

Додаток 1

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 № 3 з дисципліни
«Основи програмування 1. Базові конструкції»

«Організація циклічних процесів. Складні цикли»

Варіант №34

Виконав студент Щербацький Антон ІП-14
(шифр, прізвище, ім'я, по батькові)

Перевірив Камінська Поліна Анатоліївна
(прізвище, ім'я, по батькові)

Київ 2021

Лабораторна робота №5

Варіант 34

Мета: вивчити особливості організації складних циклів.

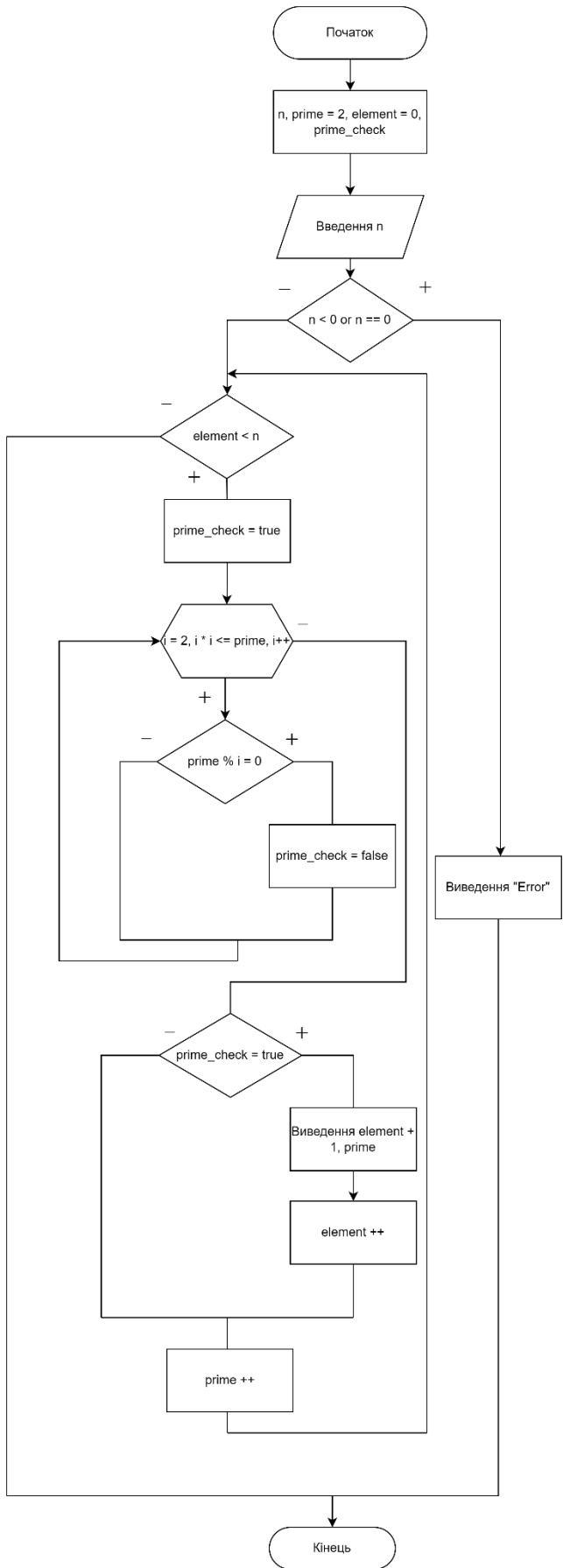
Хід роботи

Постановка задачі: визначити n перших простих чисел.

Математична модель

Змінна	Тип	Ім'я	Призначення
Кількість перших простих чисел	Натуральний	n	Вхідні дані
Просте число	Цілий	prime	Проміжні дані, результат
Перевірка на простоту числа	Булеван	Prime_check	Проміжні дані
Поточний елемент ітерації	Натуральний	element	Проміжні дані
Лічильник циклу	Натуральний	i	Проміжні дані

Блок схема



Кінець

Випробування коду:

