

프로그래밍 언어론

Assignment #1

전기컴퓨터공학부 정보컴퓨터공학전공

201524582 정희석

Q1. 알고 있는 프로그래밍언어와 그 언어의 특징, 사용 목적, 편리한 점, 불편한 점을 정리

1) C

- 특징: 절차형 언어, 명령형 언어, 구조화 프로그래밍
- 사용 목적: UNIX 운영체제, 임베디드 시스템, 응용프로그램 개발
- 편리한 점: 메모리 관리를 직접 할 수 있으며 저 레벨 언어인 어셈블리어와 함께 혼용이 가능해서 속달되면 동작속도와 메모리 효율을 높일 수 있다.
- 불편한 점: 절차형 언어의 특성상 구조가 복잡해지기 쉬우며 가독성이 떨어진다.

2) C++

- 특징: 절차형 언어, 함수형 언어, 객체지향 언어, 제네릭 프로그래밍 언어, 구조화 프로그래밍
- 사용 목적: C언어에서 객체지향 프로그래밍을 지원하기 위해 개발되었으며
- 편리한 점: 클래스를 이용하여 다중상속을 구현할 수 있고 캡슐화를 통해 코드 재사용을 가능케 함으로써 클래스 사용법만 알고 내부 구현을 자세히 알 필요가 없다.
- 불편한 점: C언어의 절차형 언어 특성을 이어받아서 구조가 복잡해지기 쉽다.

3) Java

- 특징: 객체지향 언어, 제네릭 프로그래밍 언어, 구조화 프로그래밍
- 사용 목적: 응용프로그램 개발, 안드로이드 어플리케이션 개발, 웹 응용프로그램 개발.
- 편리한 점: 다양한 라이브러리를 이용해 다양한 개발이 가능, 자바 가상 머신 위에서 동작하므로 여러 다양한 플랫폼(cross-platform)에서 사용이 가능한 프로그램을 만들기 쉽다.
- 불편한 점: C, C++처럼 컴파일을 하여 실행하는 아니라 인터프리터 방식으로 실행할 때마다 코드를 불러오므로 느리고 메모리를 직접 접근이 어렵다(garbage collector가 있음, 애초에 Java Virtual Machine에서 동작)

4) Python

- 특징: 함수형 언어, 명령형 언어, 객체지향 언어
- 사용 목적: 프로그래밍에 있어서의 코드 가독성을 높이기 위해 개발되었으며 이를 기반으로 다양한 어플리케이션이 개발, 최근에는 머신러닝(TensorFlow)의 기반언어로 사용된다.
- 편리한 점: 인간의 언어에 가까우며 줄 단위로 개별 실행이 가능하며(인터프리터 언어) 변수의 형식 지정이 자유롭다. 다양한 오픈소스 기반 라이브러리를 통해 생각보다 많은 유용한 기능들이 구현되어 있다.
- 불편한 점: 인터프리터 언어이며 여러 라이브러리들을 실행마다 import하기 때문에 메모리를 많이 차지하며 동작속도가 C, C++에 비해서 현저하게 느리다.

5) Javascript

- 특징: 절차형 언어, 함수형 언어, 이벤트 중심(event-driven) 언어
- 사용 목적: 웹 어플리케이션의 동작 구현(WWW의 핵심 기술)
- 편리한 점: 웹 어플리케이션의 거의 모든 기능의 근간이 되며 구현이 쉽다.
- 불편한 점: 디버깅하기가 어려우며 구현에 오류가 발생해도 무시하고 동작하기 때문에 오류를 찾기 어려움

6) C#:

- 특징: 명령형 언어, 객체지향 언어, 제네릭 프로그래밍 언어, 이벤트 중심(event-driven) 언어, 구조화 프로그래밍
- 사용 목적: 마이크로소프트사의 닷넷(.net)프레임워크를 위해 개발, 게임엔진 유니티 및 각종 어플리케이션 개발에 널리 사용
- 편리한 점: C++과 JAVA와 같이 객체지향성으로 클래스와 클래스 상속을 사용 가능하며 garbage collector 리소스를 제어함.
- 불편한 점: 클래스간 단일 상속만 가능, 메모리에 직접 접근이 어려움(unsafe 블록을 통해서 사용은 가능함)

Q2. Orthogonality가 C언어에서 발생시킬 수 있는 오류의 예

- Orthogonality란?

한국어로 직역하면 직교성을 뜻하며 서로 다른 기능이 함께 사용되어도 각각의 기능이 서로에게 영향을 주지 않는다는 뜻. 프로그래밍 언어론에서는 작은 단위의 구문(명령, 변수 등(primitive constructs) 집합을 가장 적게 조합하여 다른 구문과 자료구조를 조합하는 기능을 뜻 한다.¹

예시로 ADD명령어를 예시로 들어보면

ADD Operand1 Operand2에서 Operand1과 2에는 Register와 Address가 모두 들어갈 수 있는 것을 가리켜서 직교성을 가진다고 할 수 있다.

Orthogonality가 발생시킬 수 있는 c언어에서의 오류의 예시 중 첫째는 함수에 parameter로 인자를 전달할 때 정수, 실수형 변수, 포인터는 값을 전달하는데(call by value) 배열은 해당 배열의 값이 아닌 시작주소를 전달하는(call by reference) 부분에서 오류가 발생할 수 있고, 두번째는 switch-case문에서 c언어는 case에 정수형, enum, character형 변수만 사용 가능하고 character의 배열, 실수형 변수는 사용이 불가능하다는 점이 있다.

¹ "Orthogonality(Programming)",Wikipedia, Accessed Mar 28th,
[https://en.wikipedia.org/wiki/Orthogonality_\(programming\)](https://en.wikipedia.org/wiki/Orthogonality_(programming)),