



# 0. Програмиране с Python

Стефан Кънев  
Николай Бачийски  
Димитър Димитров  
Йоан Карадимов  
Алек Андреев



**За курса**

# За курса

## Цели на курса

**За курса**  
**Цели на курса**  
**Организационни**

**За курса**  
**Цели на курса**  
**Организационни**  
**Езикът Python**

**За курса**

**Цели на курса**

**Организационни**

**Езикът Python**

**Защо Python?**



**За курса**



Софийски университет „Св. Климент Охридски“  
ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА





$$D = \frac{1}{c} \frac{1}{l} \frac{dl}{dt} = \frac{1}{c} \frac{1}{P} \frac{dP}{dt}$$

$$D^2 = \frac{1}{P^2} \frac{P_0 - P}{P} \sim \frac{1}{P^2} \quad (1a)$$

$$D^2 = \frac{\kappa_0}{3} \frac{P_0 - P}{P} \sim \kappa_0 \quad (2a)$$

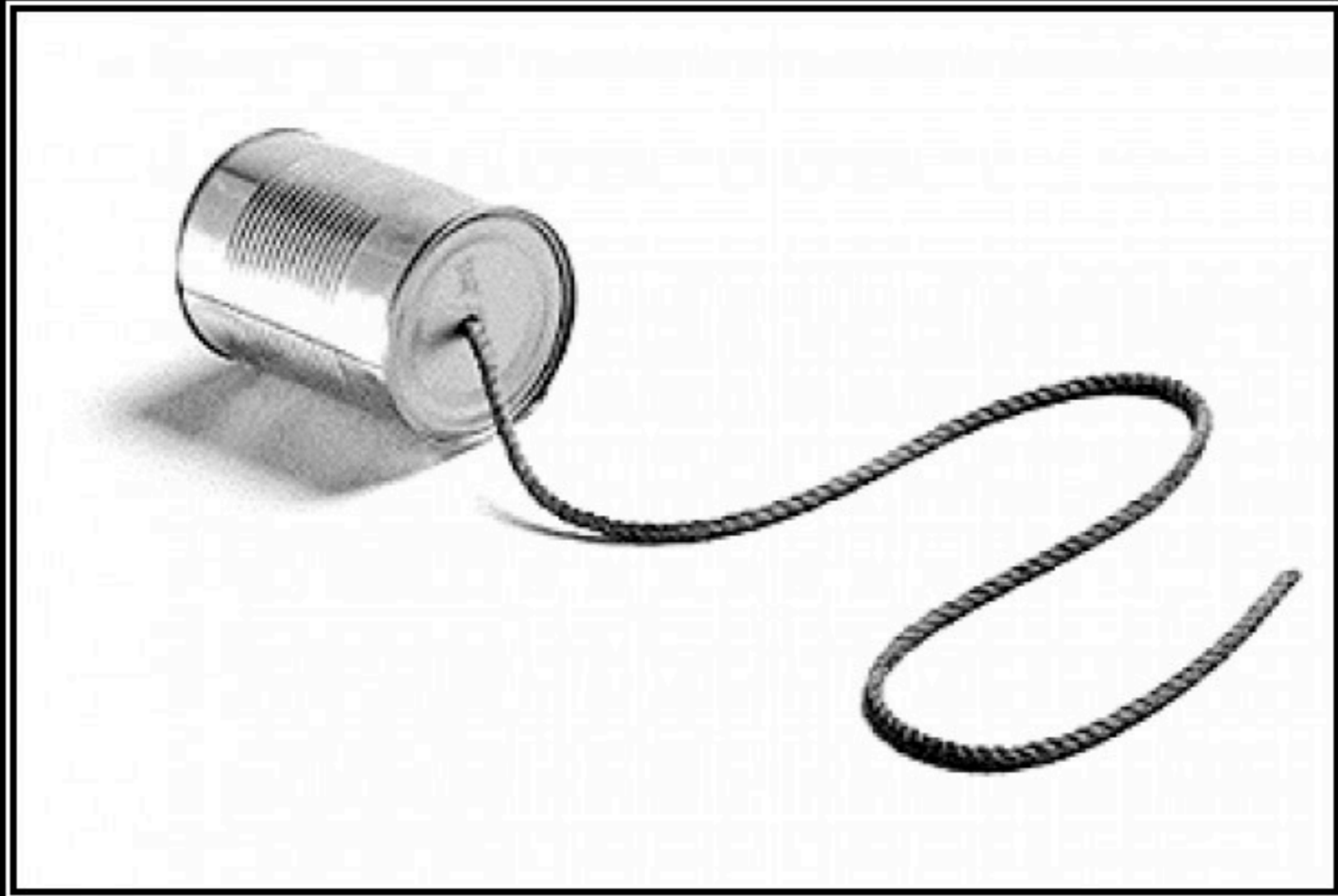
$$D^2 \sim 10^{-53}$$

$$\rho \sim 10^{-26}$$

$$P \sim 10^8 \text{ g. } \gamma$$

$$\pi \sim 10^{10} (10^{11}) \gamma$$



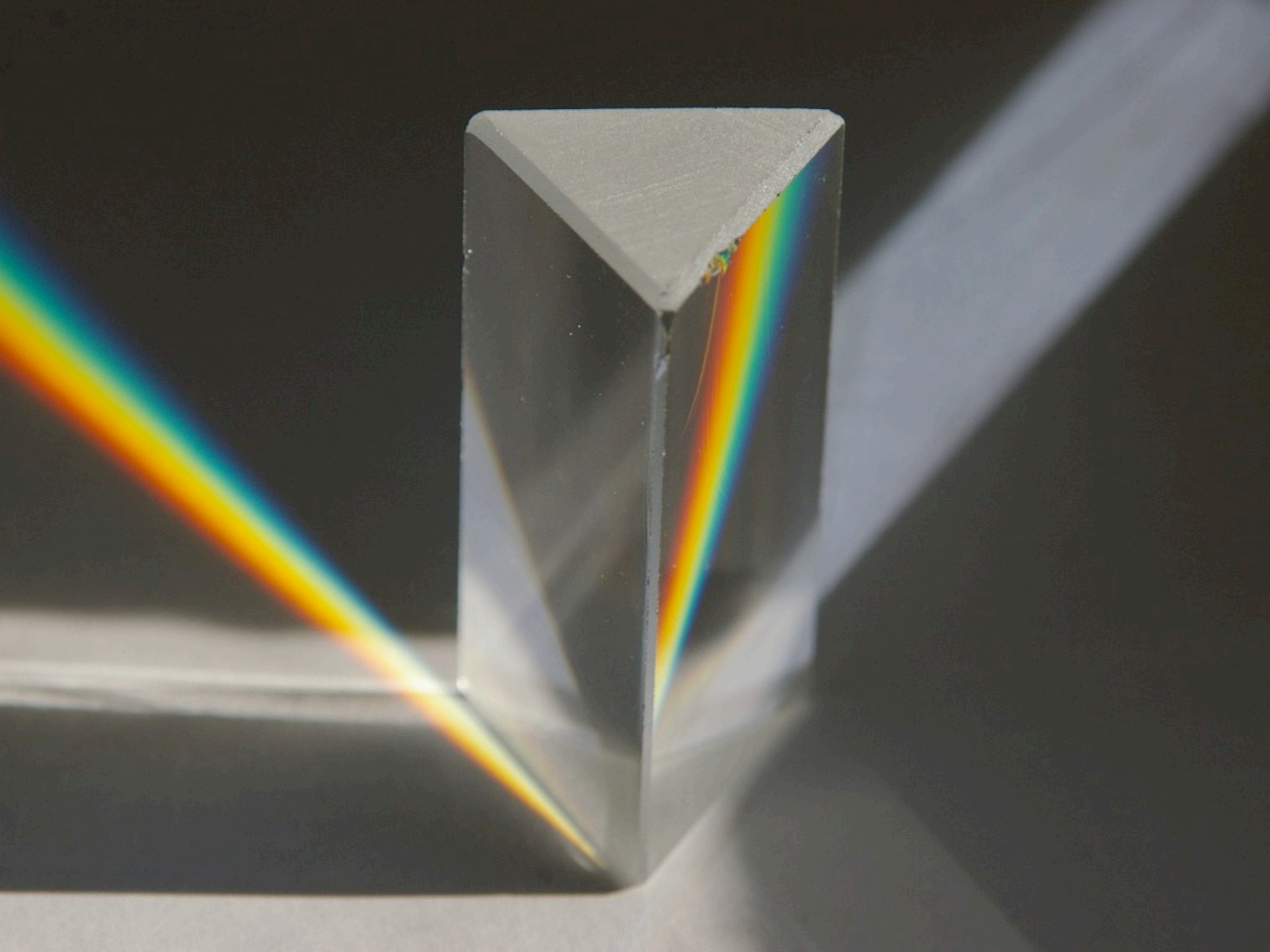


# COMMUNICATION

It's best to pretend people are actually listening to you.

