

## <고급 C프로그래밍 및 실습> 12장 동적 메모리 할당 실습 문제

### ※ 문제에 대한 안내

- 입출력 예시에서 ↳ 이 후는 각 입력과 출력에 대한 설명이다.

**3절 [ 문제 8 ] [레벨 0]** 사용자로부터 정수  $N$ 을 입력받고,  $N$  줄의 공백을 포함하는 문자열을 동적할당을 이용해 입력 받으시오. 그 후, 이 중 가장 길이가 작은 공백을 포함하는 문자열을 출력하는 프로그램을 작성하시오.

- 문자열의 길이는 최대 100 이다.
- 문자열 처리 표준 함수를 사용해도 됨
- 길이가 같은 경우에는 먼저 입력된 문자열을 출력한다.

입력 예시 1

출력 예시 1

4 ↳ N=4 개 문자열 입력 Program Good This is string language	Good
---	------

**3절 [ 문제 9 ] [레벨 1]** 사용자로부터 정수  $N$ 을 입력받고,  $N$  줄의 공백을 포함하는 문자열을 동적할당을 이용해 입력 받으시오. 그 후, 이 중 가장 길이가 긴 문자열부터 가장 길이가 작은 문자열까지 정렬하고, 순서대로 출력하는 프로그램을 작성하시오.

- 문자열의 길이는 최대 100 이다.
- 문자열 처리 표준 함수를 사용해도 된다.
- 같은 길이의 문자열은 없다고 가정한다.
- 각 줄 문자열 저장에 필요한 크기만큼만 메모리 동적할당 한다.

**Hint:** 문자열 교환할 때, 문자열 교환을 할 것인가? 아니면, 문자열을 가리키는 주소만 바꿀 것인가?

입력 예시 1

출력 예시 1

4 ↳ N=4 개 문자열 입력 This is string language Program Good This is string language	This is string language Program Good
---	---

**3절 [ 문제 10 ] [레벨 1]** 사용자로부터 정수 한 개를 입력받는다. 이것을  $N$  이라고 하자.  $N$ 은 홀수라고 가정한다. 정수  $N$ 개를 저장할 수 있는 메모리를 동적 할당한 후에, 이 메모리에 0부터  $N-1$  까지 정수를 저장한다 (0 1 2 ...  $N-1$ ). 이 배열의 중간에 위치한 값을 제외한 나머지 값들을 순서대로,  $M$  ( $N-1$ )개 새로 동적할당된 메모리에 대입한 후 출력한다.

- (예시 1)에서 N이 5이므로 0에서 4까지의 정수, 0 1 ~~2~~ 3 4 를 저장한다. 배열의 중간에 위치한 값을 제외하면 0 1 3 4가 된다.
- 크기 N 동적할당 메모리와, 크기 M 동적할당 메모리를 아용한다.  $M=N-1$
- N의 크기는 최대 20 이다.

입력 예시 1

출력 예시 1

5                      ↦ N	□0□1□3□4
----------------------------	----------

입력 예시 2

출력 예시 2

11	0 1 2 3 4 6 7 8 9 10
----	----------------------

**3절 [ 문제 11 ] [레벨 1]** 사용자로부터 -1을 입력받기 전까지 계속 정수를 입력받는다. 초기 배열의 크기는 5로 설정하되, 사용자가 계속 입력하면 길이를 3씩 증가시킨다. 길이를 증가시킬 때, 새로운 배열을 하나 만들어서 증가시킨다. 사용자가 -1을 입력하면 배열의 값을 모두 출력한다.

- 동적할당(malloc함수)을 아용한다. ~~realloc~~ 쓰지 말 것.
- 최대 증가 시킬 수 있는 배열의 크기는 20으로 제한한다.

