



- c언어 기초 100제 오프라인 노트북(v1.0)- 2016.09.25

- c언어 기초 100제는...

텍스트 기반 프로그래밍 언어를 처음 배우는 초중고 학생들을 위해 설계되었습니다.

- Dev-C++, Code::Blocks 와 같은 공개 IDE를 이용하고,

- <http://koistudy.net>

- <http://codeup.kr>

- <http://judgeon.net>

과 같은 학교 정보 선생님들이 운영하는 온라인 채점 사이트를 통해서,

- 다른 사람들의 도움이 없이도

- 스스로 문제를 읽고,

- 주어지는 예시 코드를 붙여넣고,

- 코드를 읽고 이해하고 수정하며,

- c언어 기초 학습 문제들을 스스로 해결해가며,

- 매우 빠른 시간 내에 c언어의 기초를 충분히 학습하고,

- 자신의 생각을 체계적, 논리적으로 c언어로 표현하는 능력을 기를 수 있습니다.

- 본 오프라인 노트북 파일(양면 인쇄용)은...

- 한국 내의 개인적학습/초중고교육을 위한 자유로운 인쇄 및 활용이 가능하며,

- 학교 공교육 이외의 영리 목적의 사교육/학원 등의 교육을 위한 인쇄는 불허합니다.

- 대학 교육 및 한국 이외에서의 인쇄/출판/활용은 저작자의 허락/허가가 필요합니다

- 중고등학교 정보 선생님 외에 **저작자와 협의하지 않은 개인/단체 임의 배포 금지**

- c언어 기초 100제의 모든 내용과 채점 데이터들은...

- 10여년 이상 학교 현장에서 c언어를 교육해 온 정보 선생님들의 순수 창작물로서,

- 수 많은 초중고학교에서의 교육활동에 무료로 활용되고 있습니다.



Computer Science & Love ™

Computer Science teachers' League ™

한국 컴퓨터과학교사 교과연구 카페/동호회 Since 2008





- 본 오프라인 노트북에 사용된 모든 그림의 저작권은 김지혜 선생님에게 있습니다.

## 출력하기01

C/C++언어에서 가장 기본적인 명령이 출력문이다.

printf()를 이용해 다음 단어를 출력하시오.

Hello

참고

C/C++ IDE 프로그램을 설치한 후,

아래와 같은 소스 코드를 작성하고 "컴파일-실행"을 수행시키면,

지정한 "문장"이 출력된다.

```
#include <stdio.h>
int main()
{
    printf("문장");
    return 0;
}
```

### ◎ 입력 형식

입력 없음

### ◎ 출력 형식

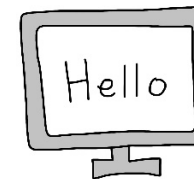
Hello

를 출력한다.

입력 예시

출력 예시

Hello



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 출력하기02

이번에는 공백(' ')을 포함한 문장을 출력한다.  
다음 문장을 출력해보자.

Hello World  
(대소문자에 주의한다.)

참고  
#include <stdio.h>  
int main()  
{  
 printf("문장1 문장2");  
 return 0;  
}

을 실행시키면 "문장1", "문장2"가 공백(' ')을 사이에 두고 출력된다.

### ◎ 입력 형식

입력 없음

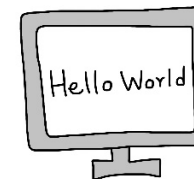
### ◎ 출력 형식

Hello World  
를 출력한다.

## 입력 예시

## 출력 예시

Hello World



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

- 01 .....
- 02 .....
- 03 .....
- 04 .....
- 05 .....
- 06 .....
- 07 .....
- 08 .....
- 09 .....
- 10 .....
- 11 .....
- 12 .....
- 13 .....
- 14 .....
- 15 .....
- 16 .....
- 17 .....
- 18 .....
- 19 .....
- 20 .....
- 21 .....
- 22 .....
- 23 .....
- 24 .....
- 25 .....
- 26 .....
- 27 .....
- 28 .....
- 29 .....

