In the name of Allah

How CPython Compiler Works

Mahdi Haghverdi

Isfahan University

Content

Overview

Overview

- Which steps does CPython takes to compile your source code?
- Why these steps?
- How they are done?

Diagram

- ```
  ------------------------------------------------------------
  | Decoding -> Tokenizing -> Parsing -> AST | -> Compiling |
  ------------------------------------------------------------
  ```
- Front-end: Decoding, Tokenizing, Parsing and AST
- Back-end: Compiling
  ```

Explanation

- We've got a front-end and a back-end part in this process.
- Front-end: getting down to the AST
- Back-end: to get the generated AST and compile it down to something
- Good example is PyPy which is a front-end for Python
- Ease of writing the code
- A better view to the process

Decoding - “Bytes” to “Text”

Tokenizing - ‘Text‘” to “Words”

Parsing - “Words” to “Sentence”

Abstract Systax Tree - “Sentence” to “Semantics”

Compiling - “Sematics” to “Bytecode”