1 Background

Primes are the building blocks of numbers, indeed the Fundamental Theorem of Arithmetic states that every natural number larger than 1 can be written as the product of two or more prime numbers. For example

$$520 = 2 \cdot 2 \cdot 2 \cdot 5 \cdot 13 = 2^3 \cdot 5 \cdot 13$$

In this session we will study some of the properties of prime numbers

2 Problems

(P1) Create a function that decides whether a number is a prime. The function should return 0 if the number is not prime and 1 otherwise

```
def isprime(x)
```

(P2) *Goldbach's conjecture: every even integer greater than 2 can be written as the sum of two primes*

Write a function that given an even number decides whether Goldbach's conjecture is true. It should return 1 if the conjecture is true and 0 otherwise

(P3) *Legendre's conjecture: for every positive integer n there is a prime number between n^2 and $(n+1)^2$*

Write a function that given an positive integer decides whether Legendre's conjecture is true. It should return 1 if the conjecture is true and 0 otherwise

(P4) *Waring's conjecture: every odd integer exceeding 3 is either a prime number or the sum of three prime numbers*

Write a function that given an positive integer larger than 3 decides whether Waring's conjecture is true. It should return the number if it is prime, the three prime integers that add up to the number or 0 if the conjecture is false.

(P5) **Advanced Problem:** Design an algorithm to factor an integer into its prime factors. (hint: use Euler's factorization method)

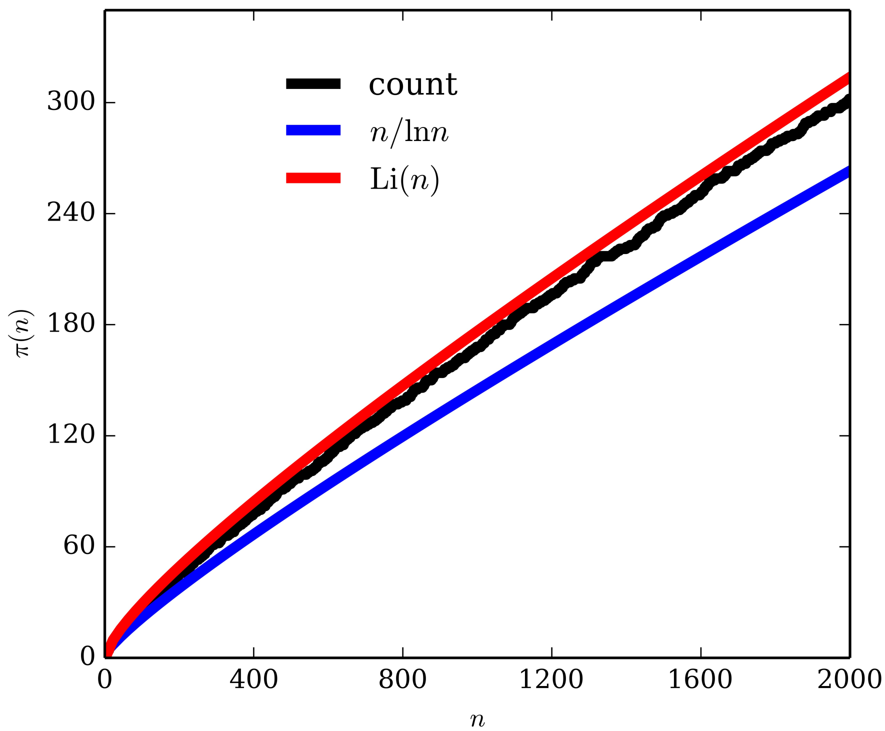
(P6) **Advanced Problem:** The prime counting function $\pi(n)$ is defined as the number of primes not greater than n. Make a plot of π for $n \leq 2000$ and overplot the approximations

$$\pi(x) \approx \frac{n}{\ln n} \tag{1}$$

and

$$\pi(x) \approx \mathrm{Li(n)} \tag{2}$$

(hint: The function Li(x) is contained in the package `mpmath`)


count
$n/\ln n$
Li(n)
$\pi(n)$
0
60
120
180
240
300
0
400
800
1200
1600
2000
n