

## <고급 C프로그래밍 및 실습> 9장 포인터 실습 문제

### ※ 문제에 대한 안내

- 특별한 언급이 없으면 문제의 조건에 맞지 않는 입력은 입력되지 않는다고 가정하라.
- 특별한 언급이 없으면, 각 줄의 맨 앞과 맨 뒤에는 공백을 출력하지 않는다.
- 출력 예시에서 □는 각 줄의 맨 앞과 맨 뒤에 출력되는 공백을 의미한다.
- 입출력 예시에서 ↳ 이 후는 각 입력과 출력에 대한 설명이다.

### ★★★★★ 중 요 사 항 ★★★★★

※ 문제 설명에서 ► 로 표시된 문장은 “포인터 사용을 강제”하기 위한 제약 조건으로, 이에 주의하여 프로그램을 작성하시오.

(인증 시험에서 조건을 지키지 않고 프로그램 작성 시, 감점 or 0점 처리됩니다.)

1절 [ 문제 1][레벨 0] 다음과 같은 순서에 따라 프로그램을 작성하시오.

- 1) 포인터 변수 px를 이용하여 사용자로부터 값을 입력받아 x에 저장 (즉, scanf 함수의 매개변수로 포인터 변수 px 이용, 변수 x 사용 금지). 변수 y, z도 동일한 방법으로 값을 입력 받아 저장한다.
- 2) px, py, pz에 있는 주소들을 py, pz, px로 이동시킨다.
- 3) 순서가 바뀐 최종 값을 포인터를 이용하여 출력하시오.
- 4) 아래 코드를 사용하시오.

```
int x, y, z;  
int *px, *py, *pz, *tmp ; // tmp는 포인터이다
```

입력 예시 1

출력 예시 1

1 2 3 ↳ x

3 1 2 ↳ 한 칸씩 오른쪽으로 밀기

2절 [ 문제 2 ][레벨 1] 문자 #을 포함한 임의의 개수(1 이상 20 이하)의 문자를 입력 받은 후, 첫 번째 # 이전까지의 문자들을 포인터를 사용하여 역순으로 출력하시오. (#이 두 개 있는 경우도 있음)

► 배열 표기 [ ] 는 사용 금지 (포인터 표기 사용) <-- 추가된 조건

즉, char ch[20]; 배열 선언하고, 이 다음부터 배열 표기 [ ] 사용 금지

입력 예시 1

출력 예시 1

duck#duck

kcud

입력 예시 2

출력 예시 2

duck#pond

kcud

**2절 [ 문제 3 ][레벨 0]** N개의 정수를 입력 받은 후, 0 이 나오기 전까지의 정수 개수를 출력하는 프로그램을 작성하시오. ( $N \leq 50$ )

- 0 은 오직 한 번만 나타난다.

▶ 배열 선언 이후, 배열 표기 [ ] 는 사용 금지 (포인터 표기 사용)

▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (강의자료 2주차 2강 p. 9 참조)  
 <-- 추가 조건, (예) for( p = ar ; p < ar+5 ; p++ ) printf("%d\\n", \*p);

```
int ar[50]={0}, N ;
int *p ;

scanf("%d", &N);

for (p = ar; p < ar+N; p++) 반복문으로 배열 훑어보기 ⇨ 주소를 이용하여 반복문 구현
    scanf("%d", p);
```

```
int ar[50]={0}, N, i ;
int *p ;

scanf("%d", &N);

for (i=0; i < N; i++) 이렇게 해도 실행되지만, 인증시험에서는 감점이 됩니다.
    scanf("%d", ar+i);
```

입력 예시 1

출력 예시 1

10      ↳ N	6
4 5 8 9 8 1 0 1 9 3	

입력 예시 2

출력 예시 2

5	0
0 1 3 -2 -4	

**2절 [ 문제 4 ][레벨 0]** 크기 3의 int형 배열을 사용자 입력을 통해 초기화 한 후 중앙값을 출력하는 프로그램을 작성하시오.

▶ 배열 선언 이후, 배열 표기 [ ] 는 사용 금지 (포인터 표기 사용)

즉, int x[3]; 배열 선언하고, 이 다음부터 배열 표기 [ ] 사용 금지

▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (강의자료 2주차 2강 p. 9 참조)

(예) for( p = ar ; p < ar+5 ; p++ ) printf("%d\\n", \*p);

- 중앙값(median)이란 오름차순으로 정렬했을 때 순서가 중앙에 해당하는 숫자이다.

