

PL Homework 1 - 20191583 김태곤

#7. LISP가 순수 함수형 언어로 시작됐지만, 점차 더 많은 명령형 기능을 가지게 된 이유는?

LISP는 본래 인공지능에 쓰이는 수학적 표기법을 나타내기 위해 개발되었다. 그러나 대부분의 프로그래밍이 명령형 스타일로 이루어져 있어, 보다 더 효율적인 코드를 작성하기 위해 명령형 기능이 추가되었다.

#12 Programming language design의 직교성(orthogonality)에 대해 설명하라

직교성은 언어의 기능들이 서로 독립적이며 어떤 방식으로든 결합할 수 있음을 말한다. 예를 들어 C언어에서 + 연산자는 정수, 실수, 문자열 등 다양한 데이터 타입에 사용할 수 있다. 연산자 + 와 데이터 타입들 간의 직교성으로 인해 문제가 발생하지 않으며, 서로 독립적이고 의미 있는 결합을 할 수 있다.

#14 타입이 없는 언어에 대한 논쟁

타입이 없으면 코드의 양을 줄이고, 간결하고, 좀더 빠르게 코드를 작성시킬 수 있다.

타입이 없으니 다시 정의할 필요 없이 기존 코드를 더 빨리 수정할 수 있다.

하지만 type checking이 없어 다양한 데이터 타입을 사용할 때 실수하기 쉬우며

특정하기 힘든 버그가 발생할 수 있다. 또한, 다른 사용자가 코드를 보고 이해하기 힘들 수 있고,

표준화된 데이터 타입에서만 연산이 수행되도록 런타임 체크가 많이 일어나므로 정적 타입 언어에 비해 성능이 많이 떨어진다.