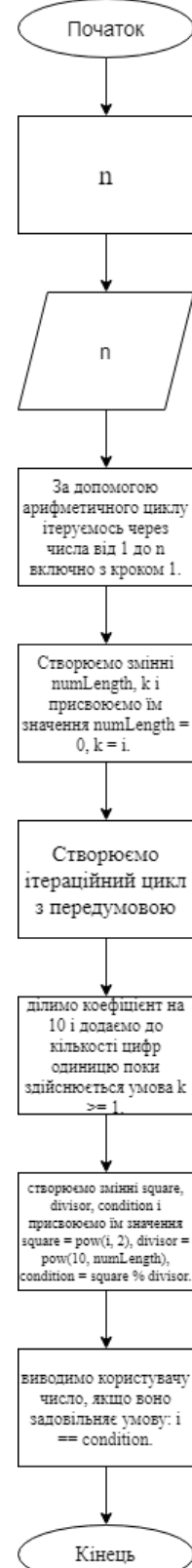
Крок 1



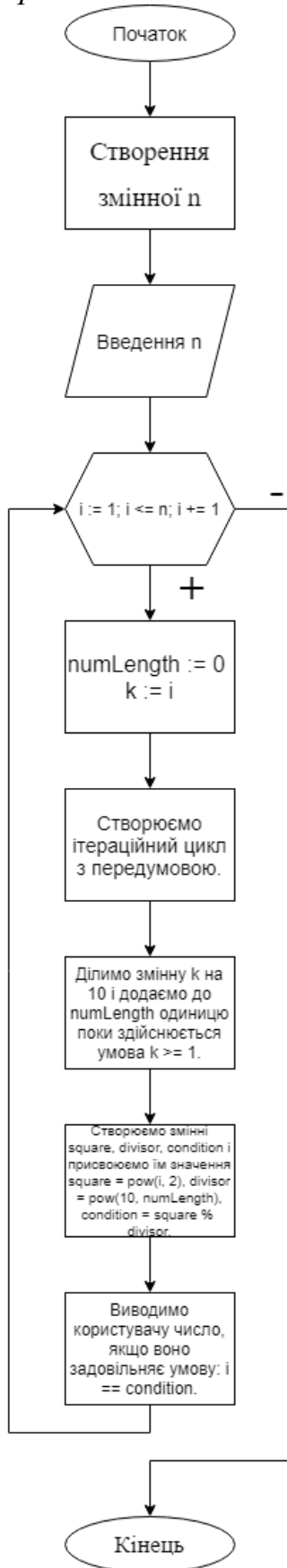
Крок 2



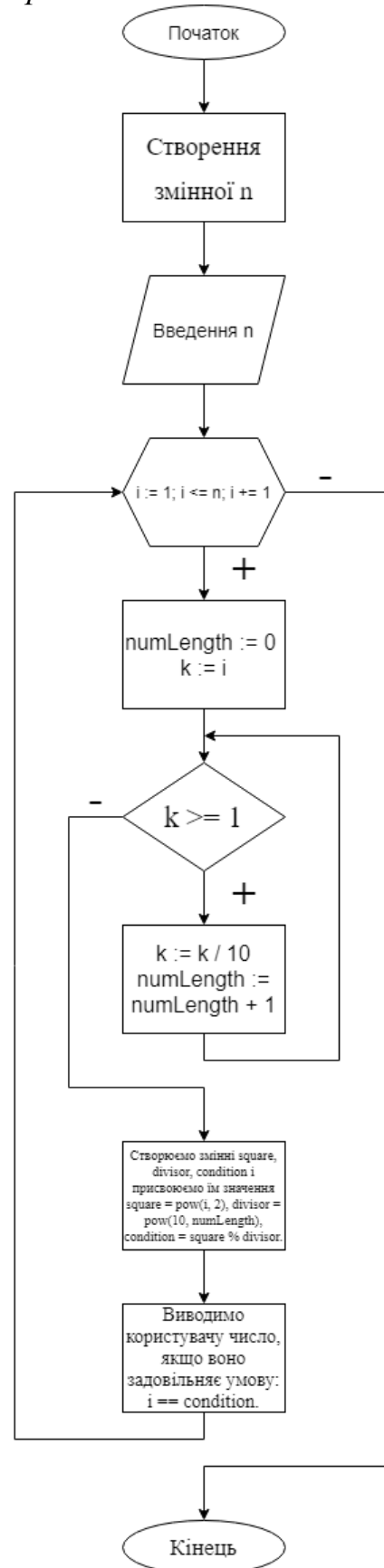
Крок 3



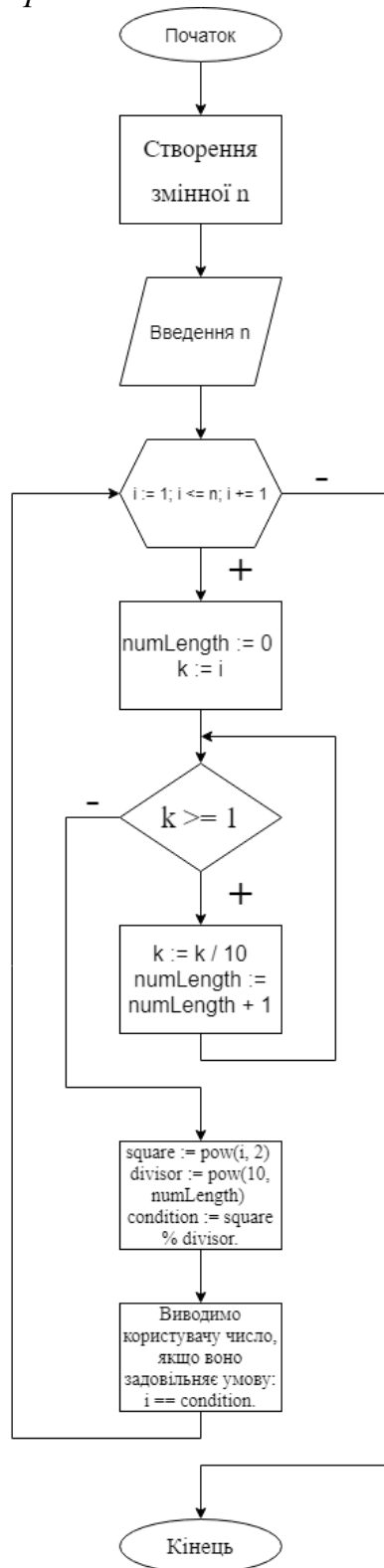
Крок 4



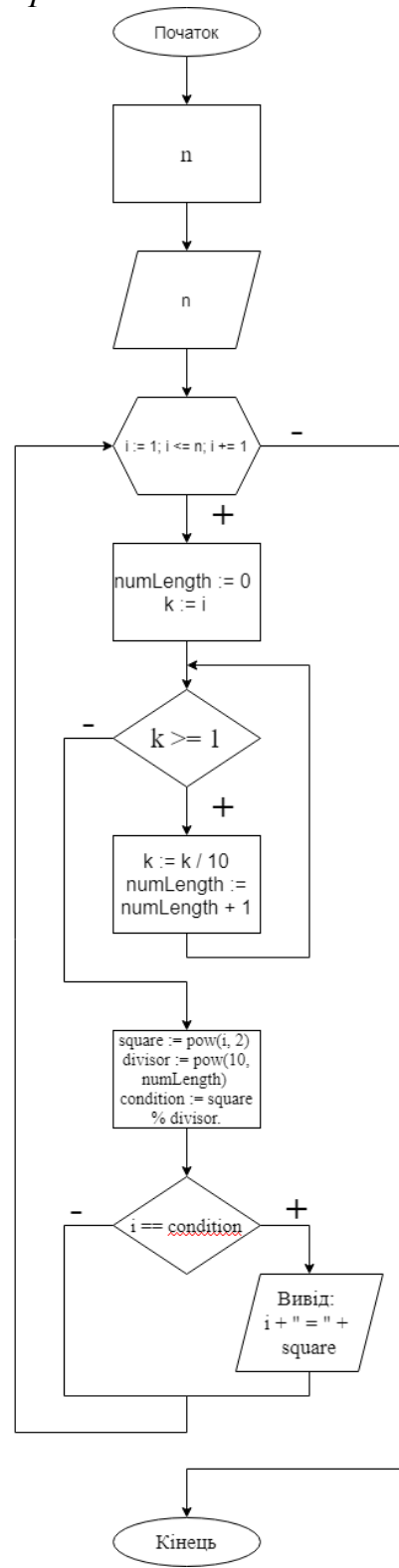
Крок 5



Крок 6



Крок 7



Випробування алгоритму

Блок	Дія
	Початок
1	Введення $n = 10000$
2	$i = 1$
3	$numLenght = 1$
4	$square = 1$
5	$divisor = 10$
6	$condition = 1$
7	$1 == 1$
8	Виведення: $1 = 1$
9	$i = 2$
10	$numLenght = 1$
11	$square = 4$
12	$divisor = 10$
13	$condition = 4$
14	$2 == 4$
15	$i = 3$
16	$numLenght = 1$
17	$square = 9$
18	$divisor = 10$
19	$condition = 9$
20	$3 == 9$
...	
30	$i = 6$
31	$numLenght = 1$
32	$square = 36$
33	$divisor = 10$
34	$condition = 6$

35	$6 == 36$
36	Виведення: $6 = 36$
...	
125	$i = 25$
126	$\text{numLenght} = 2$
127	$\text{square} = 625$
128	$\text{divisor} = 100$
129	$\text{condition} = 25$
130	$25 == 625$
131	Виведення: $25 = 625$
...	
380	$i = 76$
381	$\text{numLenght} = 2$
382	$\text{square} = 5776$
383	$\text{divisor} = 100$
384	$\text{condition} = 76$
385	$76 == 5776$
386	Виведення: $76 = 5776$
...	
1880	$i = 376$
1881	$\text{numLenght} = 3$
1882	$\text{square} = 141376$
1883	$\text{divisor} = 1000$
1884	$\text{condition} = 376$
1885	$376 == 141376$
1886	Виведення: $376 = 141376$
...	
3125	$i = 625$
3126	$\text{numLenght} = 3$

3127	square = 390625
3128	divisor = 1000
3129	condition = 625
3130	625 == 390625
3131	Виведення: 625 = 390625
...	
46880	i = 9376
46881	numLenght = 4
46882	square = 87909376
46883	divisor = 10000
46884	condition = 9376
46885	9376 == 87909376
46886	Виведення: 9376 = 87909376
	Кінець

Висновок

Протягом п'ятої лабораторної роботи я дослідив особливості роботи складних циклів та набув практичних навичок їх використання під час складання програмних специфікацій. В результаті виконання роботи я отримав алгоритм, який знаходить числа, запис яких співпадає з останніми цифрами запису їх квадрату.