20.03.24. 화. 소프트웨어 코딩

**1. 컴퓨팅적 사고 (Computational Thinking)**

백과사전에서 컴퓨팅적 사고의 의미

단편적인 학습에서 벗어나 복합적 사고로 나가는 수단으로, 창의적 문제를 해결하는 핵심 능력으로 주목받고 있다. 컴퓨터의 해결 능력인 데이터 수집 ⬝ 분석, 표현, 문제 분해 ⬝ 추상화, 자동화 등을 사고에 적용시켜 여러 분야에서 문제 해결을 하는데 사용한다.

* 거창하게 설명하고 있으나, 간단하게 설명하면, 현상을 컴퓨터가 이해하는 관점으로, 이해 가능하도록, 말 그대로 컴퓨터처럼 생각해보는 것.
* 컴퓨터가 작업을 수행하는 것처럼 생각해보는 것.
* 실제로 컴퓨터는 생각을 하는 고등 생물이 아니지만, 컴퓨터가 작업하는 방식처럼. 논리적이고 모든 것에 근거가 있는 사고를 해보는 것을 의미하는 것으로 보임.

한국사람은 한국어를 사용하고 미국사람은 영어를 사용한다. 한국사람의 사고방식과 영어권 사람들의 사고방식이 다른 것은 여러가지 요인이 있겠지만 기본적으로 언어적 차이에서 유발되는 부분도 있다. 생각이라는 것은 기본적으로 언어 형태로 이루어지기 때문이다. 따라서 해당 문화권의 언어를 이해하는 것은 해당 문화권 사람들의 사고방식을 이해하는 첫 걸음이라고 할 수 있다.

컴퓨터의 언어는 일반적으로 기계어이나, 기계어는 사람이 읽고 이해하기 어렵다. 따라서 일반적으로 컴퓨터의 언어라고 하면 사람(일반적으로 프로그래머 또는 개발자)과 컴퓨터 사이의 커뮤니케이션에 사용되는 언어인 프로그래밍 언어 (C, Java, Python 등)를 의미한다. 프로그래밍 언어는 컴퓨터 언어의 부분 집합이다. 프로그래밍 언어가 컴파일과 어셈블리를 통해 컴퓨터가 이해가능한 기계어인 일종의 비트 패턴으로 변환되고 컴퓨터는 이를 바탕으로 작업을 수행한다. 소프트웨어 프로그래머는 사람의 일상적인 명령을 컴퓨터가 이해할 수 있는 언어인 프로그래밍 언어로 명령을 내려주는 일을 하는 사람들을 의미한다. 따라서 프로그래밍 언어를 이해하는 것은 컴퓨터적 사고방식을 이해하는 첫 걸음이라고 할 수 있다.

**2. 관점의 차이**

동일한 언어로 된 동일한 단어이지만 단순히 문 ⬝ 이과 구분만으로도 해석에 차이가 있다. 동일한 대상에 대한 해석은 어떠한 관점에서 바라보는가에 따라 다양하다. 코딩의 경우, 컴퓨팅적 사고관점에서 바라봐야 한다.

**3. Hello World?**

작업 환경 세팅이 완료되었다는 것, 작업 준비가 완료된 것을 개발자들이 은유적으로 표현하는 문구이다. 단순히 알파벳 10 글자이지만 이 문자열을 출력하기 위해서는 프로그래밍 언어, IDE, 환경변수 설정 등 최소한의 프로그래밍 개발 환경이 구축되어 있어야 한다. 따라서 Hello World는 단순히 알파벳 10 글자가 아닌, 컴퓨터가 Hello World라는 문장을 출력하기 위한 준비가 완료되었음을 의미한다. Hello World라는 문장의 기원은 일반적으로 널리 알려진 Kernighan 과 Ritchi의 저서인 “The C Programming Language”에서 Brian Kernighan의 ‘Tutorial to the B programming language’의 내용 중 다음 내용을 언급함으로써 최초로 시작되었다.

Main() {

Printf(“hello, world\n”);

}

일단 프로그래밍 언어 하나를 배워 두면, 다른 언어를 추가적으로 배우는 것은 어렵지 않다. 또한 컴퓨터 과학 분야는 매우 빠르게 발전하고 변화하기 때문에 프로그래밍 언어 하나만 붙잡고 있는 것은 위험하다. 어떤 프로그래밍 언어를 사용하느냐 보다 한 가지 언어라도 제대로 하는 것이 중요하고, 프로그래밍 자체를 잘하는 것이 더욱 의미가 있다. 지속적으로 변화를 받아들이고 적응할 줄 아는 센스가 필요하다.

**4. 코딩이란?**

우리는 이미 일상 속에서 자연스럽게 소프트웨어를 접한다. 스마트폰의 네비게이션, 배달 앱으로 음식주문하기, SNS, 게임 등이 그러하다. 앞으로 일상 속의 소프트웨어를 보고 ‘이 소프트웨어는 어떻게 만들어졌겠구나’, 일상 속 생활의 상황에 소프트웨어 적용을 고민해보는 습관을 들여야 한다. 이미 다양한 산업에서 컴퓨터과학(CS)와 해당 분야의 결합이 이미 진행됐거나 진행될 예정이다. 이러한 융합은 더욱 생산적인 방향으로 시너지 효과를 발휘하고 있으며 그러할 예정이다. 따라서 분야에 가릴 것 없이 기본적인 프로그래밍 능력은 필수적이 될 것이다. 프로그래밍 능력은 단순히 코딩하는 방법만이 전부가 아니라, 문제해결을 위한 과학적, 논리적 사고 능력을 의미한다. 이는 앞에서 언급한 Computational thinking을 의미한다. 단순히 코딩을 잘하거나 문법에 능숙한 것보다 다양한 분야의 업무를 소프트웨어를 통해 ‘잘’ 해결하는 것이 훨씬 중요하다. 프로그래밍 언어는 도구일 뿐이다. 따라서 이를 위해 컴퓨팅적 사고 관점에서 문제에 접근하는 것이 필요하다.

알고리즘은 이미 실생활에서 널리 사용되고 있다 (Ex. 압축 알고리즘). 알고리즘은 문제에서 일정한 패턴을 발견하고, 그 패턴(규칙)을 통해 문제를 해결하는 절차를 의미한다. 이 알고리즘을 코드로 표현하는 행동이 코딩이다.

※ 알고리즘과 메커니즘의 차이

알고리즘(Algorithm): 목적은 어떤 문제를 해결하는 것

* 문제해결을 위한 단계별 동작 순서, 일련의 수학적 절차 등

메커니즘(Mechanism): 목적은 어떤 대상을 이해하는 것

* 구조, 형태, 동작 원리, 현상의 발생 원리
* 메커니즘 ⊃ 알고리즘