## 출력하기03

이번에는 줄을 바꿔 출력하는 출력문을 연습해보자.  
다음과 같이 줄을 바꿔 출력해야 한다.

```
Hello
World
(두 줄에 걸쳐 줄을 바꿔 출력)
```

참고  
백슬래시 문자 '\'는 글꼴 설정에 따라,  
'w'와 비슷한 원표시('w')로 보이기도 하지만 같은 문자이다.  
printf( ) 함수를 이용해 문장을 출력할 때, 줄을 바꾸기 위해서는  
'\n'(new line을 의미)을 줄을 바꿀 위치에 넣어야 한다.  
'\t'는 탭(tab),  
'\r'은 캐리지리턴(carriage return, 그 줄의 맨 앞으로 커서를 보냄)  
등이 있다.

이러한 문자를 이스케이프(escape) 문자라고 하고,  
출력하는 문장 안에서 원하는 형식에 맞추어 출력할 수 있도록  
줄을 바꾸는 등의 특별한 의미들을 나타내기 위해 사용한다.

예시  
printf("Hello\nWorld");

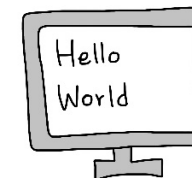
◎ 입력 형식  
입력 없음

◎ 출력 형식  
Hello  
World  
를 출력한다.

입력 예시

출력 예시

Hello  
World





Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 출력하기04

이번에는 작은 따옴표(single quotation mark)가 들어있는 특수한 형태의 출력문에 대한 연습을 해보자.

다음 문장을 출력하시오.

'Hello'

참고

printf( ) 함수로 따옴표 문자 ' 를 출력하기 위해서는 \' 로 입력해야 한다.

\' 는 작은따옴표 문자 자체 출력

\" 는 큰따옴표 문자 자체 출력 ... 등.

작은따옴표나 큰따옴표는 문자나 문장을 나타내기 위한 기호들로 쓰이는데, 그렇게 쓰이는 의미가 아니라 따옴표 그 자체를 표현해 주기 위해서이다.

예시

```
printf("\'");
```

### ◎ 입력 형식

입력 없음

### ◎ 출력 형식

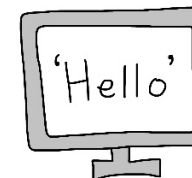
'Hello'

를 출력한다.

입력 예시

출력 예시

'Hello'



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 출력하기05

이번에는 큰따옴표(double quotation mark)가 포함된 출력문을 연습해보자.

다음 문장을 출력하시오.

"Hello World"

(단, 큰따옴표도 함께 출력한다.)

참고

printf( ) 함수로 큰 따옴표 문자 " 를 출력하기 위해서는 \" 로 출력해야 한다.

예시

```
printf("\"");
```

◎ 입력 형식

입력 없음

◎ 출력 형식

"Hello World"

를 출력한다.

입력 예시

출력 예시

"Hello World"



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 출력하기06

이번에는 특수문자 출력에 도전하자!!

다음 문장을 출력하시오.

```
"!@#$$%^&*()"
```

(단, 큰따옴표도 함께 출력한다.)

참고

% 기호가 앞에 붙는 문자를 형식 지정자(format specifier)라고 하며,  
그 위치에서 지정한 형식으로 값을 출력해 준다.

따라서, 퍼센트 문자 %를 출력하기 위해서는 %%로 출력해 주어야 한다.

예시

```
printf("%%");
```

### ◎ 입력 형식

입력 없음

### ◎ 출력 형식

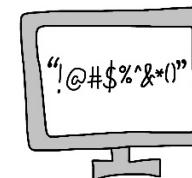
```
"!@#$$%^&*()"
```

를 출력한다.

입력 예시

출력 예시

```
"!@#$$%^&*()"
```



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 출력하기07

윈도우 운영체제의 파일 경로를 출력하는 연습을 해보자.

파일 경로에는 특수문자들이 포함된다.

다음 경로를 출력하시오.

"C:\Download\hello.cpp"  
(단, 큰따옴표도 함께 출력한다.)

