

데이터처리프로그래밍

변수(variable) & 자료형(data type)



강원대학교 교육혁신원 송혜정

<hjsong@kangwon.ac.kr>



변수(variable)와 자료형(data type)

✓ 학습목표

- 컴퓨터 동작 원리와 데이터 표현을 이해한다.
- 변수(variable)와 자료형(data type)을 이해한다

✓ 학습내용

- 컴퓨터 동작원리
- 변수(Variable)의 이해
- 간단한 계산
- 데이터 표현
- 자료형(Data Type)
- 형변환(Type Conversion)

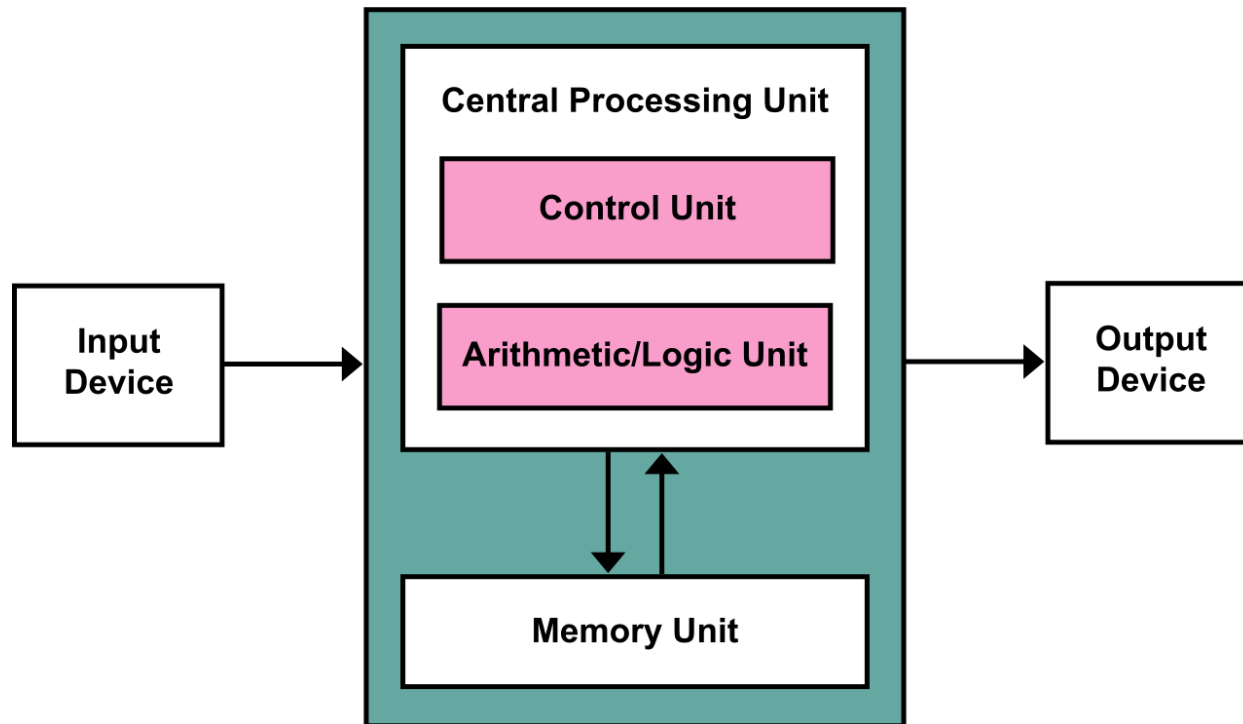


강의에 앞서서..

