

Do it!

C 언어 입문

003

자료형

3.1

자료형

데이터 저장 단위

비트 (Bit)

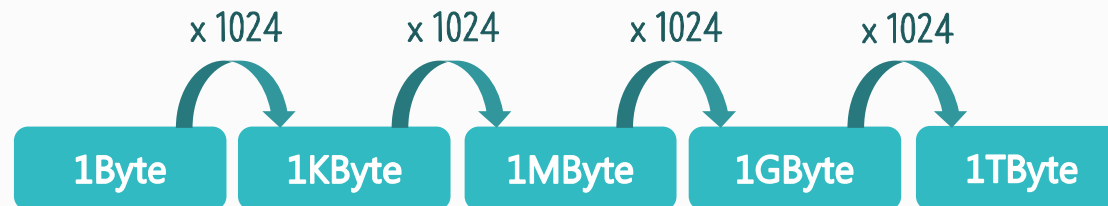
두 가지 중 한 가지 값을 저장할 수 있는 단위

바이트 (Byte)

8개의 비트로 구성된 저장 단위



$$2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 = 2^8 = 256$$



4 Bytes = 32 bits

$$2^{32} = 2^2 \times 2^{10} \times 2^{10} \times 2^{10} = 4 \text{ Gbytes}$$

3.1

자료형

부호 비트

부호 비트 양수와 음수를 구별하기 위한 비트

부호 비트 (0 : 양수, 1 : 음수)



$$2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 = 2^7 = 128$$

양수의 표현 범위 0 ~ 127

음수의 표현 범위 -128 ~ -1

부호의 개념을 가지는 1Byte 의 표현 범위 -128 ~ 0 ~ 127

3.1

자료형

부호 비트 예제

부호 비트

1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

부호가 있는 자료형 -128

부호가 없는 자료형 128

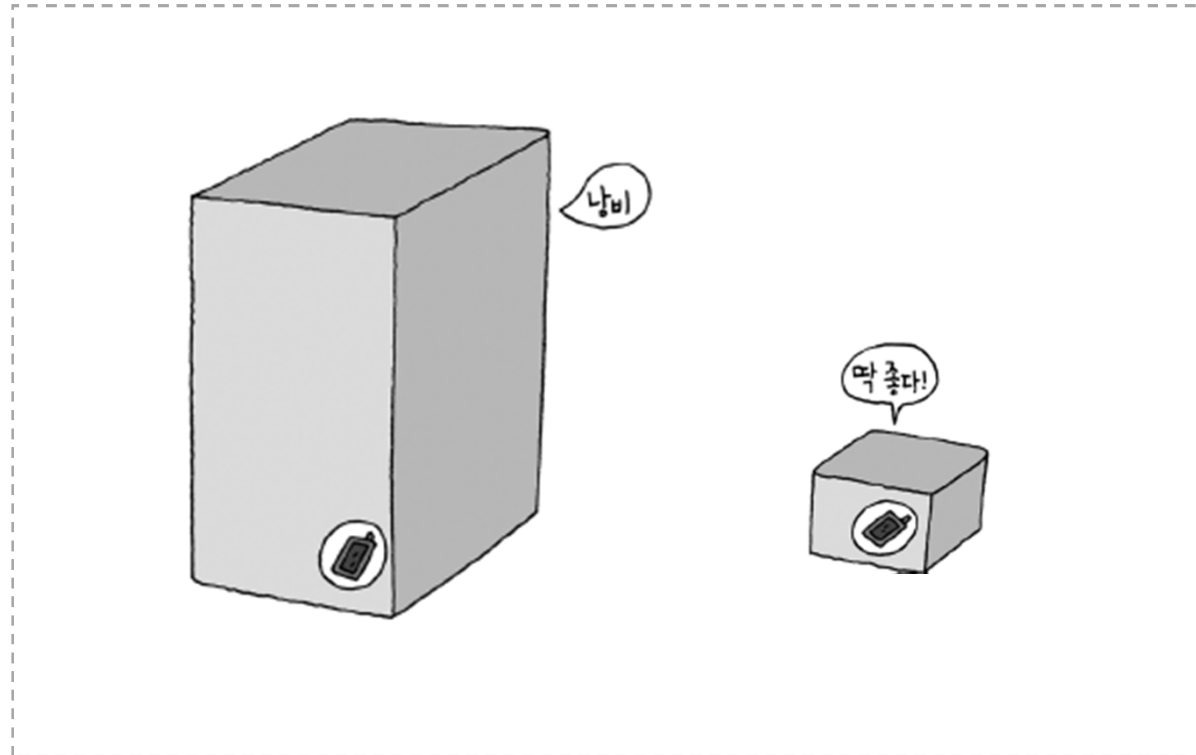
부호 비트

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

부호가 있는 자료형 -1

부호가 없는 자료형 255

메모리를 절약하는 것이 중요한가?