입력 예시 1

출력 예시 1

3□4 5 6 7 8 9 -1	□3□4□5□6□7□8□9□-1
------------------	-------------------

입력 예시 2

출력 예시 2

3 4 5 -1	3 4 5 -1
----------	----------

**3절 [ 문제 12 ] [레벨 1]** 정수 N을 입력받고 N개의 문자열을 입력받아 사전 순으로 출력하는 프로그램을 작성하시오.

- malloc( ) 함수를 이용한다.
- 배열을 미리 선언하지 않고 실행 중에 동적할당으로 메모리를 필요한 만큼만 만든다.
- 문자열의 길이는 최대 100 이다.
- 소문자만 입력된다고 가정한다.
- 표준 문자열 함수를 사용해도 된다.

**Hint:** 문자열 교환할 때, strcpy로 **문자열 교환**하면 배열의 크기가 달라서 안 되고, 문자열을 가리키는 **주소만 바꾸어야** 된다.

입력 예시 1

4 apricot peach willow birch	apricot birch peach willow
--	-------------------------------------

출력 예시 1

**4절 [ 문제 13 ] [레벨 0]** 다음과 같은 기능을 하는 학번관리 프로그램을 작성하시오.

- 등록할 학번 수 N을 입력받고, N개의 학번을 동적할당을 이용해 입력 (int형 사용)
- 삭제할 학번 개수 D (D<N)를 입력받고, D만큼 메모리 크기 조정 후 출력
- 마지막에 입력한 학번부터 D 개를 삭제한다.
- 프로그램 종료 시 할당한 메모리 해제
- 힌트: **malloc** 함수 사용 후, **realloc** 함수를 사용하시오.

입력 예시 1

3            ↪    N 16011111 16011123 16011145 2            ↪    D	16011111
--	----------

출력 예시 1

$$N - D = 3 - 2 = 1 \text{ 개}$$

입력 예시 2

4            ↪    N 120111 15011123 16011145 16011300 1            ↪    D	120111 15011123 16011145
--	--------------------------------

출력 예시 2

### 10주차 실습문제

[ 응용1 ] 정수 N과 M을 입력받고, 공백이 포함된 N개의 문자열을 입력받는다. 각 문자열의 첫 번째 문자부터 M번째 문자까지(M번째 문자 포함)의 문자들 중에 영어 알파벳과 공백을 제외한 문자의 개수가 가장 많은 문자열부터 내림차순으로 문자열을 출력하는 프로그램을 아래의 count\_c 함수를 사용하여 작성하시오. 단, M번째 문자까지 영어 알파벳과 공백으로만 이루어진 문자열(영어 알파벳과 공백을 제외한 문자의 개수가 0인 경우)은 출력하지 않는다. (

- $N \leq 10$  이고, 문자열의 최대 길이는 100이다.
- 문자열의 첫 번째 문자를 1번째 문자라 한다. M이 10인 경우, 문자 배열의 [9]번 원소까지를 검사한다.
- M값이 문자열의 길이보다 큰 경우에는 문자열 끝까지만 검사한다.
- "영어 알파벳과 공백을 제외한 문자"는 숫자, 특수문자(!@#\$%^&\*(){}?.,<>`~=-+ 등)이다.
- "영어 알파벳과 공백을 제외한 문자"의 수가 같은 경우 먼저 입력된 문자열을 먼저 출력한다.
- count\_c 함수
  - o 함수원형 : `int count_c(char *p, int M)`
  - o 문자열의 첫 번째 문자의 주소(p)와 M을 인자로 받아, "영어 알파벳과 공백을 제외한 문자의 개수"를 세어 반환한다.
  - o 단, M이 문자열의 마지막 원소의 길이보다 더 큰 경우에는 문자열의 마지막 원소까지, "영어 알파벳과 공백을 제외한 문자의 개수"를 세어 반환한다.
  - o M이 문자열의 길이보다 더 큰지를 count\_c에서 판단함.