참고

printf( ) 함수로 백슬래시 문자 \ 를 출력하기 위해서는 \\ 로 출력해야 한다.

예시

```
printf("\\");
```

### ◎ 입력 형식

입력 없음

### ◎ 출력 형식

"C:\Download\hello.cpp"  
를 출력한다.

## 입력 예시

## 출력 예시

"C:\Download\hello.cpp"





Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 출력하기08

이번에는 아스키코드를 출력하는 연습을 해보자.

키보드로 입력할 수 없는 다음 모양을 출력해보자.

(\*\* 참고 : 운영체제의 문자 시스템에 따라 아래와 같은 모양이 출력되지 않을 수 있다.)



참고

다음과 같은 10진수 코드로 다음과 같은 특수 문자를 표현한다고 하자.

(운영체제 또는 컴파일러에 따라 사용되는 문자의 코드표가 다르다.)

Microsoft Windows [Version 10.0.14393]  
활성 코드 페이지: 949

```
( 0) r( 1) t( 2) L( 3) J( 4)
|( 5) -( 6) ( 7) ( 8) \ ( 9)
(10) s(11) w(12) (13) p(14)
x(15) + (16) <(17) ↑(18) ! (19)
@ (20) - (21) T(22) + (23) ↑(24)
+ (25) - (26) (27) L(28) - (29)
▲(30) ▼(31) (32) !(33) "(34)
...
```

```
7:bell
8:backspace
9:tab
10:new line
27:escape
32:space
```

윈도우에서는 'ㅁ+한자' 키를 누르면 선문자를 입력할 수 있다.

선문자를 이용하면 일반적인 텍스트 모드화면(메모장 등)에서도 표를 그릴 수 있다.

정수값을 문자로 출력하는 방법,

```
...
printf("%c%c%c\n", 1, 22, 2);
...
```

등등 여러 가지 방법이 있다.



을 그려 출력하면 표처럼 선이 붙어 출력된다.

예전에 CUI 모드에서 표나 그림을 그리기 위해 많이 사용되던 방법이다.

◎ 입력 형식

입력 없음

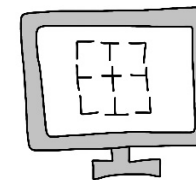
◎ 출력 형식



를 출력한다.

## 입력 예시

## 출력 예시



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 출력하기09

출력문 연습의 마지막 문제이다.

(\*\* 이 문제는 건너 뛰어도 된다.)

(생각과 시도를 많이 해야하는 문제들은 한 두 문제씩 넘겼다가 나중에 풀어보면 된다.)

이번에는 다음과 같은 C프로그램의 소스파일을 출력해보자.

(들여쓰기 부분은 공백(' ') 4개이다.)

```
#include <stdio.h>
main()
{
    printf("Hello World\n");
}
```

위 프로그램을 출력하시오.

### ◎ 입력 형식

입력 없음

### ◎ 출력 형식

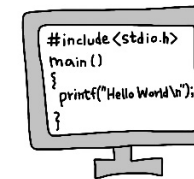
```
#include <stdio.h>
main()
{
    printf("Hello World\n");
}
```

를 그대로 출력한다.

## 입력 예시

## 출력 예시

```
#include <stdio.h>
main()
{
    printf("Hello World\n");
}
```



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 정수 한 개 입력받아 그대로 출력하기

정수형(int)으로 변수를 선언하고, 변수에 정수값을 저장한 후  
변수에 저장되어 있는 값을 그대로 출력해보자.

예시

```
int main()
{
    int n;
    scanf("%d", &n);
    printf("%d", n);

    return 0;
}
```

와 같은 형태로 가능하다.

### ◎ 입력 형식

하나의 정수가 입력된다.

### ◎ 출력 형식

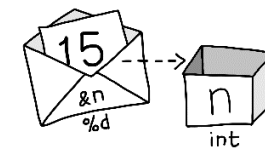
입력받은 정수를 그대로 출력한다.

## 입력 예시

15

## 출력 예시

15



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 문자 한 개 입력받아 그대로 출력하기

문자형(char)으로 변수를 하나 선언하고, 변수에 문자를 저장한 후 변수에 저장되어 있는 문자를 그대로 출력해보자.

