

Lab 3.3.4 Prime numbers – how do we find them?

Objectives

- familiarize the student with classic notions and algorithms;
- improve the student's skills in defining and using functions.

Scenario

*A natural number is **prime** if it is greater than 1 and has no divisors other than 1 and itself.*

Complicated? Not at all. Look: 8 isn't a prime number, as you can divide it by 2 and 4 (we can't use divisors equal to 1 and 8 as the definition prohibits this). On the other hand, 7 is a prime number, as we can't find any legal divisors for it. Your task is to write a function checking whether a number is prime or not.

The function:

- is called `IsPrime`;
- takes one argument (the value to check)
- returns `True` if the argument is a prime number, and `False` otherwise.

Hint: try to divide the argument by all subsequent values (starting from 2) and check the remainder – if it's zero, your number cannot be a prime; think carefully about when you should stop the process.

If you need to know the square root of any value, you can utilize the `**` operator. Remember: the square root of x is the same as $x^{0.5}$.

Complete the code presented below.

Run your code and check whether your output is the same as ours.

```
def IsPrime(num):  
    #  
    # put your code here  
    #  
    for i in range(20):  
        if IsPrime(i + 1):  
            print(i, end=" ")  
  
print()
```

Example output

2 3 5 7 11 13 17 19