3.2

자료형

아스키(ASCII)코드란?

10	16	문자	10	16	문자	10	16	문자	10	16	문자
0	0	Null	33	21	!	66	42	B	99	63	c
1	1	Start Of Heading	34	22	*	67	43	C	100	64	d
2	2	Start Of Text							101	65	e
3	3	End Of Text							102	66	f
4	4	End Of Transmit							103	67	g
5	5	Enquiry							104	68	h
6	6	Acknowledge							105	69	i
7	7	Audible Bell (Beep)	40	28	(73	49	I	106	6A	j
			29))	74	4A	J	107	6B	k
			2A	*	*	75	4B	K	108	6C	l
			2B	+	+	76	4C	L	109	6D	m
			2C	,	,	77	4D	M	110	6E	n
			2D	-	-	78	4E	N	111	6F	o
			2E	.	.	79	4F	O	112	70	p
			2F	/	/	80	50	P	113	71	q
14	E	Shift Out	47	2F	/	80	50	P	113	71	q
15	F	Shift In	48	30	0	81	51	Q	114	72	r
16	10	Data Link Escape	49	31	1	82	52	R	115	73	s
17	11	Device Control 1	50	32	2	83	53	S	116	74	t
18	12	Device Control 2	51	33	3	84	54	T	117	75	u
19	13	Device Control 3	52	34	4	85	55	U	118	76	v
20	14	Device Control 4	53	35	5	86	56	V	119	77	w
21	15	Neg. Ack	54	36	6	87	57	W	120	78	x
22	16	Synchronous Idle	55	37	7	88	58	X	121	79	y
23	17	End Tras. Block	56	38	8						
			39	9	9						
			3A	:	:						
			3B	;	;						
			3C	<	<						
			3D	=	=						
			3E	>	>	95	5F	-			
30	1E	Record Separator	63	3F	?	96	60	`			
31	1F	Unit Separator	64	40	@	97	61	a			
32	20	Space Bar	65	41	A	98	62	b			

초항기 128개이전 아스키(ASCII) 값

ASCII

(American Standard Code
Information Interchange)

컴퓨터에서 문자를
숫자로 표현하기 위한 약속

자료형은 왜 필요할까?

자료형 (Data Type) 실제로 메모리를 사용하는 단위

Built-in Data Type

사용 빈도가 높은 자료형을 예약어로 제공

User-defined Data Type

사용자가 만들어서 사용할 수 있는 문법 제공

정수형 자료형

C 언어 프로그램에서 정수 값을 메모리에 저장하기 위해 사용하는 자료형

실수형 자료형

C 언어 프로그램에서 실수 값을 메모리에 저장하기 위해 사용하는 자료형

3.3

자료형

정수를 표현하는 자료형 char

signed char : 부호가 있는 1바이트 저장 공간

```
signed char temperature;
```

```
temperature = -2; /*영하 2°C를 temperature 변수에 저장함*/
```

unsigned char : 부호가 없는 1바이트 저장 공간

```
unsigned char age;
```

```
age = 52; /*나이 52살을 age 변수에 저장함*/
```

C 언어는 1Byte라 하지 않고 왜 char라 하는가 ?

정수를 표현하는 자료형 short int

signed short int : 부호가 있는 2바이트 저장 공간

```
signed short dday;
```

```
dday = -20; /* D-day가 -20일 남았다고 dday 변수에 저장함*/
```

unsigned short int : 부호가 없는 2바이트 저장 공간

```
unsigned short int seconds;
```

```
seconds = 35000; /*3만 5000초를 second 변수에 저장함*/
```

정수를 표현하는 자료형 long int

signed long int : 부호가 있는 4바이트 저장 공간

```
signed long int money;  
money = 7000000; /* 700만원을 money 변수에 저장함*/
```

