

# Assignment 1

- Create a methods that counts (in parallel) a number of primes in a given range
  - `int countPrimes(long start, long stop)`

# Assignment 2

- define the Matrix and Vector classes to represent a matrix and vector
  - the type of matrix/vector entries is double
- create the methods
  - `Vector multiply(Matrix m, Vector v)`
  - `Matrix multiply(Matrix m, Matrix v)`
  - implement parallel multiplication
  - in a case of bad size of matrix/vector, the method throws an exception

# Assignment 1

- implement a parser of INI files
  - a method that obtains a path to the file and returns `HashMap<String, HashMap<String, String>>`
  - a INI file format
    - empty lines are ignored

```
[section1]
key1=val1
key2=val2
[section2]
key3=val3
key4=key4
```

# Assignment 2

- Create a class that
  - in the constructor, it obtains a path to a directory
  - it watches changes in the directory
    - ideally use the WatchService
  - it allows other objects to register on it and obtain notifications about changes in the directory
    - observer pattern
    - create your own suitable Observer interface

```
class DirWatcher {  
    public DirWatcher(Path p);  
    public void addObserver(Observer o);  
    protected void notifyObservers();  
}
```

Tests...

# Test 1

- Is it possible to declare the variable `i` so that the following cycle never terminates?

```
while (i != i + 0) {  
  
}
```

# Test 2

- Is it possible to declare the variables **i** and **j** so that the following cycle never terminates?

```
while (i <= j && j <= i && i != j) {  
  
}
```



Slides version P12.en.2020.01

This slides are licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/).