
Lecture 2. 자료형과 입출력

BAE SEKANG(배세강)

Contents

1. Data Types

- 변수
- 기본 types
- unsigned
- 형 변환
- sizeof()
- 서식 지정자

2. Output

- scanf
- 전처리기

3. Exercise

Data Types

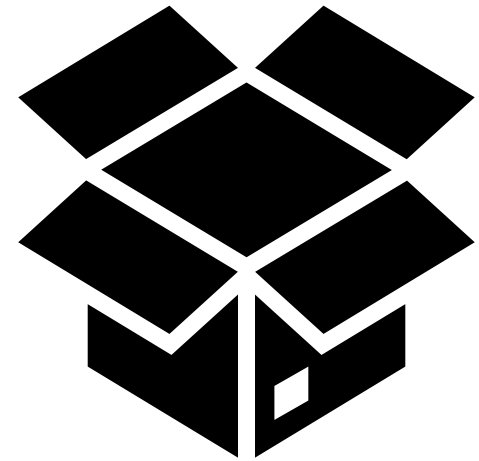
♣ 변수(variable)

- 값을 저장하는 공간

```
#include <stdio.h>

int main() {
    int apple_count = 5;
    apple_count = 10;
    printf("사과 개수: %d개\n", apple_count);

    return 0;
}
```



Data Types

♣ 변수(variable)

- 변수를 초기화하지 않으면 쓰레기 값(Garbage Value)이 들어있다.
- `const`를 앞에 붙이면 상수 변수로 선언한다.
- 변수명 규칙이 존재한다.

[변수명 규칙]

1. 영문자, 숫자, 언더바(_)만 사용 가능
2. 숫자로 시작할 수 없음
3. 대소문자를 구분함
4. 예약어 사용 불가(int, return, if 등)
5. 공백(띄어쓰기) 사용 불가

좋은 변수명: `user_age`, `total_score`, `student_count` 등

Data Types

♣ 정수형 Type

- 모든 Type에는 저장할 수 있는 공간의 크기가 유한하다.

Types	Size	범위
char	1 Byte	-128 ~ 127
short	2 Bytes	-32768 ~ 32767
int	4 Bytes	약 -21억 ~ 21억
long	4 or 8 Bytes	int와 비슷 or 더 큼
long long	8 Bytes	약 -922경 ~ 922경

Data Types

♣ 실수형 Types

- 실수는 항상 오차가 존재할 수 있으니 유의해야 한다.

Types	Size	범위
float	4 Bytes	$-3.4 \times 10^{38} \sim 3.4 \times 10^{38}$
double	8 Bytes	$-1.7 \times 10^{308} \sim 1.7 \times 10^{308}$

- 지수 표기법: [기수]e[지수]

예) 300: 3e2, 1234000: 1.234e6, 0.12: 1.2e-1

Data Types

♣ ASCII Code

- 일정 범위의 숫자에 문자를 할당하여 문자를 처리한다.