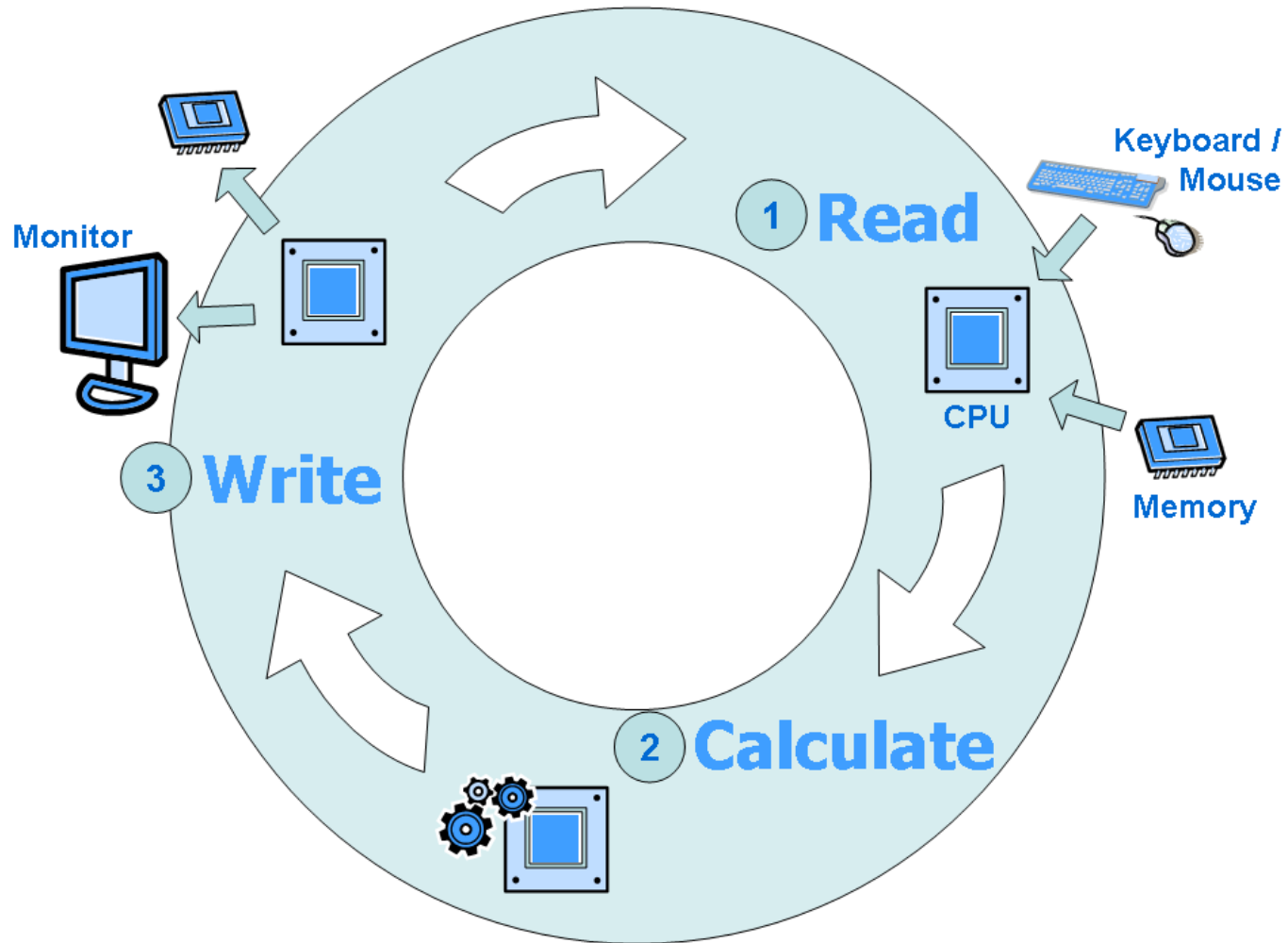
- 본 강의자료는 아래의 자료들을 참고하여 만들어 졌음을 알립니다
 1. 데이터과학을 위한 파이썬 프로그래밍, 최성철, 한빛아카데미,2019
 2. Python (<https://docs.python.org>)

컴퓨터 구조

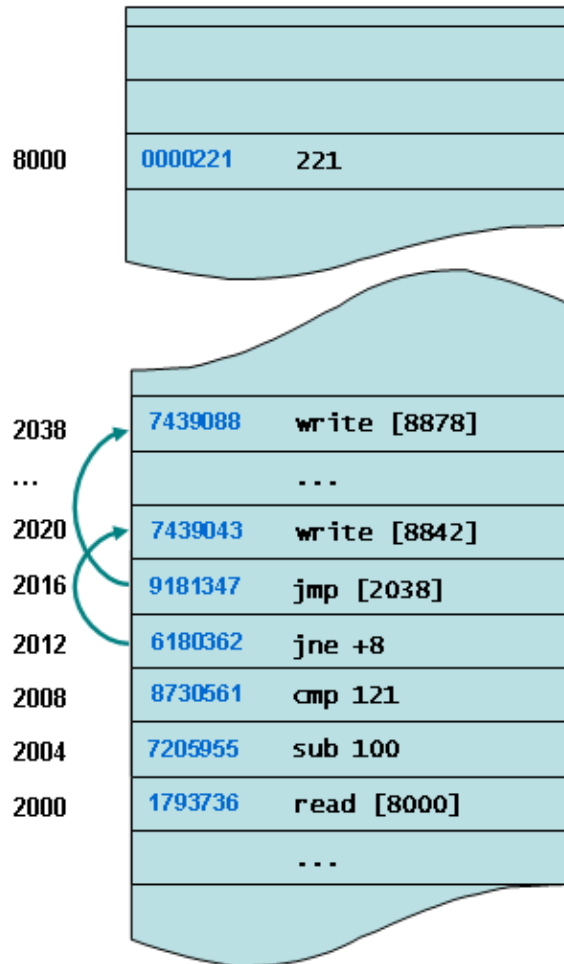
- 폰 노이만 아키텍처 (**Von Neumann architecture**)
 - 명령어와 데이터를 내장 메모리에 저장하여 순차적으로 CPU에서 처리하는 방식



