



# 시스템프로그래밍기초 실습 2주차

---

Ch1.

# printf() 형식지정자

- printf()
  - 다양한 자료형 변수들을 문자열로 변환해서 화면상에 출력하는 함수

```
printf("Sum is %d.\n", sum)
```

Type	형식지정자	의미
char	%c	문자
int	%d, %i	부호가 있는 정수, 10진수
	%e	부동소수 (지수, 가수부 나눠 표현)
float	%f	부동소수
	%g	e 형식 / f 형식 중 적절한 형식으로 출력
string	%s	문자열

= 3.564e-5  
(scientific e notation)

= 0.00003564  
(real number)

# scanf() 형식지정자

- scanf()
  - 주어진 문자열 스트림 소스에서 지정된 형식으로 데이터를 읽어내는 함수

```
scanf("%d", &sum)
```

Type	형식지정자	의미
char	%c	문자
int	%d, %i	부호가 있는 정수, 10진수
float	%f	단일 정밀도 부동소수
long double	%lf, %LF	2배 정밀도 부동소수
string	%s	문자열

# 실습 예제 1) sum.c

- sum.c 작성

```
$ vi sum.c
```

```
spubuntu@sp: ~/sysPro
1 #include <stdio.h>
2
3 int main()
4 {
5     int a, b, sum;
6     printf("Input two integers : ");
7     scanf("%d%d", &a, &b);
8     sum = a + b;
9     printf("%d + %d = %d\n", a, b, sum);
10    return 0;
11 }
```

- 컴파일 및 실행

```
spubuntu@sp:~/sysPro$ gcc -o sum sum.c
spubuntu@sp:~/sysPro$ ./sum
Input two integers : 3 4
3 + 4 = 7
```

# output 파일(binary) 생성

- compile

```
$ gcc -o [결과파일명] [소스코드(.c)]
```

- 예시

```
$ gcc -o sum sum.c
```

# 실습 예제 2) consecutive\_sum1.c

- while문

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int i = 1, sum = 0;
6     while (i <= 5) {
7         sum += i;
8         i++;
9         printf("sum = %d\n", sum);
10    }
11    return 0;
12 }
13
```

- 실행결과

```
spubuntu@sp:~/sysPro$ ./consecutive_sum1
sum = 1
sum = 3
sum = 6
sum = 10
sum = 15
```

# 실습 예제 3) consecutive\_sum2.c

- for문

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int i, sum = 0;
6
7     for (i = 1; i <= 5; i++){
8         sum += i;
9         printf("for loop sum = %d\n", sum);
10    }
11    return 0;
12 }
13
```

- 실행결과

```
spubuntu@sp:~/sysPro$ ./consecutive_sum2
for loop sum = 1
for loop sum = 3
for loop sum = 6
for loop sum = 10
for loop sum = 15
```

# 실습 예제 4) function\_sum.c

- if/else문, function call

```
1 #include <stdio.h>
2
3 int func_sum(int a, int b)
4 {
5     return a + b;
6 }
7
8 int main()
9 {
10     int a, b, sum;
11
12     printf("Input two integers : ");
13     scanf("%d%d", &a, &b);
14
15     sum = func_sum(a, b);
16     if (sum < 0) {
17         printf("a + b : %d, 0보다 작습니다.\n", sum);
18     }
19     else if (sum == 0) {
20         printf("a + b : %d, 0입니다.\n", sum);
21     }
22     else {
23         printf("a + b : %d, 0보다 큼니다.\n", sum);
24     }
25
26     return 0;
27 }
28
```

- 실행결과

```
spubuntu@sp:~/sysPro$ ./function_sum
Input two integers : 19 4881
a + b : 4900, 0보다 큼니다.
spubuntu@sp:~/sysPro$ ./function_sum
Input two integers : -578 55
a + b : -523, 0보다 작습니다.
spubuntu@sp:~/sysPro$ ./function_sum
Input two integers : 31 -31
a + b : 0, 0입니다.
```



# 실습 과제 1) odd\_even\_sum.c

```
1 #include <stdio.h>
2
3 int odd_sum(int onum)
4 {
5     int result=0;
6
7     /* 이 부분을 작성하시오
8      * 반복문은 while문 사용
9      */
10
11     return result;
12 }
13
14 int even_sum(int onum)
15 {
16     int result=0;
17
18     /* 이 부분을 작성하시오
19      * 반복문은 for문 사용
20      */
21
22     return result;
23 }
24
25 int main()
26 {
27     int onum;
28     printf("홀수, 짝수 덧셈을 합니다. 1부터 어느 수 까지 할까요?\n");
29     scanf("%d", &onum);
30
31     printf("1부터 %d까지 홀수의 합 : %d\n", onum, odd_sum(onum));
32     printf("1부터 %d까지 짝수의 합 : %d\n", onum, even_sum(onum));
33
34     return 0;
35 }
36
```

- mod 연산자(%)를 사용할 것.

- $7 \% 2 == 1$
- $6 \% 2 == 0$
- $13 \% 4 == 1$

# 실습 과제 1) odd\_even\_sum.c 결과

```
spubuntu@sp:~/sysPro$ ./odd_even_sum
홀수, 짝수 덧셈을 합니다. 1부터 어느 수 까지 할까요?
50
1부터 50까지 홀수의 합 : 625
1부터 50까지 짝수의 합 : 650
spubuntu@sp:~/sysPro$ ./odd_even_sum
홀수, 짝수 덧셈을 합니다. 1부터 어느 수 까지 할까요?
100
1부터 100까지 홀수의 합 : 2500
1부터 100까지 짝수의 합 : 2550
```

# 과제 제출 파일

1. sum.c
2. sum
3. consecutive\_sum1.c
4. consecutive\_sum1
5. consecutive\_sum2.c
6. consecutive\_sum2
7. function\_sum.c
8. function\_sum
9. odd\_even\_sum.c
10. odd\_even\_sum

# 과제 제출 형식(1) (실습 시간 내에 완료 못한 경우)

1. 모든 파일은 sys\_02\_자기학번.tar.gz으로 압축하여 제출한다.

- tar.gz 압축하기

```
$ tar -zcvf [sys_02_자기학번.tar.gz] [폴더명]
```

- tar.gz 압축풀기

```
$ tar -zxvf [압축파일명.tar.gz]
```

옵션	설명
-c	파일을 tar로 묶음
-p	파일 권한을 저장
-v	묶거나 파일을 풀 때 과정을 화면으로 출력
-f	파일 이름을 지정
-C	경로를 지정
-x	tar 압축을 풀
-z	gzip으로 압축하거나 해제함



## 과제 제출 형식(2) (실습 시간 내에 완료 못한 경우)

### 2. 이메일 제목 :

[시프기]\_02\_이름\_학번

형식 다를 시,  
메일 수신 불가.

양인식 (3공 318호) : [inshik@hanyang.ac.kr](mailto:inshik@hanyang.ac.kr)

조윤수 (4공 412호) : [suejoe@hanyang.ac.kr](mailto:suejoe@hanyang.ac.kr)



# 감사합니다.

---