# Цели на курса











**Какво можем да  
направим с един  
динамичен език**

**Функционалния  
стил в живия  
живот**

# Итератори и генератори

# **Автоматизирани тестове**





**Организаціонни**

**19–21 часа**

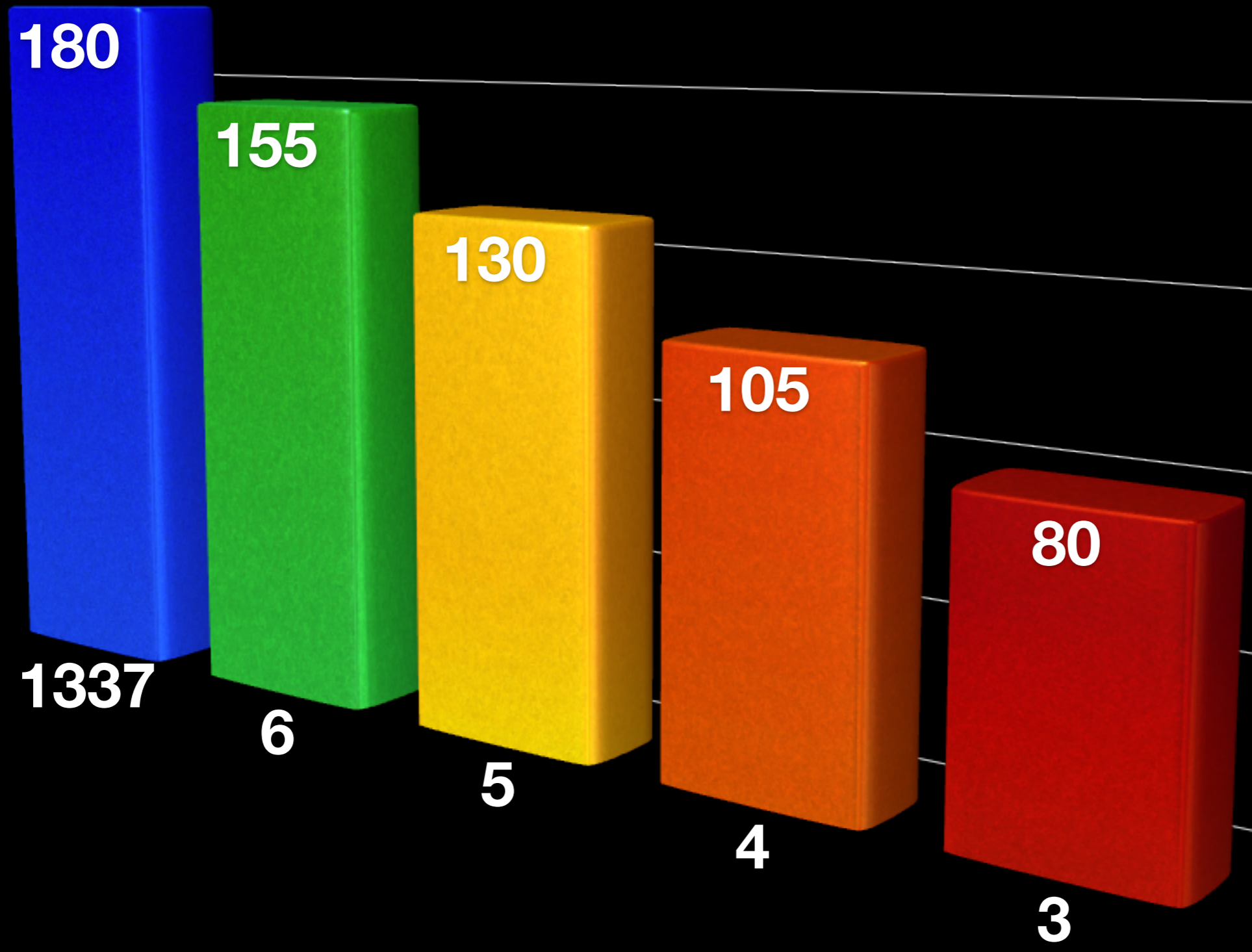
**понеделник – 325**

**четвъртък – 229**

**ФМИ**

<http://fmi.py-bg.net/>







**PIZZA**  
**TORONTO**







**ЕЗИКЪТ**

**Python**

**ПИТОН vs. пайтън**

oo 7

**Процедурният стил  
също е на почит**

**Интерпретируем**

**Динамичен**



# Динамично типизиран

## Duck typing





**Свободен!**





# Испанската инквизиция

(Не я очаквахте, нали?)



# Monty Python

**Защо  
Python?**

# Супер прост СИНТАКСИС

```
print("Hello, World!")

def answer(question):
    if question == 'Life' or question == 'Universe':
        return 42
    else:
        return 6 * 9 # сорок и две
```

# Интерактивен интерпретатор

```
nb@creep:~$ python
Python 2.5.1 (r251:54863, Jan 13 2009, 10:26:13)
>>> a = 4 + 12
>>> a * 42
672
```

# Култура на езика



# Култура на езика

- използване на интроспекция за по-малко код

# Култура на езика

- използване на интроспекция за по-малко код
- итератори/генератори

# Култура на езика

- използване на интроспекция за по-малко код
- итератори/генератори
- Test Driven Development

# Култура на езика

- използване на интроспекция за по-малко код
- итератори/генератори
- Test Driven Development
- функционален стил в нефункционалния Python

# Култура на езика

- използване на интроспекция за по-малко код
- итератори/генератори
- Test Driven Development
- функционален стил в нефункционалния Python
- документация още в изходния код

C/C++



**GvR**

**Guido van Rossum**

**BDFL**

Google™

NASA



SID MEIER'S  
CIVILIZATION  
IV

