

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І
ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

Кафедра комп'ютерних наук

ЛАБОРАТОРНА РОБОТА №1

Виконав:
студент групи ІПЗ-210076
Щур В. М.

Завдання: Створити віртуальне оточення (ім'я оточення - прізвище студента). В цьому оточенні створити наступні програми на мові Python(в різних файлах)

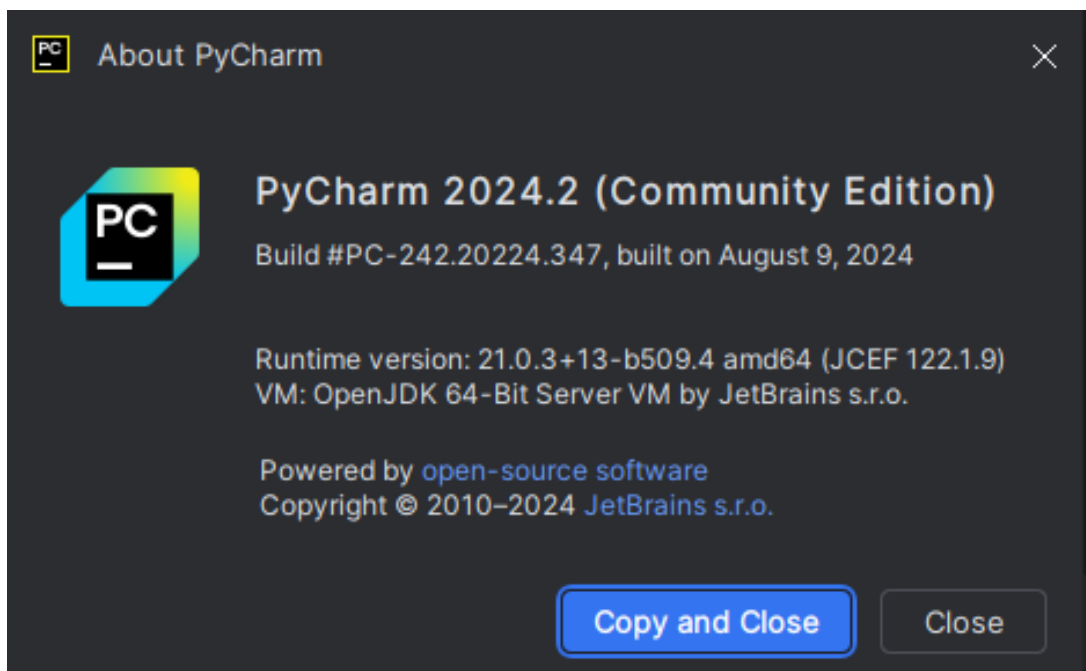
1. Програма для виводу всіх простих чисел, які розташовані між числами a та b , які задає користувач (не обов'язково $a < b$).
2. Створити рекурсивну функцію `fib(k)`, яка повертає список n чисел Фібоначчі. Написати програму яка виводить k чисел Фібоначчі, використовуючи функцію `fib(k)`.

Хід виконання

1. Скріншот терміналу з виводом версії Python

```
(Shchur) PS D:\PythonLab\lab1> python --version  
Python 3.12.4
```

2. Скріншот вікна About програми PyCharm (або Visual Studio Code)



3. Коди програм на мові Python

Програма №1:

```
# Функція для перевірки, чи є число простим
def is_simple_number(num):
    if num <= 1:
        return False
    if num == 2:
        return True
    if num % 2 == 0:
        return False
    for divisor in range(3, int(num**0.5) + 1, 2):
        if num % divisor == 0:
            return False
    return True

first_num = int(input("Введіть перше число: "))
second_num = int(input("Введіть друге число: "))

lower_bound = min(first_num, second_num)
upper_bound = max(first_num, second_num)

# Збір всіх простих чисел у списку
prime_numbers = [number for number in range(lower_bound, upper_bound + 1) if
is_simple_number(number)]

print(f"Прості числа в інтервалі [{lower_bound}, {upper_bound}]:")
print(" ".join(map(str, prime_numbers)))
```

Програма №2:

```
# Рекурсивна функція для генерації списку з n чисел Фібоначчі
def fibonacci_sequence(n):
    if n == 0:
        return []
    elif n == 1:
        return [0]
    elif n == 2:
        return [0, 1]
    else:
        sequence = fibonacci_sequence(n - 1)
        sequence.append(sequence[-1] + sequence[-2])
        return sequence

count = int(input("Введіть кількість чисел Фібоначчі: "))

# Виклик функції для отримання n чисел Фібоначчі
fibonacci_numbers = fibonacci_sequence(count)

print(f"{count} чисел Фібоначчі:")
print(" ".join(map(str, fibonacci_numbers)))
```

4. Скріншоти виконання програм в системному терміналі і в середовищі PyCharm. (на скріншоті обов'язково повинно бути видно, що програма виконується в віртуальному оточенні, яке було створенно)

Скріншоти виконання програм в системному терміналі:

```
D:\PythonLab\lab1>Shchur\Scripts\activate

(Shchur) D:\PythonLab\lab1>python program1.py
Введіть перше число: 100
Введіть друге число: 1
Прості числа в інтервалі [1, 100]:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

(Shchur) D:\PythonLab\lab1>python program2.py
Введіть кількість чисел Фібоначчі: 15
15 чисел Фібоначчі:
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377
```

Скріншоти виконання програм в середовищі PyCharm:

```
PS D:\PythonLab\lab1> Shchur\Scripts\activate
(Shchur) PS D:\PythonLab\lab1> python program1.py
Введіть перше число: 100
Введіть друге число: 1
Прості числа в інтервалі [1, 100]:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97
(Shchur) PS D:\PythonLab\lab1> python program2.py
Введіть кількість чисел Фібоначчі: 15
15 чисел Фібоначчі:
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377
```