- (예시 1)에서 1 9 7인 경우 오름차순으로 정렬하면 1 7 9 가 되고 중앙에 있는 수는 7 이다.

- 입력이 1 1 2인 경우 1을 출력한다. 입력이 1 2 2인 경우 2를 출력한다.

입력 예시 1

출력 예시 1

1 9 7	7
-------	---

입력 예시 2

-7 10 0

출력 예시 2

0

**2절 [ 문제 5 ][레벨 1]** 5개의 정수를 입력 받아 배열 arr[ ]에 저장하고, 각 정수들의 내림차순 상의 순서를 배열 rank[ ]에 저장 한 후 출력하는 프로그램을 작성 하시오.

- ▶ 배열 선언 이후, 배열 표기 [ ] 는 사용 금지 (포인터 표기 사용)
- ▶ 입출력 시에도 포인터 사용
- ▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (강의자료 2주차 2강 p. 9 참조)  
(예) for( p = ar ; p < ar+5 ; p++ ) printf("%d\n", \*p);
- (예시 2)에서 1 6 4 9 6의 역순은 9 6 6 4 1 이 되고 2등이 2개이며,  
2등이 2개인 경우 3등은 없고 4등부터 계속 된다. (즉 같은 등수가 여러 개 있는 경우이다)

입력 예시 1

1 2 3 4 5

출력 예시 1

1=r5 2=r4 3=r3 4=r2 5=r1

입력 예시 2

1 6 4 9 6

출력 예시 2

1=r5 6=r2 4=r4 9=r1 6=r2

**2절 [ 문제 6 ][레벨 1]** 10개의 영문자를 입력받은 후 가장 빈도수가 높은 문자와 그 문자의 빈도수를 출력하시오. 문자의 빈도수가 같은 경우 먼저 나오는 문자를 출력하시오.

- ▶ 배열 선언 이후, 배열 표기 [ ] 는 사용 금지 (포인터 표기 사용)  
즉, char ch[10]; 배열 선언하고, 이다음부터 배열 표기 [ ] 사용 금지
- ▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (강의자료 2주차 2강 p. 9 참조)  
(예) for( p = ar ; p < ar+5 ; p++ ) printf("%d\n", \*p);

입력 예시 1

domination

출력 예시 1

o 2

**4절 [ 문제 7 ][레벨 0]** 정수 3개를 사용자로부터 입력 받아서, 화면에 출력하시오.

- input( ) 함수를 정의하여 사용하시오.
  - 인자: 세 개의 int형 포인터
  - 정수 3개를 사용자로부터 입력 받는다.
  - input( ) 함수에서는 함수 인자 이외에 어떤 변수도 사용하지 않는다.
  - 반환 값: 없음
- output( ) 함수를 정의하여 사용하시오.
  - 인자: 세 개의 int형 포인터
  - 정수 3개를 화면에 출력 한다.
  - output( ) 함수에서는 함수 인자 이외에 어떤 변수도 사용하지 않는다.
  - 반환 값: 없음

- 아래 코드를 사용시오.

```
int main( ){
    int x, y, z;
    input( ... );
    output( ... );
}
```

입력 예시 1

출력 예시 1

1 2 3

1 2 3

**4절 [ 문제 8 ][레벨 1]** N개의 정수를 입력 받아 배열에 저장한 후 ( $N \leq 50$ ), 다시 배열에서 값을 교환하고 싶은 인덱스 a 와 b를 입력 받는다. 인덱스 a 와 b 위치의 요소 값들이 교환된 배열을 출력하는 프로그램을 작성시오.

※ 8번 문제는 함수를 연습하는 문제로 7번 문제와 달리 포인터 사용을 위한 제약 조건이 없다.

※ 포인터 사용을 연습하기 위하여 9장 모든 함수 실습문제에서 전역변수 사용금지

- 인덱스 a, b는 상대적 크기에 상관없다.  $a < b$  혹은  $a > b$
- swap( ) 함수를 정의하여 사용시오.
  - 인자: 두 개의 int 포인터
  - 반환 값: 없음
  - 인자가 가리키는 두 변수의 값 교환
- 입출력은 main( ) 함수에서 수행

