

```

import numpy
import re
import random
from scipy.stats import norm

def main():
    not_in = 0.05
    within = 0
    # reading of data, with regex :)
    rawdata = []
    ptrn = '([1-9][0-9]*.[0-9]*)$'
    lines = [line.strip() for line in open('tijden-medium.log')]
    for line in lines:
        searchResult = re.search(ptrn, line)
        if searchResult:
            if searchResult.group(1):
                rawdata.append(float(searchResult.group(1)))
        else:
            print 'line did not pass regex'

    for pla in range(100):
        # transform array to numpy array, for further numpy use.
        data = numpy.array(rawdata)
        data_mean = numpy.mean(data)

        # draw fifty random numbers from list.
        fifty_random = []
        for i in range(50):
            random_index = random.randrange(len(data))
            fifty_random.append(data[random_index])
            data = numpy.delete(data, random_index)

        fifty_random = numpy.array(fifty_random)

        random_mean = numpy.mean(fifty_random)
        random_sigma = numpy.sqrt(numpy.var(fifty_random))

        # definition of Cumulative distributive function of x, in this case:
        # 1-(a/2)
        z = norm.ppf(1.0 - (0.5 * not_in))
        lower = random_mean - (z * random_sigma /
                               numpy.sqrt(50))
        upper = random_mean + (z * random_sigma /

```

```
numpy.sqrt(50))
    if data_mean >= lower and data_mean <= upper:
        within += 1
    print within
if __name__ == '__main__':
    main()
```