Dec	Hex	Name	Char	Ctrl-char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	0	Null	NUL	CTRL-@	32	20	Space	64	40	@	96	60	`
1	1	Start of heading	SOH	CTRL-A	33	21	!	65	41	A	97	61	a
2	2	Start of text	STX	CTRL-B	34	22	"	66	42	B	98	62	b
3	3	End of text	ETX	CTRL-C	35	23	#	67	43	C	99	63	c
4	4	End of xmit	EOT	CTRL-D	36	24	\$	68	44	D	100	64	d
5	5	Enquiry	ENQ	CTRL-E	37	25	%	69	45	E	101	65	e
6	6	Acknowledge	ACK	CTRL-F	38	26	&	70	46	F	102	66	f
7	7	Bell	BEL	CTRL-G	39	27	'	71	47	G	103	67	g
8	8	Backspace	BS	CTRL-H	40	28	(	72	48	H	104	68	h
9	9	Horizontal tab	HT	CTRL-I	41	29	)	73	49	I	105	69	i
10	0A	Line feed	LF	CTRL-J	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	VT	CTRL-K	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	FF	CTRL-L	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage feed	CR	CTRL-M	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	SO	CTRL-N	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	SI	CTRL-O	47	2F	/	79	4F	O	111	6F	o
16	10	Data line escape	DLE	CTRL-P	48	30	0	80	50	P	112	70	p
17	11	Device control 1	DC1	CTRL-Q	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	DC2	CTRL-R	50	32	2	82	52	R	114	72	r
19	13	Device control 3	DC3	CTRL-S	51	33	3	83	53	S	115	73	s
20	14	Device control 4	DC4	CTRL-T	52	34	4	84	54	T	116	74	t
21	15	Neg acknowledge	NAK	CTRL-U	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	SYN	CTRL-V	54	36	6	86	56	V	118	76	v
23	17	End of xmit block	ETB	CTRL-W	55	37	7	87	57	W	119	77	w
24	18	Cancel	CAN	CTRL-X	56	38	8	88	58	X	120	78	x
25	19	End of medium	EM	CTRL-Y	57	39	9	89	59	Y	121	79	y
26	1A	Substitute	SUB	CTRL-Z	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	ESC	CTRL-[	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	FS	CTRL-\	60	3C	<	92	5C	\	124	7C	
29	1D	Group separator	GS	CTRL-]	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	RS	CTRL-^	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	US	CTRL-~	63	3F	?	95	5F	_	127	7F	DEL

Data Types

♣ ASCII Code

- A는 ASCII Code로 65에 해당한다.

```
#include <stdio.h>
```

```
int main() {  
    char c = 'A';  
    printf("%d", c);  
}
```

>> 65

Data Types

♣ Escape Sequence

예약 문자	내용
\n	줄 바꿈(New line)
\t	Tab
\\	\ 문자 자체
'\''	' 문자 자체
'\"'	" 문자 자체
\b	Backspace
...	...

Output

[Practice]

아래 텍스트를 출력한다.

[출력]

물건: "갤럭시 6 프로"

가격: 2,800,000₩

Data Types

♣ unsigned

- unsigned: 음수 범위를 포기하고, 양수 표현 범위를 2배 늘린다.

```
#include <stdio.h>

int main() {
    unsigned int num = 4000000000;
    printf("값: %u\n", num);
    return 0;
}
```

- 만약 unsigned에 -1을 넣으면 어떻게 될까? ➔ Integer Underflow

Data Types

♣ Integer Overflow/Underflow

- Overflow: 표현할 수 있는 최댓값을 넘어서 다시 최솟값으로 돌아감
- Underflow: 표현할 수 있는 최솟값보다 작아져 다시 최댓값으로 돌아감

```
#include <stdio.h>

int main() {
    unsigned int num;
    scanf("%u", &num); // -1 입력
    printf("값: %u\n", num); // 값: 4294967295
    return 0;
}
```

※ 참고

$$\begin{array}{r} 10000 (0) \\ - \\ 0001 (1) \\ \hline 1111 (15) \end{array}$$

Data Types

♣ Type Conversion

- 묵시적 형변환(Implicit Conversion): 자동으로 자료형을 바꿔줌
- 명시적 형변환(Explicit Conversion): 사용자가 직접 자료형을 바꿈

```
#include <stdio.h>

int main() {
    int a = 5;
    double b = 2.5;
    printf("%f", a + b); //a가 double로 변함

    return 0;
}
```

>> 7.500000

Data Types

♣ Explicit Conversion

- 형 변환을 할 값 앞에 ([변환할 자료형])을 입력한다.

```
#include <stdio.h>
```

```
int main() {
```

```
    float sum = (float) 5 / 2;
```

```
    printf("%f", sum);
```

```
}
```

>> 2.500000

Data Types

♣ sizeof()

- 자료형의 크기를 확인할 수 있다.

```
#include <stdio.h>
```

```
int main() {  
    printf("%d", sizeof(int));  
    return 0;  
}
```

>> 4

Data Types

♣ 서식 지정자

Types	서식 지정자	Types	서식 지정자
int	%d	long	%ld
float	%f	long long	%lld
char	%c	8진수 정수	%o
string	%s	16진수 정수	%x
double	%lf	지수 형태 표기	%e
unsigned int	%u	포인터의 메모리 주소	%p