참고  
char x;  
scanf("%c", &x);  
printf("%c", x);

와 같은 방법으로 가능하다.

### ◎ 입력 형식

문자 하나가 입력된다.  
(단, 입력되는 문자는 알파벳 대, 소문자 및 숫자를 비롯한 아스키코드로 표현할 수 있는 문자들만 입력된다.)

### ◎ 출력 형식

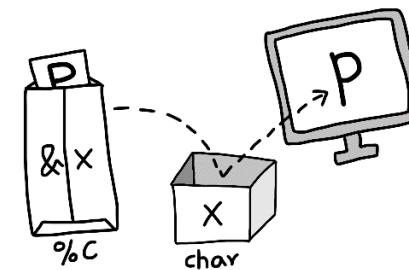
입력받은 문자를 그대로 출력한다.

## 입력 예시

p

## 출력 예시

p





Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 실수 한 개 입력받아 그대로 출력하기

실수형(float)로 변수를 선언하고 그 변수에 실수값을 저장한 후 저장되어 있는 실수값을 출력해보자.

참고

```
float x;
```

```
scanf("%f", &x);
```

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

와 같은 방법으로 가능하다.

### ◎ 입력 형식

실수값 하나가 입력된다.

(단, float로 저장할 수 있는 실수 값들만 입력된다.)

### ◎ 출력 형식

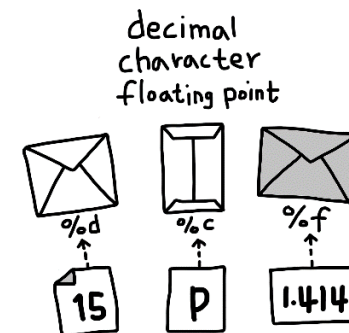
입력받은 실수값을 출력한다.

## 입력 예시

1.414

## 출력 예시

1.414000



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 정수 두 개 입력받아 그대로 출력하기

정수(int) 2개를 입력받아 그대로 출력해보자.

참고

```
int a, b;  
scanf("%d%d", &a, &b); //엔터/공백으로 입력 데이터가 구분되어 입력  
printf("%d %d", a, b);
```

와 같은 방법으로 가능하다.

### ◎ 입력 형식

두 개의 정수가 공백으로 구분되어 입력된다.

### ◎ 출력 형식

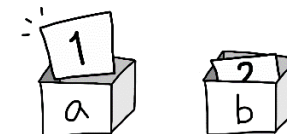
입력받은 두 정수를 공백으로 구분하여 출력한다.

## 입력 예시

1 2

## 출력 예시

1 2



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 문자 두 개 입력받아 순서 바꿔 출력하기

두 개의 문자(ASCII CODE)를 입력받아서 순서를 바꿔 출력해보자.

참고

```
char x, y;
```

```
scanf("%c%c", &x, &y);
```

```
printf("%c %c", y, x); //출력되는 순서를 작성
```

와 같은 방법으로 해결할 수 있다.

### ◎ 입력 형식

두 개의 문자가 공백으로 구분되어 입력된다.

### ◎ 출력 형식

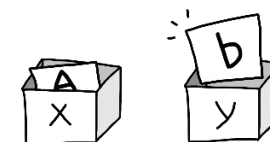
두 문자의 순서를 바꿔 출력한다.

## 입력 예시

A b

## 출력 예시

b A



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 실수 입력받아 둘째 자리까지 출력하기

실수(float) 한 개를 입력받아 저장한 후,  
저장되어 있는 값을 소수점 셋째 자리에서 반올림하여  
소수점 이하 둘째 자리까지 출력하시오.

참고

%.3f 와 같은 형식으로 지정하면,  
소수점 이하 넷째 자리에서 반올림하여 소수점 이하 셋째 자리까지 출력하라는 의미이다.