입력 예시 1

출력 예시 1

6        ↦ N  
3 2 0 1 4 6  
2 4        ↦ a b

□ 3 2 4 1 0 6

**4절 [ 문제 9 ][레벨 1]** 크기 3의 int형 배열을 사용자 입력을 통해 초기화 한 후 중앙값을 출력하는 프로그램을 mid 함수를 사용하여 작성시오.

- mid( ) 함수를 다음과 같이 정의하여 사용시오.
  - 인자: int 배열
  - 배열에 저장된 세 개의 정수 중 중앙값을 찾아내어 중앙값의 위치를 반환 (세 개의 정수를 정렬하는 등 배열 내 정수의 이동 없이 중앙값을 찾아 낼 것)
  - 반환 값: 중앙값이 저장된 배열 원소의 주소
- 입출력은 main( ) 함수에서 수행
- 중앙값(median)이란 오름차순으로 정렬했을 때 순서가 중앙에 해당하는 숫자이다.
- (예시 1)에서 1 9 7인 경우 오름차순으로 정렬하면 1 7 9 가 되고 중앙에 있는 수는 7 이다.
- 입력이 1 1 2인 경우 1을 출력한다. 입력이 1 2 2인 경우 2를 출력한다.

- ▶ 배열 선언 이후, 배열 표기 [ ] 는 사용 금지 (포인터 표기 사용)  
 즉, `int x[3];` 배열 선언하고, 이 다음부터 배열 표기 [ ] 사용 금지

입력 예시 1

출력 예시 1

1 9 7	7
-------	---

입력 예시 2

출력 예시 2

-7 10 0	0
---------	---

**[ 문제 10 ][레벨 1]** 최대 공약수와 최소 공배수를 포인터를 사용하여 전달하는 프로그램을 작성하시오.

1) `gcdlcm( )` 함수

- 인자: `int`형 변수 두 개, `int`형 포인터 2개  
 함수(`int`형 변수, `int`형 변수, `int`형 포인터, `int`형 포인터)
- 정수값 2개를 넘겨받아 최대 공약수와 최소 공배수를 계산한 후, 포인터를 사용하여 전달  
 1도 최대공약수가 될 수 있음. 입력된 정수도 최대공약수가 될 수 있음
- 반환 값: 없음

2) `main` 함수

- 정수 두 개를 사용자로부터 입력 받는다.
- 함수를 호출한다.
- 최대 공약수와 최소 공배수를 화면에 출력한다.

입력 예시 1

출력 예시 1

6 10	2 30
------	------

입력 예시 2

출력 예시 2

72 36	36 72
-------	-------

**4절 [ 문제 11 ][레벨 1]** 포인터를 사용하여 문자 6개를 복사하는 함수를 만들고 출력하시오.

- ▶ 배열 선언 이후, 배열 표기 [ ] 는 사용 금지 (포인터 표기 사용) (배열 이름은 사용해도 됨)
- 문자열을 복사하는 함수 `strcpy( )`를 정의하여 사용하시오.
    - 인자: 두 개의 문자 배열 `a[ ]` 와 `b[ ]`
    - 배열 `a[ ]`에 담긴 문자 6개를 배열 `b[ ]`에 복사
    - 반환 값: 없음
  - 입출력은 `main( )` 함수에서 수행
    - 입출력 시에 포인터 표기 사용

입력 예시 1

출력 예시 1

beyond

beyond

**4절 [ 문제 12 ][레벨 1]** N개의 정수를 두 번 입력받아 ( $N \leq 20$ ), 역방향으로 더하여 출력하는 프로그램을 작성하시오.

- ▶ 배열 선언 이후, 배열 표기 [ ] 는 사용 금지 (포인터 표기 사용)
- ▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (강의자료 2주차 2강 p. 9 참조)  
(예) `for( p = ar ; p < ar+5 ; p++ ) printf("%d\n", *p);`

- `addArray( )` 함수를 이용하여 작성하시오.
  - 인자: 세 개의 배열을 나타내는 세 개의 `int` 포인터, 배열의 크기를 나타내는 정수
  - 첫 번째와 두 번째 포인터가 나타내는 배열의 수를 역방향으로 더해서, 세 번째 포인터가 나타내는 배열에 저장
  - 반환 값: 없음
- `main( )` 함수
  - 입출력 수행
  - `addArray( )` 함수를 한번만 호출한다.

입력 예시 1

출력 예시 1

3      ↳ 배열 크기 N=3  
1 2 3  
5 10 15

□ 16 12 8    ↳ 16=1+15, 12=2+10, 8=3+5

입력 예시 2

출력 예시 2

4      ↳ 배열 크기 4  
3 8 9 5  
0 1 -5 6

□ 9 3 10 5

**4절 [ 문제 13 ][레벨 1]** 정수 N, S, E를 입력 받는다. N개의 정수를 입력 받아 ( $N \leq 100$ ) 배열에 저장한다. 배열에 저장된 N개 중에서 인덱스 S에서 시작해서 인덱스 E까지 요소의 총 합을 출력하시오.