컴퓨터 동작원리



메모리 안의 프로그램(실행명령)과 데이터



read [8000]	주소 8000의 값을 CPU로 읽습니다.
sub 100	CPU에 로드된 값에서 100을 뺍니다. 앞서 CPU로 읽어 들인 값이 빼기 연산의 결과 값으로 바뀝니다.
cmp 121	CPU의 현재 값을 121과 비교합니다.
jne +8	CPU 값이 121과 같지 않으면 현재 명령에서 8바이트 앞(주소 2020)으로 이동하여 이 메모리 위치에 들어 있는 명령을 실행합니다. 여기서 jne는 Jump Not Equal을 나타냅니다.
	CPU 값이 121이면 CPU가 점프하지 않고 순서상 다음 메모리 위치(이 예의 경우 2016)에서 명령을 계속 읽습니다.
jmp [2038]	주소 2038로 이동하여 이 메모리 위치에 들어 있는 명령을 실행합니다.
write [8842]	CPU의 값을 메모리 위치 8842에 기록합니다.

간단한 파이썬 프로그램

숫자를 이용한 간단한 계산

```
In [1]: 10 + 20
```

```
Out[1]: 30
```

```
In [2]: 10 / 3 # 나누기 연산은 실수(floating point number) 반환
```

```
Out[2]: 3.3333333333333335
```

```
In [3]: 10 // 3 # 나눈 결과를 정수로 내림(floor division)
```

```
Out[3]: 3
```

```
In [4]: 10 % 3 # 나머지 연산
```

```
Out[4]: 1
```

```
In [5]: 100 - (30 * 2) / 3 #괄호로 우선연산 처리
```

```
Out[5]: 80.0
```

```
In [6]: 2 ** 4 # 2 to the power of 4
```

```
Out[6]: 16
```

변수의 이해

- 변수(Variable)
 - 프로그램에서 필요한 데이터 값을 저장하기 위한 기억장소

- 변수에 값 대입

변수명 = 값

- 변수명 : 변수에 붙여진 이름
- '=' 연산자로 값을 대입

```
In [7]: #변수 x, y, name에 값을 대입
x = 100
y = 20.5
name = "홍길동"
```

```
In [8]: # 변수 x값 확인
x
```

```
Out[8]: 100
```

```
In [9]: # 변수 name값 확인
name
```

```
Out[9]: '홍길동'
```

```
In [10]: #여러변수에 값 할당
f1, f2, f3 = "Apple", "Orange", "Banana"
```

```
In [11]: f3
```

```
Out[11]: 'Banana'
```

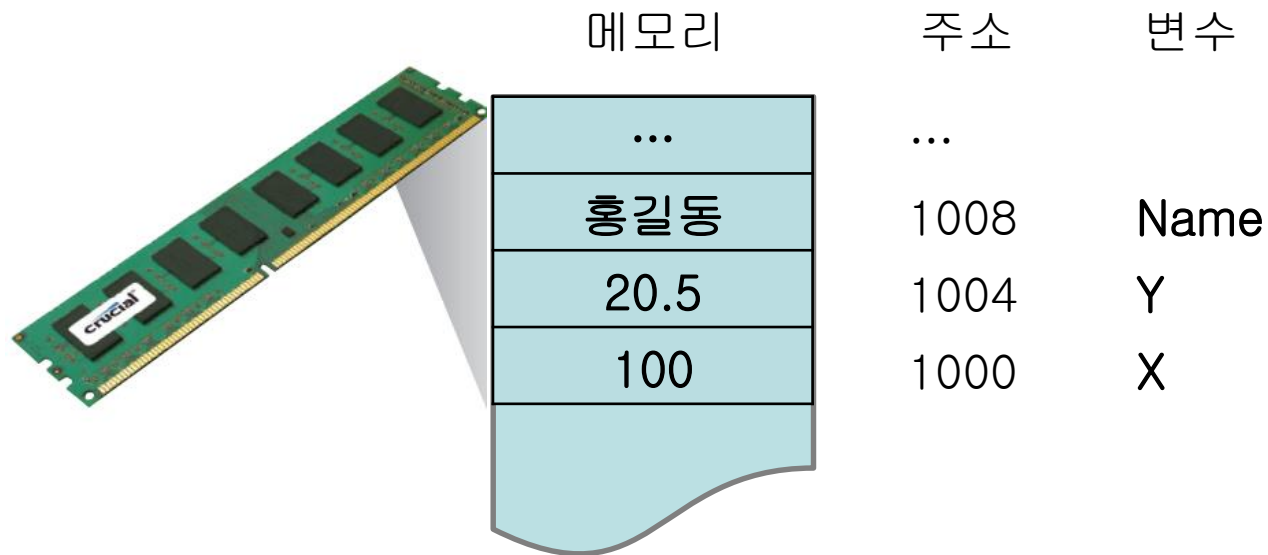
```
In [12]: # 다른 변수에 동일한 값 할당
n1 = n2 = n3 = "Apple"
```

```
In [13]: n1
```

```
Out[13]: 'Apple'
```