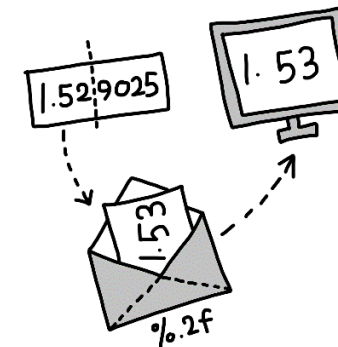
### ◎ 입력 형식

실수 하나가 입력된다.

### ◎ 출력 형식

결과를 소수점 셋째 자리에서 반올림 해, 소수점 이하 둘째 자리까지 출력한다.  
(%.2f를 이용하면 된다.)

입력 예시1	출력 예시1	입력 예시2	출력 예시2
1.59254	1.59	1.529025	1.53





Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

- 01 .....
- 02 .....
- 03 .....
- 04 .....
- 05 .....
- 06 .....
- 07 .....
- 08 .....
- 09 .....
- 10 .....
- 11 .....
- 12 .....
- 13 .....
- 14 .....
- 15 .....
- 16 .....
- 17 .....
- 18 .....
- 19 .....
- 20 .....
- 21 .....
- 22 .....
- 23 .....
- 24 .....
- 25 .....
- 26 .....
- 27 .....
- 28 .....
- 29 .....

## 정수 한 개 입력받아 세 번 출력하기

int형 정수 하나를 입력받아 공백을 사이에 두고 3번 출력해보자.

참고

```
printf("%d %d %d", a, a, a);
```

와 같은 방법으로 출력할 수 있다.

### ◎ 입력 형식

정수 하나가 입력된다.

### ◎ 출력 형식

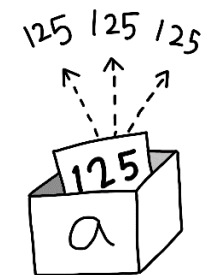
입력받은 정수를 공백으로 구분해 세 번 출력한다.

## 입력 예시

125

## 출력 예시

125 125 125



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 시간 입력받아 그대로 출력하기

어떤 형식에 맞추어 시간이 입력될 때, 그대로 출력하는 연습을 해보자.

참고

scanf( ) 함수는 지정한 형식(format)에 따라 입력받는 함수이다.

따라서, 입력받고자 하는 형식으로 표현해주면 된다.

예시

```
scanf("%d:%d", &h, &m); //콜론(:) 기호를 기준으로 두 수가 각 변수에 저장된다.
```

### ◎ 입력 형식

시(hour)와 분(minute)이 ":" 으로 구분되어 입력된다.

### ◎ 출력 형식

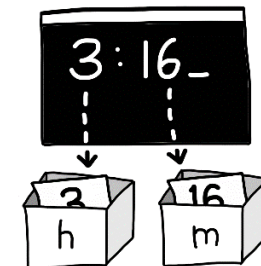
입력받은 시간을 "시:분" 형식으로 출력한다.

## 입력 예시

3:16

## 출력 예시

3:16



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 연월일 입력받아 그대로 출력하기

년, 월, 일을 입력받아 지정된 형식으로 출력하는 연습을 해보자.

### ◎ 입력 형식

년, 월, 일이 ".(닷)"으로 구분되어 입력된다.

### ◎ 출력 형식

입력받은 년, 월, 일을 yyyy.mm.dd 형식으로 출력한다.

(%02d를 사용하면 2칸을 사용해 출력하는데, 1자리 수인 경우 앞에 0이 붙어서 출력된다.)

입력 예시1

출력 예시1

입력 예시2

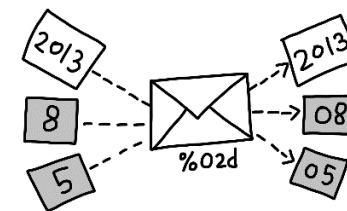
출력 예시2

2012.11.21

2012.11.21

2013.8.5

2013.08.05



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 주민번호 입력받아 형태 바꿔 출력하기

주민번호는 다음과 같이 구성된다.

XXXXXX-XXXXXX

앞의 6자리는 생년월일(yyymmdd)이고 뒤 7자리는 성별, 지역, 오류검출코드이다.  
주민번호를 입력받아 형태를 바꿔 출력해보자.

