

# Practice 03. Characters & Math

Biointelligence Laboratory  
School of Computer Science and Engineering  
Seoul National University

# ASCII

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	\$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(	72	48	H	104	68	h
9	09	Horizontal tab	41	29	)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	0	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	
29	1D	Group separator	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□

# ASCII Features

- 'a'부터 'z'까지 알파벳 순서대로 값이 1씩 증가한다.
  - 'a' (97) 'b' (98) 'c' (99) ... 'z' (122)
- 'A'부터 'Z'까지 알파벳 순서대로 값이 1씩 증가한다.
  - 'A' (65) 'B' (66) 'C' (67) ... 'Z' (90)
- '0'부터 '9'까지 순서대로 값이 1씩 증가한다.
  - '0' (48) '1' (49) '2' (50) ... '9' (57)

# ASCII Features

- [Example] Print the ASCII code value of a character.

```
#include <stdio.h>

int main()
{
    char c;
    scanf("%c", &c);
    printf("ASCII code of %c is %d", c, c);
}
```

# ASCII Features

- [Example] Convert a lower case character to a upper case character.

```
#include <stdio.h>

int main()
{
    char c;
    scanf("%c", &c); // assume that input is a lower case character

    printf("Upper case of %c is %c", c, c - 32);
}
```

# Tricks with Division and Modulus Op.

- 정수 나눗셈과 나머지 연산으로 특정 자리에 있는 수 추출
  - $1265287398472 / 1000 \% 10$
  - $1265287398 \% 10$
  - $8$
- 나머지 연산으로 홀짝 구분하기
  - $n \% 2 == 1$  (odd)
  - $n \% 2 == 0$  (even)

# Tricks with Division and Modulus Op.

- [Example] Print out hundreds digit of an positive integer.

```
#include <stdio.h>

int main()
{
    int v;
    scanf("%d", &v);

    printf("%d", v / 100 % 10);
}
```

# Standard Math Library

## ■ #include <math.h>

- math.h에는 수학과 관련된 함수들이 모여있다.
- sin, cos, tan, sqrt, pow, ...
- gcc로 컴파일시 -lm 옵션을 줘야 한다. libm를 링크한다.

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

int main() // calculate distance between two points (x1,y1),(x2,y2)
{
    double x1,y1,x2,y2;
    scanf("%lf%lf%lf%lf", &x1, &y1, &x2, &y2);

    printf("%f", hypot(x1 - x2, y1 - y2));
}
```



# Standard Math Library

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

int main() // compare two input value
{
    double x1,x2;
    scanf("%lf%lf", &x1, &x2);

    printf("%f is smaller than or equal to %f\n", fmin(x1, x2), fmax(x1, x2));
}
```

# Standard Math Library

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

int main() // calculate e^x
{
    double e = 2.71828182845904523, x;
    scanf("%lf",&x);
    printf("%f\n", pow(e, x));
}
```

# Type in Formatted IO

- printf의 형식 지정자는 타입변환을 하지 않는다. 인위적인 타입변환을 통해 정확한 타입의 값을 전달해야 한다.

```
#include <stdio.h>

int main()
{
    double d = 15.34;

    printf("%d", d); // strange value..
    printf("%d", 1.0); // strange value..
    printf("%d", (int)d); // 15
    printf("%d", (int)1.0); // 1
}
```

# Practice Submission

- **Submit the practice problems if they are not checked in the class time.**
- Submit the solution code of **practice problem 01, 02** by email.
- [hnikwak@bi.snu.ac.kr](mailto:hnikwak@bi.snu.ac.kr)
- Mail title: **prg\_[student number]\_practice03**
  - prg\_2014-12345\_practice03
- Submit two source files named **p01.c, p02.c**, for each problem.
- Due to : **3/25(Wed) 23:59 pm**

# Assignment Submission

- Create a directory named **assignment** in you home directory.
- Create a directory named **03** in you **assignment** directory.
- Put your C files named **p[# of problem].c** for each problem.
  - p01.c
  - p02.c
  - ...
- Due to : **3/25(Wed) 23:59 pm**

# Practice 01 - time

- Convert the milliseconds given by an input to hours(h), minutes(m), seconds(s), and milliseconds(ms) where the input value is equal to  $[\text{input}] = h * 3600000 + m * 60000 + s * 1000 + ms$ .

[Input]

19602323

[Output]

5h 26m 42s 323ms

# Practice 02 - character

- Two lower case characters are given by input separated by a whitespace.
- Output the character which is closer to 'a'.
- if, (C) ? A: B, while, do-while, for, switch statements are not available.
- Use **fmin** function in math.h. (It's strange, but just for the practice!)

[Input]

t b

[Output]

b

# Assignment 01 - time

- The input contains hour, minute, and elapsed minutes. Output the hour and minute when time is elapsed from the given time.
- Hour is in 12-hour(1 to 12) mode. There's no need to think about p.m. or a.m.

[Input]

7 50 1205

[Output]

3 55



# Assignment 02 - math

- The input contains mean( $\mu$ ), standard deviation ( $\sigma$ ),  $x$ .
- Calculate the value of

Normal Probability Density Function

$$F(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

[Input]

3.5 1 5

[Output]

0.129518