변수의 이해

- 변수와 메모리



변수의 이해

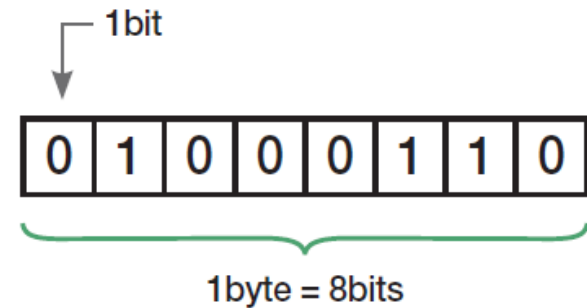
- 파이썬 변수명 명명 규칙
 - 변수 이름은 문자 또는 밑줄 문자로 시작
ab, _ab (O)
 - 변수 이름은 숫자로 시작할 수 없다
2ab , 1x (X)
 - 변수 이름은 영숫자((A-z, 0-9), 밑줄 (_)) 만 포함
A_9, a2 (O) a!, X%, x_+b (X)
 - 변수 이름은 대소 문자를 구분
AB변수와 ab변수는 다름
 - 변수명은 의미 있는 단어로 표기하는 것이 좋다.
name, score, age
 - 파이썬에서 특별한 의미로 정의된 단어(예약어), 명령어는 사용할 수 없다.
if, else, for (X)

데이터 표현 단위

- 모든 정보를 2진수로 표현하는 컴퓨터
 - 0 또는 1의 2진수 개념은 모든 전기적인 장치의 on/off를 표현하기에 적합
- 컴퓨터의 데이터 표현단위

- Bit (binary digit)
 - 0 또는 1을 표현하는 단위
 - 2진수의 1자리 표현
 - n 비트로 2^n 개의 정보를 표현
- 1bit : 0, 1
- 2bit : 00, 01, 10, 11
- 4bit :

1	0000	9	1000
2	0001	10	1001
3	0010	11	1010
4	0011	12	1011
5	0100	13	1100
6	0101	14	1101
7	0110	15	1110
8	0111	16	1111



데이터 표현 단위

– Byte

- 정보를 표현하는 최소단위 (1 Byte = 8bit)
- 1byte = 1character
 - 영어는 1byte로 1 문자 표현, 한글은 2byte가 필요
- 기억장치(메모리)의 저장단위
(1KB = 2^{10} B, 1MB = 2^{20} B, 1GB = 2^{30} B, 1TB = 2^{40} B)

단위	크기	비유
bit	1bit	두 가지 상태만 표현 가능
byte	8bit	제한된 문자 표현
KB(Killo Byte)	1024byte	간단한 편지
MB(Mega Byte)	1024KB	한 장의 사진
GB(Giga Byte)	1025MB	한 편의 영화
TB(Tera Byte)	1024GB	?



수의 체계

- 컴퓨터 내부에서 수를 표현하는 방법
- 10진법, 2진법, 8진법, 16진법 사용

10진수	2진수	8진수	16진수
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10
17	10001	21	11
18	10010	22	12



수의 체계

진법변환

1) 십진수를 이진수로 변환

- 연속적으로 2로 나눴셈을 수행하면서 얻어지는 나머지로 생성
- $(41)_{10}$ 의 이진수로 변환
 - 41을 2로 연속해서 나누어 생성된 나머지를 역순으로 정렬
 - $(41)_{10} = (101001)_2$

2		41		
2		20	1
2		10	0
2		5	0
2		2	1
		1	0

2) 이진수를 십진수로 변환

- 이진수를 2^N 로 표현하여 계산

110111 (2진수)

$$\begin{aligned}
 &\downarrow \\
 &(1 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) \\
 &\downarrow \\
 &(1 \times 32) + (1 \times 16) + (0 \times 8) + (1 \times 4) + (1 \times 2) + (1 \times 1) \\
 &\downarrow \\
 &32 + 16 + 0 + 4 + 2 + 1 \\
 &\downarrow \\
 &55 \text{ (10진수)}
 \end{aligned}$$