### ◎ 입력 형식

주민번호 앞 6자리와 뒤 7자리가 " - " 로 구분되어 입력된다.

(입력값은 가상의 주민번호이다.)

ex) 110011-0000000

### ◎ 출력 형식

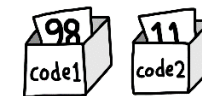
"-"를 제외한 주민번호 13자리를 모두 붙여 출력한다.

### 입력 예시

980907-1121112

### 출력 예시

9809071121112



"%d-%d" -->	98-11
"%d %d" -->	98 11
"%d%d" -->	9811



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 단어 한 개 입력받아 그대로 출력하기

하나의 단어를 입력받아 그대로 출력해보자.

### ◎ 입력 형식

한 단어가 입력된다.(단, 단어의 길이는 50자 이하이다.)

문자를 50개 저장하기 위해서는 `char data[51]` 로 선언하면 된다.

```
char data[51]="";
scanf("%s", data);
```

를 실행하면, `data[51]` 에 한 단어가 저장된다.

### ◎ 출력 형식

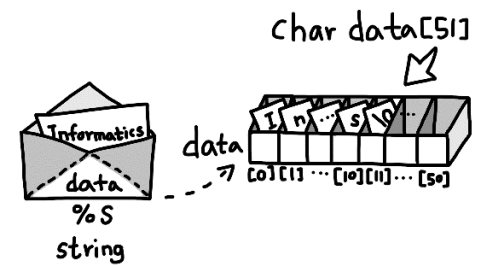
입력 받은 단어를 그대로 출력한다.

## 입력 예시

Informatics

## 출력 예시

Informatics



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 문장 한 개 입력받아 그대로 출력하기

공백이 포함되어 있는 한 문장을 입력받아 그대로 출력하는 연습을 해보자.

### ◎ 입력 형식

공백이 포함되어 있는 한 문장이 입력된다.

입력되는 문장은 여러 개의 단어로 구성되고, 엔터로 끝나며,  
최대 길이는 2000 문자를 넘지 않는다.

예시

```
char data[2001];  
fgets(data, 2000, stdin);  
printf("%s", data);
```

위와 같이 실행하면 공백이 포함된 문장을 키보드(stdin)로 입력받아  
마지막 2000자까지 data[] 공간에 문자들(영문)로 저장하고 출력할 수 있다.

%s 는 문자열(string, 실로 연결된 문자들 즉 문장)로 출력하라는 의미이다.  
최대 50개 문자로 구성된 문장을 저장하기 위해서는 char data[51] 로 선언하면 된다.

fgets( ) 를 사용하면 공백문자가 있는 문장을 입력받아 저장할 수 있다.  
scanf("%s", ... ) 를 사용하면 공백문자 있는 문장을 저장할 수 없다.

```
char data[51] = ""; //영문자를 최대 50 개까지 저장 가능  
scanf("%s", data);
```

를 실행하면, data[51]에 한 단어가 저장된다.

### ◎ 출력 형식

한 문장을 그대로 출력한다.

## 입력 예시

C++ programming is very fun!!

## 출력 예시

C++ programming is very fun!!



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 실수 한 개 입력받아 부분별로 출력하기

실수 한 개를 입력받아 정수 부분과 실수 부분으로 나누어 출력한다.

\*\* 여러 가지 방법들로 해결할 수 있다.

만약 실수 부분이 0으로 시작하지 않는다면(예를 들어 1.000009)  
scanf("%d.%d", &a, &b)도 하나의 방법이 될 수 있다.

### ◎ 입력 형식

실수 한 개가 입력된다.

(단, 입력값은 절댓값이 10000을 넘지 않으며, 소수점 이하 자릿수는 최대 6자리까지이고 0이 아닌 숫자로 시작한다.)

### ◎ 출력 형식

첫 번째 줄에 정수 부분을, 두 번째 줄에 실수 부분을 출력한다.

