

프로그래밍 언어 - C

최백준 choi@startlink.io

scanf/printf

C

- scanf, printf는 입출력을 받는 함수

포맷 문자열

C

- %d: 정수
- %i: 정수 (형태에 따라 다르게 입력 받음)
- %x: 16진수 정수
- %o: 8진수 정수
- %s: 문자열
- %c: 문자
- %f: float
- %lf: double
- %Lf: long double

%i

C

- %i는 수의 형태에 따라 다르게 입력 받는다.

```
#include <stdio>
```

```
int main() {
```

```
    int x,y;
```

```
    scanf("%d %i",&x,&y);
```

```
    printf("%d %d\n",x,y);
```

```
    return 0;
```

```
}
```

%i

C

- 10 10 -> 10 10
- 10 010 -> 10 8
- 10 0x10 -> 16

%i

C

- 연습문제: <https://www.acmicpc.net/problem/11816>

scanf의 리턴값

C

- scanf의 리턴값은 성공적으로 입력받은 인자의 개수이다.
- 따라서, 파일의 끝까지 입력받아야 하는 경우에는
- `while (scanf("%d %d",&a,&b) == 2)`
- 와 같이 사용할 수 있다.
- 연습 문제: <https://www.acmicpc.net/problem/10951>

공백과 줄바꿈은 무시한다

C

```
for (int i=0; i<5; i++) {  
    scanf("%d", &n);  
}
```

- 모두 같은 값이 들어간다

- 입력

10 20 30 40 50

- 입력

10

20

30

40

50

%C

C

입력

3

A B C

D E F

G H I

출력

10 65 32

66 32 67

10 68 32

```
#include <stdio>
int main() {
    int n;
    scanf("%d",&n);
    while (n--) {
        char x, y, z;
        scanf("%c%c%c",&x,&y,&z);
        printf("%d %d %d\n",x,y,z);
    }
    return 0;
}
```

%C

10

C

입력

3

A B C

D E F

G H I

출력

10 65 66

32 67 68

32 69 70

```
#include <stdio>
int main() {
    int n;
    scanf("%d",&n);
    while (n--) {
        char x, y, z;
        scanf("%c %c %c",&x,&y,&z);
        printf("%d %d %d\n",x,y,z);
    }
    return 0;
}
```

%C

C

입력

3

A B C

D E F

G H I

출력

65 66 67

68 69 70

71 72 73

```
#include <stdio>
int main() {
    int n;
    scanf("%d\n",&n);
    while (n--) {
        char x, y, z;
        scanf("%c %c %c\n",&x,&y,&z);
        printf("%d %d %d\n",x,y,z);
    }
    return 0;
}
```

%C

12

C

입력

3

A B C

D E F

G H I

출력

65 66 67

68 69 70

71 72 73

```
#include <stdio>
int main() {
    int n;
    scanf("%d",&n);
    while (n--) {
        char x, y, z;
        scanf(" %c %c %c",&x,&y,&z);
        printf("%d %d %d\n",x,y,z);
    }
    return 0;
}
```

%[]

C

- %[123]: 1,2,3만 입력을 받는다.
- `scanf("%[123]", a);`
- 123123123189798785979 을 입력하면
- a에는 1231231231이 들어간다.
- %[^123]: 1,2,3을 제외하고 입력을 받는다.
- `scanf("%[^123]", a);`
- 987654323456789를 입력하면
- a에는 987654이 들어간다.

그대로 출력하기

<https://www.acmicpc.net/problem/11718>

```
scanf("%[^\\n]\\n", s);
```

- 줄바꿈을 입력받지 않기 때문에, 편리한 방법이지만, 각 줄의 앞 뒤에 있는 공백은 무시하고 입력을 받게 된다.
- 따라서, 빈 줄을 입력받을 수 없다.
- 또, 공백으로 시작하는 경우 공백을 무시하고 문자부터 입력받게 된다.
- 이 문제는 위의 두가지 경우가 없기 때문에 사용 가능.
- C/C++: <https://gist.github.com/Baekjoon/edc22d3d680a3bbd7ba7>

그대로 출력하기 2

<https://www.acmicpc.net/problem/11719>

```
scanf("%[^\\n]\\n", s);
```

- 줄바꿈을 입력받지 않기 때문에, 편리한 방법이지만, 각 줄의 앞 뒤에 있는 공백은 무시하고 입력을 받게 된다.
- 따라서, 빈 줄을 입력받을 수 없다.
- 또, 공백으로 시작하는 경우 공백을 무시하고 문자부터 입력받게 된다.
- 이 문제는 위의 두가지 경우가 있기 때문에 사용 불가능.
- C/C++: <https://gist.github.com/Baekjoon/66b4f9451a28dd416a33>

숫자의 합

<https://www.acmicpc.net/problem/11720>

```
scanf("%1d",&x);
```

- %d 사이에 수를 넣으면, 그 길이 만큼 입력을 받게 된다.
- 12345에 1, 2, 3, 4, 5 따로 따로 입력 받을 수 있다.
- C/C++: <https://gist.github.com/Baekjoon/0aba0c466380b9e10c2f>

열 개씩 끊어 출력하기

<https://www.acmicpc.net/problem/11721>

```
scanf("%10s", s);
```

- %s의 경우도 개수를 지정해서 입력받을 수 있다.
- 만약, 입력받을 수 있는 것의 개수가 지정한 개수 보다 적으면 그만큼만 입력을 받게 된다.
- C/C++: <https://gist.github.com/Baekjoon/90de3b2f72d6300e3e5e>



C

```
#include <stdio>
int main() {
    int x,y;
    scanf("%d %*d %d",&x,&y);
    printf("%d %d\n",x,y);
    return 0;
}
```

- *을 붙이면 입력은 받지만 변수에 저장하지는 않는다.