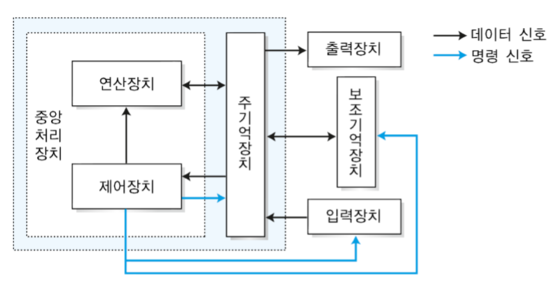
빅데이터 연구소 1차 과제 – 송기태

1. **컴퓨터의 구조와 그 기능을 설명하고 논리회로가 컴퓨터에서 어떤 역할을 하는지 설명하세요**

컴퓨터의 구조는 크게 세가지로 나뉘는데 그게 바로 CPU(Central Processing Unit)와 RAM(Random Access Memory), HDD/SSD이다.

이때 RAM과 HDD/SSD는 정보 또는 자료를 저장할 수 있는 곳, 즉 기억장치(Memory)라고 불리고, CPU는 연산장치라고 불린다.



[그림 1]

CPU는 사람의 머리에 해당하는 부분이라고 할 수 있는데, 뇌가 사람 몸에 “팔을 올려라” 라고 명령을 내리듯이 CPU는 컴퓨터에 중앙에서 모든 데이터를 처리하도록 부품에 명령을 내리는 장치이다. 입력 받은 명령어를 해석하고, 연산하는 일을 한다.

RAM은 프로그램이 실행되는 동안 필요한 정보를 저장하는 컴퓨터 메모리이다.

하는 일은 저장된 데이터를 순차적이 아닌 임의의 순서로 액세스할 수 있는 데이터 저장소이다.

HDD/SSD는 주 기억장치인 RAM을 확장한 것으로, 속도는 느리지만 용량이 큰 것이 특징이다.

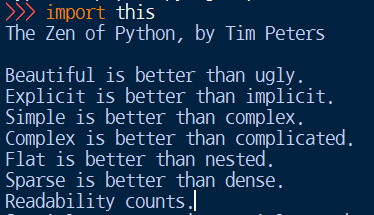
‘처리속도가 빠름 – 가격 비쌈 – 구조 복잡 – 저장용량 많음’ 순으로 정리를 해보자면,

**레지스터 – 캐시 기억장치 – 주 기억장치 – 보조 기억장치**로 나타낼 수 있다.

1. **파이썬의 언어의 특징을 설명하고 파이썬의 철학에 대해 알아보세요**

* 소스코드 전체를 한번 훑고 기계어로 변환하는 컴파일 언어인 C, C++, JAVA와 달리, 파이썬은 한 줄씩 실행하는 인터프린터 언어이다.
* 컴파일 된 기계어를 읽는 것이 아니기 때문에 속도가 컴파일 언어보다 느리다.
* 중괄호({})나 세미콜론(;) 같은 문법적인 요소들이 필요가 없다.
* 들여쓰기를 제대로 하지 않으면 코드가 실행되지 않는다.
* 다른 언어와 섞어서 제작이 가능하다. (파이썬으로 일부 작업, C언어로 나머지 작업)

아래는 파이썬의 철학이다.



"아름다운 게 추한 것보다 낫다." (Beautiful is better than ugly)

"명시적인 것이 암시적인 것 보다 낫다." (Explicit is better than implicit)

"단순함이 복잡함보다 낫다." (Simple is better than complex)

"복잡함이 난해한 것보다 낫다." (Complex is better than complicated)

"가독성은 중요하다." (Readability counts)

1. **부동소수점 연산에 대해서 알아보세요**

컴퓨터에서 실수를 표현하는 2개의 방식 중 하나이다.

하나는 ‘고정 소수점 방식’, 또 다른 하나는 ‘부동 소수점 방식’

부동 소수점 방식은 소수점 위치가 바뀌기 때문에 컴퓨터가 실수를 표현할 때 주로 사용한다.

고정 소수점 방식에 비해 덧셈이나 곱셈 계산이 복잡해지고 그만큼 연산시간도 늘어나지만, 표현할 수 있는 범위가 매우 넓은 게 장점이다. (굉장히 넓은 범위의 숫자를 표현할 수 있으면서 상대적으로 높은 정밀성을 보장한다.)

1. **MIPS assembly어의 특징과 명령어의 필요성, 명령어 집합의 fetch방식에 대해서 알아보세요**

기계와 바로 통신이 가능하다는 점이 어셈블리어에 가장 큰 특징이다.

이런 특징 때문에 어셈블리어는 다른 고급 언어와 차별되는 특징이 몇가지가 있다.

1. 명령 실행 속도가 가장 빠르다.

기계에 바로 명령을 내릴 수 있는 언어이기에, 어떤 언어보다 빠르게 기계를 제어할 수 있다.

1. 어셈블리어는 하드웨어 특성을 탄다.

어셈블리어는 하드웨어를 사실상 직접 제어하는 형태이기 때문에 하드웨어의 구조에 따라서 프로그램을 달리 해야 한다.

CPU마다 사용하는 번지의 수와 위치가 달라지기 때문에 어셈블리어로 짠 프로그램은 특정 하드웨어만 제대로 동작하는 문제가 있다

?? 명령어의 필요성, ?? fetch방식 잘 모르겠음

1. **전자계산기를 구현하세요 (언어에 상관없음)**

전자계산기 문제 : day1\_problem5.py 파일 참고

추가 구현 문제 : day1\_problem5-1.py 파일 참고

1. **Given an array of integers, find the pair of adjacent elements that has the largest product and return that product.**

실행 코드 : day1\_problem6.py 파일 참고

수식 파일 : number.txt 파일 참고

디버깅 영상 : day1\_problem6.mp4 파일 참고

1. **Given an array of strings, return another array containing all of its longest strings.**

실행 코드 : day1\_problem7.py 파일 참고 – 결국 풀지 못했음

1. **Some people are standing in a row in a park. There are trees between them which cannot be moved. Your task is to rearrange the people by their heights in a non-descending order without moving the trees. People can be very tall**

실행 코드 : day1\_problem8.py 파일 참고 – 결국 풀지 못했음

1. **Ticket numbers usually consist of an even number of digits. A ticket number is considered lucky if the sum of the first half of the digits is equal to the sum of the second half.**

실행 코드 : day1\_problem9.py 파일 참고 – 결국 풀지 못했음

1. **점프 투 파이썬의 예제 문제를 제어문 앞까지 풀어서 첨부파일 형태로 제출하세요**

실행 코드 : day1\_problem10.py 파일 참고

자료 출처 : [그림1] – dlrldy1002의 네이버 블로그

링크 : <https://blog.naver.com/PostView.nhn?blogId=dlrldyd1002&logNo=220833376113>