Data Types

♣ 서식 지정자 활용

- 소수점 자리 제한: %.[숫자]f
- 전체 폭 및 정렬: %.[숫자]d
- 빈 칸을 0으로 채움: %0[숫자]d
- 부호 표시: %+d
- 문자열 n글자 표시: %.[숫자]s

```
#include <stdio.h>

int main() {
    float pi = 3.141592;
    printf("pi의 소수점 2자리까지: %.2f\n", pi);

    return 0;
}
```

Output

♣ scanf()

- 키보드로부터 데이터를 입력 받아 변수에 저장하는 표준 입력 함수
- `scanf("서식지정자", &변수명);`

```
#include <stdio.h>

int main() {
    int n;
    printf("숫자 입력: ");
    scanf("%d", &n);
    printf("입력한 숫자: %d\n", n);
    return 0;
}
```

Output

♣ scanf()

- 주소 연산자(&): 입력받은 값을 저장할 변수의 메모리 주소 표현
- Visual Studio 오류 (C4996) 해결

```
#define _CRT_SECURE_NO_WARNINGS
```

```
#include <stdio.h>
int main() {
    int n;
    scanf("%d", &n);
    return 0;
}
```

Output

♣ scanf()

- 예시 1. 여러 개의 값 입력 받기

```
#include <stdio.h>

int main() {
    int a, b;
    printf("숫자 두 개 입력: ");
    scanf("%d %d", &a, &b);
    printf("합계: %d\n", a + b);
    return 0;
}
```

Output

♣ scanf()

- 예시 2. 문자열 입력 받기

```
#include <stdio.h>

int main() {
    char name[20]; // 배열은 추후에 중요하게 다룹니다.
    printf("이름 입력: ");
    scanf("%s", name); // 여기서는 &를 사용하지 않습니다.
    printf("반갑습니다, %s님!\n", name);
}
```

Output

♣ scanf()

- 예시 3. 키 입력 받아 소수 단위로 출력하기

```
#include <stdio.h>

int main() {
    float height;
    printf("키 입력: ");
    scanf("%f", &height);
    printf("입력된 키: %.1f cm\n", height);
    return 0;
}
```

Output

[Practice]

이름과 나이를 입력 받아서 화면에 예시와 같이 출력한다.

[입력]
홍길동 25

[입력]
심청 19

[출력]
이름: 홍길동
나이: 25세

[출력]
이름: 심청
나이: 19세

Output

♣ 전처리기(preprocessor)

- 컴파일하기 전 소스 코드를 미리 처리하는 도구
- #로 시작하는 문장

```
#define MAX 100 // MAX를 100으로 치환

#include <stdio.h>

int main() {
    printf("최대값: %d\n", MAX);
    return 0;
}
```

Output

[Practice]

전처리기를 활용해 상수 PI를 3.141592라고 선언한다. 반지름을 입력 받을 때, 원의 면적을 출력하는 프로그램을 작성한다. (소수점 첫째 자리까지 출력)

[입력]

반지름: 5

[출력]

원의 면적: 78.5

Output

[Practice] - Answer

전처리기를 활용해 상수 PI를 3.141592라고 선언한다. 반지름을 입력 받을 때, 원의 면적을 출력하는 프로그램을 작성한다. (소수점 첫째 자리까지 출력)

```
#define PI 3.141592
#include <stdio.h>
int main() {
    double radius, area;
    printf("반지름 입력: ");
    scanf("%lf", &radius);
    area = PI * radius * radius;
    printf("원의 면적: %.1lf\n", area);
}
```

Exercise

[Problem 1]

정수 A를 입력 받을 때, A를 다섯 자리 형태로 출력한다. (단 $0 \leq A \leq 9999$)

[입력 1]

321

[입력 2]

2

[출력 1]

00321

[출력 2]

00002

Exercise

[Problem 1] - Answer

```
#include <stdio.h>

int main() {
    int a;
    scanf("%d", &a);
    printf("%05d", a);

    return 0;
}
```



THANK YOU!

Any Question?