

# Prime Number Generator and Checker Report

## Title Page

**Project Name:** Prime Number Generator and Checker

**Author:** Sujal Kumar

**Date:** 10/03/2025

---

## Introduction

This report presents an overview of the **Prime Number Generator and Checker**, a Python-based program designed to efficiently determine prime numbers and generate a list of primes within a specified range. The program provides users with two functionalities: checking if a given number is prime and generating all prime numbers up to a given limit.

## Methodology

### **Prime Checking Algorithm**

The program uses an optimized **square root method** for checking prime numbers. The function is `is_prime(n)`:

- Returns False for numbers less than 2 (since primes start from 2).
- Iterates from 2 to  $\sqrt{n}$  and checks divisibility.
- Returns True if no divisors are found, confirming the number is prime.

### **Prime Number Generation**

The `generate_primes(limit)` function:

- Iterates through numbers from 2 to limit.
- Uses the `is_prime()` function to check each number.
- Stores all prime numbers in a list and returns it to the user.

## **User Interaction**

- The program prompts the user to choose between:
  - Generating prime numbers up to a limit.
  - Checking if a specific number is prime.
- Based on user input, the appropriate function is executed, and results are displayed.

## Code Typed

```
def is_prime(n):
    # Check if a number is prime
    if n < 2:
        return False
    for i in range(2, int(n ** 0.5) + 1): # Loop from 2 to sqrt(n)
        if n % i == 0:
            return False # If divisible, not a prime
    return True # If no divisors found, it's a prime

def generate_primes(limit):
    # Generate prime numbers up to a given limit
    primes = []
    for num in range(2, limit + 1):
        if is_prime(num):
            primes.append(num) # Add prime number to the list
    return primes

if __name__ == "__main__":
    # Ask user for input choice
    choice = input("Do you want to (1) generate prime numbers or (2) check if a number is prime? Enter 1 or 2: ")

    if choice == "1":
        limit = int(input("Enter the limit: ")) # Get limit from user
        print("Prime numbers:", generate_primes(limit)) # Print prime numbers
    elif choice == "2":
        num = int(input("Enter a number: ")) # Get number from user
        if is_prime(num):
            print(f"{num} is a prime number.")
        else:
            print(f"{num} is not a prime number.")
    else:
        print("Invalid choice. Please enter 1 or 2.") # Handle invalid input
```

## Screenshots Output Photo Pasted

```
Do you want to (1) generate prime numbers or (2) check if a number is prime? Enter 1 or 2: 1
Enter the limit: 7
Prime numbers: [2, 3, 5, 7]
```

```
Do you want to (1) generate prime numbers or (2) check if a number is prime? Enter 1 or 2: 2
Enter a number: 25
25 is not a prime number.
```

```
Do you want to (1) generate prime numbers or (2) check if a number is prime? Enter 1 or 2: 0
Invalid choice. Please enter 1 or 2.
```

## **Conclusion**

The **Prime Number Generator and Checker** is a simple yet effective Python program for working with prime numbers. By using an optimized algorithm, the program ensures quick and accurate results. This tool can be further enhanced by implementing additional features such as checking a range of numbers or optimizing the prime number generation process using the Sieve of Eratosthenes.

## **Future Enhancements**

- Implementing the **Sieve of Eratosthenes** for faster prime number generation.
- Adding a **graphical user interface (GUI)** for better user interaction.
- Providing an option to check prime numbers in **large datasets**.

## **References**

- Basic number theory concepts.
- Python programming best practices.