Код(C++):

```
1  #include <iostream>
2  using namespace std;
3 ▶ int main()
4  {
5      int n, prime = 2, element = 0;
6      bool prime_check;
7      cout << "Enter n: " << endl; cin >> n;
8      if(n < 0 || n == 0)
9      {
10         cout << "Error, it's not allowed to use negative numbers or 0!" << endl;
11         return 0;
12     }
13     while (element < n)
14     {
15         prime_check = true;
16
17         for (int i = 2; i*i <= prime; i++)
18         {
19             if (prime % i == 0)
20                 prime_check = false;
21         }
22         if (prime_check == true)
23         {
24             cout << element + 1 << " element = " << prime << endl;
25             element++;
26         }
27         prime++;
28     }
29 }
```

Код(Python):

```

1  n = input("Enter n: ")
2  prime = 2
3  element = 0
4
5  if(int(n) < 0 or int(n) == 0):
6      print("Error, it's not allowed to use negative numbers or 0!")
7      exit(0)
8
9  while(int(element) < int(n)):
10     prime_check = True
11
12    for i in range(2, prime):
13        if(prime % i == 0):
14            prime_check = False
15
16    if(prime_check == True):
17        print(element + 1, (" element = "), prime)
18        element = element + 1
19
20    prime = prime + 1

```

Результат(C++) при n = 0

```

"D:\My projects\untitled\cmake-build-debug\untitled.exe"
Enter n:
0
Error, it's not allowed to use negative numbers or 0!

Process finished with exit code 0

```

n = -1000

```

"D:\My projects\untitled\cmake-build-debug\untitled.exe"
Enter n:
-1000
Error, it's not allowed to use negative numbers or 0!

Process finished with exit code 0

```

n = 10

```
"D:\My projects\untitled\cmake-build-debug\untitled.exe"
Enter n:
10
1 element = 2
2 element = 3
3 element = 5
4 element = 7
5 element = 11
6 element = 13
7 element = 17
8 element = 19
9 element = 23
10 element = 29

Process finished with exit code 0
```

n = 3

```
"D:\My projects\untitled\cmake-build-debug\untitled.exe"
Enter n:
3
1 element = 2
2 element = 3
3 element = 5

Process finished with exit code 0
```

При n = -1000

$-1000 < 0 = \text{true}$

Return 0

При n = 3

Prime_check = true

$3 < 0 = \text{false}$

1 ітерація

Prime_check = true

$2 * 2 = 4 < 2 = \text{false}$

Prime check = true

Виведення 2

Element = $0 + 1 = 1$

Prime = $2 + 1 = 3$

2 ітерація $1 < 3$

```
Prime_check = true  
2 * 2 = 4 < 3 = false  
Prime_check = true  
Виведення 3
```

```
Element = 1 + 1 = 2  
Prime = 3 + 1 = 4
```

3 ітерація

```
Prime_check = true  
2 * 2 = 4 <= 4 = true  
4 % 2 != 0
```

```
Prime_check = false  
Prime = 4 + 1 = 5
```

4 ітерація

```
Prime_check = true  
2*2 = 4 <= 5  
5 % 2 = 0
```

Виведення 5

Результат(Python) **X = -55**

```
"D:\My projects\pythonProject\venv\Scripts\python.exe" "D:/My projects/pythonProject/test.py"  
Enter n: -55  
Error, it's not allowed to use negative numbers or 0!  
  
Process finished with exit code 0
```

X = 25

```
"D:\My projects\pythonProject\venv\Scripts\python.exe" "D:/My projects/pythonProject/test.py"
Enter n: 25
1 element = 2
2 element = 3
3 element = 5
4 element = 7
5 element = 11
6 element = 13
7 element = 17
8 element = 19
9 element = 23
10 element = 29
11 element = 31
12 element = 37
13 element = 41
14 element = 43
15 element = 47
16 element = 53
17 element = 59
18 element = 61
19 element = 67
20 element = 71
21 element = 73
22 element = 79
23 element = 83
24 element = 89
25 element = 97

Process finished with exit code 0
```

Висновок

На цій лабораторній роботі було досліджено та набуто навичок подадання операторів повторення, а саме складних циклів. Також побудовано математичну модель, блок-схему. В результаті виконання цієї роботи було досліджено та використано складний ітераційний та арифметичний цикли. У результаті було отримано перелік перших n простих чисел.