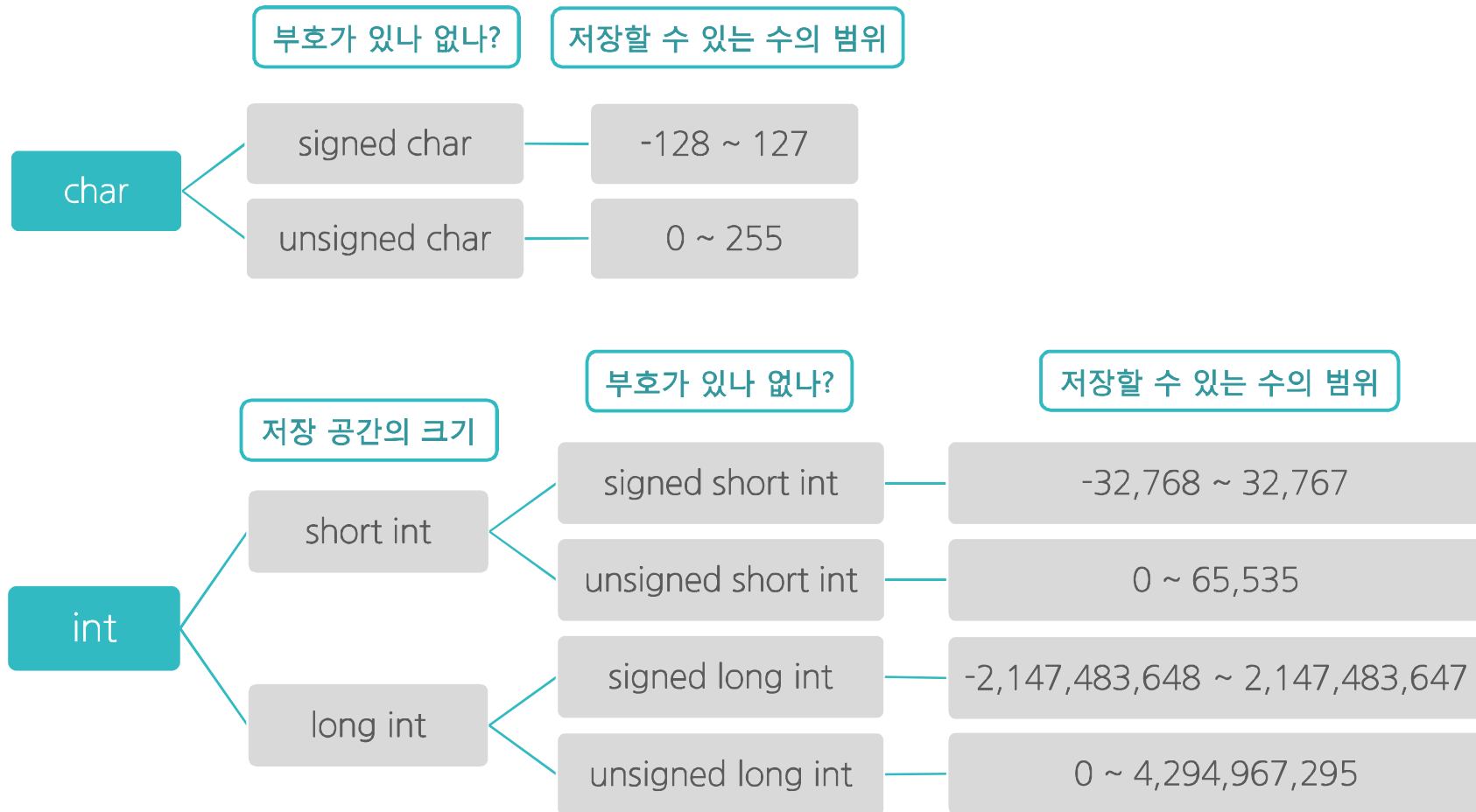
unsigned long int : 부호가 없는 4바이트 저장 공간

```
unsigned long int time_seconds;  
/* 1970년 1월 1일 부터 현재까지 시간을 초단위로 환산한 다음  
time_seconds 변수에 저장함 */  
time_seconds = 145310624;
```

