Module

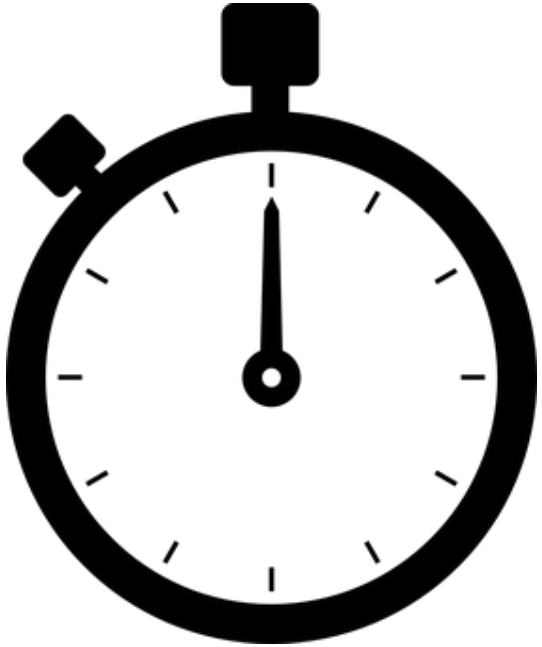
timeit

Introduction

Module:

The timeit Module

- **Purpose:**



The timeit Module

```
import timeit
```

Import statement

```
timeit.timeit("[i for i in range(10000)]", number=500)
```

timeit function

Code that you want to test

Number of repetitions

The timeit Module

```
>>> import timeit
>>> timeit.timeit("[i for i in range(10000)]", number=500)
0.30713721199999355
```

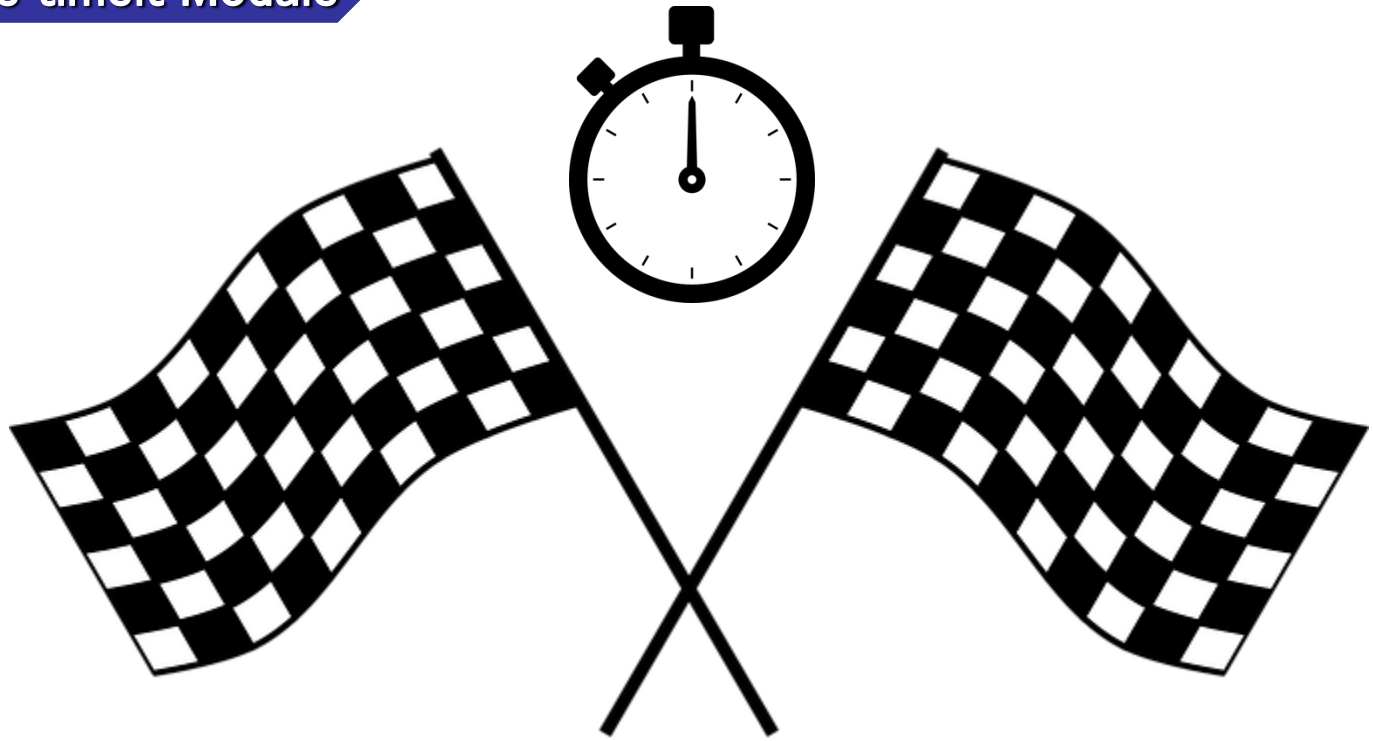
Measured time for *number* executions

The timeit Module

```
timeit.timeit(stmt='pass', setup='pass', timer=<default timer>, number=1000000, globals=None) ¶
```

- **Parameters:**
  - **stmt** (statement): code that will run (string).
  - **setup**: code that will be executed once before starting to measure statement.
  - **timer**: the timer function (use default).
  - **number**: number of repetitions.
  - **globals**: creates a namespace. You can pass current globals to use the functions that were defined in the file.

The timeit Module



The timeit Module

- Linear Search vs. Binary Search
- Bubble Sort – small and large lists
- Selection vs. Insertion Sort
- Insertion Sort vs. Merge Sort vs. Quicksort
- Analyze the Results
- Mini Project

The timeit Module

Let’s Begin!