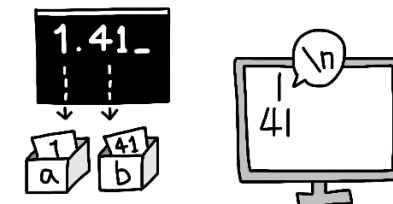
## 입력 예시

1.414213

## 출력 예시

1

414213



Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 단어 한 개 입력받아 나누어 출력하기

단어를 한 개 입력받는다.

입력받은 단어(영어)의 각 문자를 한 줄에 한 문자씩 분리해 출력한다.

참고

C언어에서는 문장을 저장하기 위해 특별한 방법을 사용하지 않고

배열이라는 연속된 공간에 문자들을 연속으로 한 개씩 저장하는 방법을 사용한다.

단, 문자열(문자들이 연속적으로 연결된 줄이라는 의미)의 마지막임을 나타내기 위해 문자의 마지막에 널(NULL) 문자를 삽입한다.

널(NULL)은 영어 단어 자체의 의미로는 '아무런 가치가 없는' 또는 '아무 것도 없는', '사용되지 않은' 등의 의미를 가지는데, 문자로는 '\0', 아스키 코드값 0 또는 NULL 로도 나타낼 수 있다.

단어나 문장을 `scanf("%s", ...);` 로 입력받게 되면,  
그 마지막에 널문자가 자동으로 입력되는 것이다.

그러므로, 본 문제에서는 그렇게 저장되어 있는 영문자들을 한 개씩 순차적으로 검사해서 문자의 마지막을 나타내는 널(NULL) 문자가 나올 때까지 모양을 만들어 출력하면 된다.

조건문, 반복문, 배열에 대해서 배우면 명확히 이해할 수 있게 된다.

`printf("%s", ...);` 는 결국,  
널문자가 나올 때까지 한 문자 씩 순서대로 출력하라는 의미를 가진다.

예시

```
char d[30]; //최대 30문자를 저장할 수 있는 저장 공간 준비. 마지막에 널문자 고려해야 함
scanf("%s", d); //키보드로 입력된 내용을 저장. 단 공백문자가 있으면 그 이전까지만 입력됨
for(i=0; d[i]!='\0'; i++) //저장된 문자를 하나씩 검사해서 널문자가 아닌 동안 반복 실행
{
    printf("%c\n", d[i]);
}
```

### ◎ 입력 형식

단어(영어) 하나를 입력받는다.(단, 단어의 길이는 20자 이하이다.)

### ◎ 출력 형식

단어의 문자(영어)를 하나씩 나누어 한 줄에 한 개씩 ' '로 묶어서 출력한다.

### 입력 예시

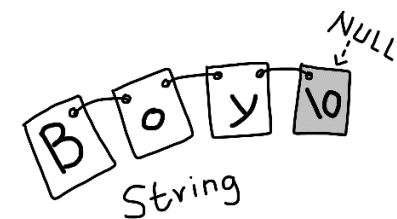
Boy

### 출력 예시

'B'

'o'

'y'





Thought && ( Idea || Curiosity )

---

( Offline Coding ) || ( Kernel Codes )

01	
02	
03	
04	
05	
06	
07	
08	
09	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	

## 정수 한 개 입력받아 나누어 출력하기

다섯 자리의 정수 한 개를 입력받아 각 자리별로 나누어 출력한다.

참고

```
scanf("%1d%1d%1d%1d%1d", &a, &b, &c, &d, &e);
```

를 사용하면 한 개의 숫자로 각각 구분되어 저장된다.

예시

읽어들인 값을 적당히 바꿔 출력하기 위해서

```
printf("[%d]\n", a*10000);
```

과 같은 방법도 가능하다.

### ◎ 입력 형식

다섯 자리로 이루어진 한 개의 정수를 입력받는다.

(단,  $10,000 \leq$  입력받는 수  $\leq 99,999$  )

### ◎ 출력 형식

각 자리의 수를 분리해 한 줄에 하나씩 [ ]속에 넣어 출력한다.

## 입력 예시

75254

## 출력 예시

[70000]

[5000]

[200]

[50]

[4]