3.3

자료형

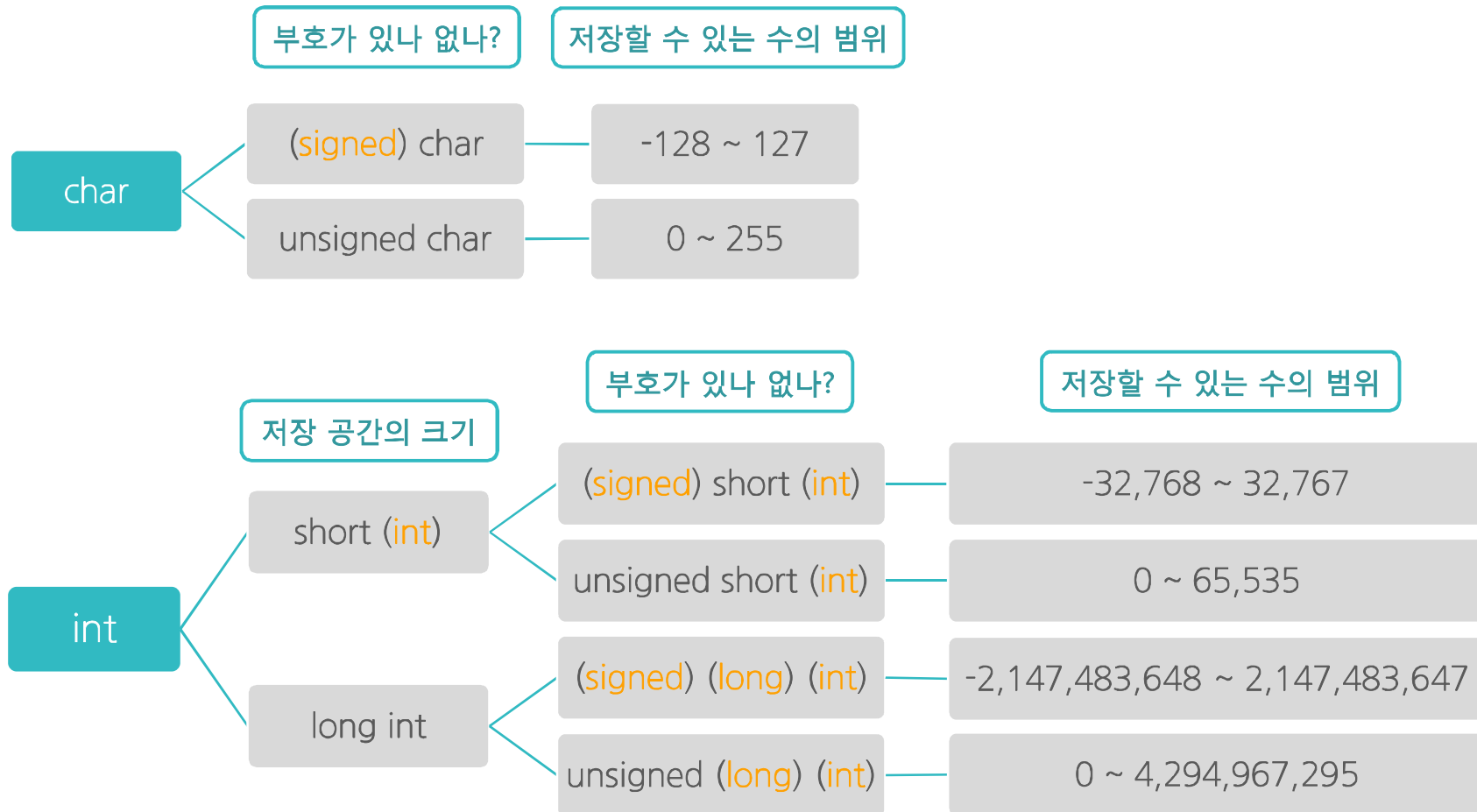
정수를 표현하는 자료형



3.3

자료형

정수를 표현하는 자료형



실수를 표현하는 자료형

부동소수점

실수를 표현할 때 소수점의 위치를 고정하지 않고 소수점의 위치를 나타내는 수를 따로 작성하는 방식이 부동소수점(floating-point) 표현 방식이다.

C 언어는 실수를 저장하기 위해 4바이트와 8바이트 자료형을 제공한다.

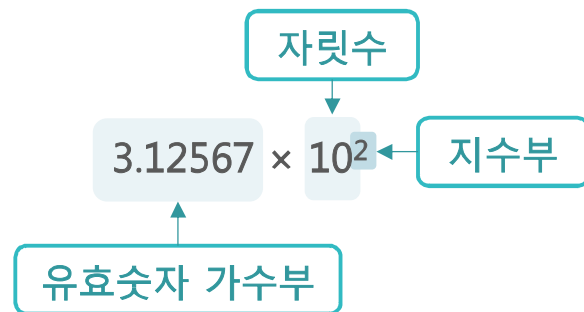
고정소수점

일정한 위치에 소수점의 위치를 고정하여 숫자를 나열하는 방법

53.8
123.45



000538000
001234500



3.3

자료형

실수를 표현하는 자료형

크기

저장할 수 있는 값의 범위

float

4 바이트

$1.2\text{E-}38 \sim 3.4\text{E}38$



IEEE 754 규약에 정의된 float형의 부동소수점 표현

double

8 바이트

$2.2\text{E-}308 \sim 1.8\text{E}308$



Do it!

C 언어 입문