- ▶ 배열 선언 이후, 배열 표기 [ ] 는 사용 금지 (포인터 표기 사용)
- ▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (강의자료 2주차 2강 p. 9 참조)  
(예) `for( p = ar ; p < ar+5 ; p++ ) printf("%d\n", *p);`
- `arrsum( )` 함수를 정의하여 사용하시오.
  - 인자: 배열에서 구간 시작 원소와 구간 마지막 원소를 가리키는, `int` 형 포인터 두 개
  - 반환 값: 구간에 해당되는 배열 요소의 총 합
- 입출력은 `main( )` 함수에서 수행

입력 예시 1

5 0 2    ↦ N S E  
3 -10 15 20 27

출력 예시 1

8    ↦    3+(-10)+15=8

**[ 문제 14 ][레벨 1]** N개 정수를 입력 받아 ( $N \leq 100$ ) int형 배열 d[100]에 저장하고, 다음 수식의 값을 출력하는 프로그램을 작성하시오.

$$d[0] + (d[0]+d[1]) + (d[0]+d[1]+d[2]) + \dots + (d[0]+d[1]+d[2]+\dots+d[N-1])$$

1) add\_to\_k() 함수를 정의하여 사용하시오.

- 인자 : int형 포인터 2개
- 첫 번째 포인터가 가리키는 배열의 원소부터 두 번째 포인터가 가리키는 배열의 원소까지, 배열의 원소 값을 더한 합을 계산한다.
- 반환 값 : 배열의 원소 값을 더한 합

2) main( ) 함수를 다음과 같이 작성하시오.

- 입출력은 main( ) 함수에서 수행한다.
- add\_to\_k() 함수를 반복 호출하여 수식의 값을 구한다.

입력 예시 1

5    ↦ N = 5  
1 2 3 4 5

출력 예시 1

35

입력 예시 2

6  
6 5 0 7 2 1

출력 예시 2

87

**4절 [ 문제 15 ][레벨 1]** 10개의 정수를 입력으로 받아, 가장 큰 수부터 내림차순으로 정렬하여 출력하는 프로그램을 다음과 같은 방법으로 작성 하시오.

- ▶ 배열 선언 이후, 배열 표기 [ ] 는 사용 금지 (포인터 표기 사용)
- ▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (강의자료 2주차 2강 p. 9 참조)  
(main( ) 함수와 ABC( ) 함수에 이 조건이 적용됨)

- main( ) 함수를 다음과 같이 작성한다.

- 입력되는 정수를 int 형 배열에 저장한다.
- 배열에서 조사 시작위치의 주소와 길이를 인수로 하여, 함수 ABC( )의 호출을 9번 반복한다. 매 호출 시, 길이는 10, 9, 8, .... 2 로 변한다.

- 배열에 저장된 정수를 순서대로 출력한다.
- ABC( ) 함수를 다음과 같이 정의하여 작성한다.
  - 인자: int 배열에서 조사 시작 원소의 주소, 조사할 원소 개수를 나타내는 정수 k
  - 배열에서 조사 대상의 첫 번째 정수부터 마지막 정수(즉, 9번째 정수)중 가장 큰 정수를 찾고, 이를 첫 번째 위치의 정수와 교환한다.
  - 반환 값: 없음

입력 예시 1

출력 예시 1

1 3 5 7 9 2 4 6 8 10	<input type="checkbox"/> 10 9 8 7 6 5 4 3 2 1
----------------------	---

입력 예시 2

출력 예시 2

13 56 27 89 43 76 32 68 91 8	<input type="checkbox"/> 91 89 76 68 56 43 32 27 13 8
------------------------------	---

※ 4번, 9번 중앙값

※ 레벨 숫자의 의미

- [레벨 0]: 레벨1 보다 쉬운 문제
- [레벨 1]: 인증시험에서 가장 쉬운 레벨
- [레벨 2]: 인증시험에서 어려운 레벨
- [레벨 3]: 인증시험에서 매우 어려운 레벨

※ 인증시험 문제 구성 (괄호 안은 점수)

		문제1	문제2	문제3	문제4	문제5
(예시 1)	[레벨 1]	1-1(50)	2-1(50)	3-1(50)	4-1(50)	5-1(50)
	[레벨 2]	2-2(100)	2-2(100)	3-2(100)	4-2(100)	5-2(100)