

1. 프로그래밍의 목적

- 코딩, 프로그래밍, SW개발, 시스템 개발, 시스템 설계
- 컴퓨터에게 작업시키기 - 어떤 방법으로 이 사실을 전달할 것인가 ?
- 컴퓨터에서 “자연언어(Natural language)”로 말할 수는 없다.
- 인공지능 만들기는 궁극적으로는 “프로그래밍” 과정의 일부이다.

2. 프로그래밍의 언어의 계층 구조

- Source Code / 파일 1개 a.py f.c ttt.cpp
- Program 개인
- Software 팀 은행온라인, 비행기 조정
- System 온라인-항공사 등록-신용카드- 금융시스템, 국방

프로그래밍 언어의 구조

- Syntax $a = z + 10$ $a = z + 10 ;$ $a := z + 10 =$
compile error

- Semantics 의미

$a = a + 10$ a의 현재 값에 10을 더해서 그것을 a에 저장

$x = (4 < 7)$ 1, True, 0 (False)

3. 프로그래밍 언어의 종류

- Compiler 기반 언어 = { C, C++, GO, FORTRAN }
source code를 수행 항상 a.exe 로 바꿈.
>> a.exe (기계어 CPU가 이해하는 언어)

김치찌개 - > 김치, 돼지고기, 밥.....

섬유질, 물,,, 지방, 단백질...

위 <- 단백질, 탄수화물, 미네랄.....최하영양단위

- Interpreter 기반언어 = { Python, Javascript, BASIC... }
source code, 해석 다른 프로그램,
번역기 기계어를 만들어 앎고. Python,바로 결과가 나옴
ByteCode 번역기 < BYteCode < 기계어

my.py my.pyc bytecode ---> 결과

my.c C/C++ ---> my.exe --> CPU

4. 컴퓨터에서 작업시키기

- 어떤 방법으로 이 사실을 전달할 것인가 ?
- 문제 확인
- 프로그래밍 하기 (언어 선택)
- Editor로 code 작성 (x.py)
- Debugging 하기

5. 프로그래밍 언어의 선택 (Python, Rust.. 비행기

- 어떤 분야의 문제인가 ?
 - 문제마다 가장 적합한 언어가 존재한다.
 - * 속도
 - * 프로그래밍의 편의성 C/C++
- e.g.) PROLOG - 아름답지만...

- C++를 개발한 Bjarne Stroustrup의 말

<http://www.stroustrup.com/>

“세상에는 2가지 언어가 있다.

욕을 하면서, 매일 사용하는 언어, 그리고

아무도 사용하지 않는 언어“

6. 컴퓨터 과학, 공학 전공자가 반드시 익혀야 할 언어

TOIBE Index = <https://www.tiobe.com/tiobe-index/>

기초 언어	고급 언어	확대 언어
C	PHP	GO Swift Kotlin
C++	SQL	
JAVA	Haskell, ML (함수형)	
Python	Ruby	
JavaScript	C#	
MATLAB, R	Perl	

7. Why Python ?

- 쉽다
- 직관적
- 무료
- 다양한 사용자 그룹
- 빠른 Prototyping 구성하기에 유리함
- 최근 인공지능 기반 시스템의 필수언어

8. Python 2.x 3.x version

- 2020년부터 Python 3.0만 유지됨
- unicode 지원 (한글 사용에 필수적)

<https://www.python.org/>

<https://www.python-course.eu/>