프로그래밍 언어들은 자기만의 목적이 있으며 컴퓨터가 해야 하는 다양한 작업에 맞춰, 그에 맞는 언어를 사용하여 문제를 해결 할 수 있다. 그래서 나는 다양한 언어가 한가지 언어보다 더 적합하다고 본다. 컴퓨터는 사실 ‘ 프로그래밍 언어’ 라는 것이 무엇인지 모른다. 단지 “하드웨어 어느 주소에 0 또는 1 값을 넣어야 하는 지와 계산하기” 외에는 알지 못한다. 프로그래밍은 컴퓨터와 대화라고 한다. 즉 컴퓨터에게 어떤 기능을 실행하라고 명령하며 대화하는 과정이다. 대화 라고 하지만 사실 명령이 99% 이다. 컴퓨터는 단지 명령을 수행을 했는지, 못했는지에 대해서만 응답한다. 이러한 과정을 돕는게 ‘프로그래밍 언어’ 인 것 이다. 초창기 컴퓨터는 쇠와 전기로 이루어진 계산기였다. 그러기에 특별히 대화라는 것이 필요하지 않았다. 입력된 수치에 원하는 계산값만 나오면 되는 것이 목적이었다. 그래서 초창기 프로그래밍은 “언어”적인 성향보다는 “계산” 을 위한 매크로 정도의 수준이었다. 그러다가 값을 비교하고 실행이 필요해지면서 언어적인 기능이 필요하게 되었다. 결국 명령을 만들어 메모리에 올리고 실행하는 언어가 필요하게 되었고 그게 “어셈블리” 라고 불렀다. 이것은 컴퓨터의 성능 자체가 낮았던 8비트 시절까지 필수로 여겨졌다. 하지만 점점 컴퓨터의 하드웨어도 발전하고 8비트시대에서 16비트로 넘어오면서 인간에게는 좀더 편한 언어가 필요 하게 되었고 여러가지 언어들이 개발되고 사용되었지만 인류역사에 공헌한 언어가 된 것은 당연히 저급언어C 라는 언어 였다. C 언어는 메모리 연산하는 것 외에 다양한 분기문, 함수와 각종 라이브러리를 가져다 쓸 수 있는 구조였고 무척 유연했다. 물론 아직도 C언어를 가르치고 배우고 사용하고 있지만, C언어는 너무 하드웨어 중심적인 언어다 보니 저급 이라고 표현했다. 그리고 하드웨어 중심 보다는 인간적인 사고를 목적으로 발전시킨 언어를 고급 이라고 표현했다. C보다는 할 일이 많지 않고 이해도 쉬운 basic 을 사용했고, basic 언어의 단점의 느리고 하드웨어제어 문제가 생겼을 때, 제약이 많다는 것 때문에 pascal 이라는 고급언어 도 많이 사용하였다. 그러나 컴퓨터 환경이 계속해서 빠르고 거대해짐에 따라 기존의 방법으로는 소프트웨어 생산에 한계를 느끼게 되고 나온 것이 OOP(Object Oriented Programming)이다. OOP는 “인간세계” 사고방식으로 프로그래밍하는 방법이다. 이 OOP에서 가장 대중화에 성공한 언어는 java 라는 언어였다. 특정한 목적을 위해 만든 언어들도 많다. 오늘날 우리에게 일상이 되어버린 SNS(Social Networking Service)는 웹과 인터넷을 기반으로 한 세상이다. 이러한 웹과 인터넷을 전문으로 하는 언어들도 있는데 javascript, 파이썬 등등 여러가지가 쓰인다. 또한 요즘은 여러가지 플랫폼이 존재하고 그 플랫폼에 맞춰 앱이 제작 된다. 그런 플랫폼에는 데스크톱을 위한 윈도우, 맥os, 리눅스등이 있고, 모바일 기기를 위한 ios(apple 회사의 기기) 와 안드로이드가 있다. 다른 플랫폼도 있지만 이 정도가 지금 가장 대중적이다. 윈도우에서 대부분의 프로그램은 C++라는 고급언어를 사용하고 IOS 를 개발하기 위해서는 자체 회사에서 swift 라 불리는 새로운 언어를 도입 했고, 안드로이드는 주로 java를 이용한다. 이렇게 벌써 어떠한 플랫폼을 이용함에 따라 그에 맞는 다른언어 들을 사용한다. 시대가 발전 하면서 하드웨어도 그만큼 발전하고 그 하드웨어 맞춰 새로운 언어가 계속 나오는건 당연한 것 같다. 그리고 하드웨어가 한가지로 만 나오는게 아니기 때문에 여러가지 언어가 계속 나오고 는 것이다. 그래서 한가지 언어가 모든 언어를 대체할 수 있는건 불가능한 일이고 오히려 앞으로 미래발전을 막는 것일수도 있다.