#### 입력 예시 1

```
5 15      ↳ N=문자열 줄 수, M=M번째 문자까지 검사
1 year is 365 days.  ↳ 알파벳 공백 제외 4개(1365)
How are you doing?  ↳ 알파벳 공백 제외 0개
**Notice**         ↳ 알파벳 공백 제외 4개(****)
Winter is coming    ↳ 알파벳 공백 제외 0개
1+1=2              ↳ 알파벳 공백 제외 5개

↳ M번째(15번째)까지 문자는 밑줄로 표시함.
```

#### 출력 예시 1

```
1+1=2
1 year is 365 days.
**Notice**

↳ M번째까지 "알파벳 공백 제외 문자 수"
  가 같은 경우 먼저 입력된 문자열 출력
↳ 0개인 경우 출력하지 않음
```

**[ 응용 2 ]** 정수 N과 M을 입력받고, 공백이 포함된 N개의 문자열을 입력받는다. 각 문자열의 첫 번째 문자부터 M번째 문자까지(M번째 문자 포함)의 문자들 중에 영어 알파벳과 공백을 제외한 문자의 개수가 가장 많은 문자열부터 내림차순으로 문자열을 출력하는 프로그램을 아래의 `count_c` 함수를 사용하여 작성하시오. 단, M번째 문자까지 영어 알파벳과 공백으로만 이루어진 문자열(영어 알파벳과 공백을 제외한 문자의 개수가 0인 경우)은 출력하지 않는다.

- $N \leq 10$  이고, 문자열의 최대 길이는 100이다.
- 문자열의 첫 번째 문자를 1번째 문자라 한다. M이 10인 경우, 문자 배열의 [9]번 원소까지를 검사한다.
- M값이 문자열의 길이보다 큰 경우에는 문자열 끝까지만 검사한다.
- "영어 알파벳과 공백을 제외한 문자"는 숫자, 특수문자(!@#\$%^&\*()[]{}?.,<>`~=-+ 등)이다.
- "영어 알파벳과 공백을 제외한 문자"의 수가 같은 경우 먼저 입력된 문자열을 먼저 출력한다.
- `count_c` 함수
  - 함수원형 : `int count_c char *p, char *q`
  - 문자열의 첫 번째 문자의 주소(p)와 검사를 할 마지막 문자의 주소(q)를 인자로 받아, "영어 알파벳과 공백을 제외한 문자의 개수"를 세어 반환한다.
  - 단, q의 값이 문자열의 마지막 원소의 주소보다 더 큰 경우에는 문자열의 마지막 원소까지, "영어 알파벳과 공백을 제외한 문자의 개수"를 세어 반환한다.
  - q의 값이 문자열의 마지막 원소의 주소보다 더 큰지를 `count_c` 함수가 아닌 main 함수에서 판단하는 경우 10% 감점
  - 함수에서 반복문으로 배열을 훑어볼 시, 배열 표기 [ ]를 사용하지 않고, 포인터를 이동시키며 반복문을 구현한다. (`count_c` 함수에서는 `p[i]` 또는 `p+i` 표기 사용금지)
- 단, main 함수에서의 주소 표현 방식은 제한 사항 없다.

입력 예시 1

```
5 15      ↳ N=문자열 줄 수, M=M번째 문자까지 검사
1 year is 365 days.    ↳ 알파벳 공백 제외 4개(1365)
How are you doing?    ↳ 알파벳 공백 제외 0개
**Notice**           ↳ 알파벳 공백 제외 4개(****)
Winter is coming      ↳ 알파벳 공백 제외 0개
1+1=2                ↳ 알파벳 공백 제외 5개

↳ M번째(15번째)까지 문자는 밑줄로 표시함.
```

출력 예시 1

```
1+1=2
1 year is 365 days.
**Notice**

↳ M번째까지 "알파벳 공백 제외 문자 수"
  가 같은 경우 먼저 입력된 문자열 출력
↳ 0개인 경우 출력하지 않음
```

**[ 응용3 ]** 정수 N을 입력받고, 공백이 포함된 N개의 문자열을 입력받는다. 각 문자열마다 영어 알파벳과 공백을 제외한 문자의 개수를 세어, 영어 알파벳과 공백을 제외한 문자의 개수가 가장 많은 문자열을 출력하는 프로그램을 아래의 count\_c 함수를 사용하여 작성하시오.

- 입력되는 문자열의 최대 길이는 100이다.
- "영어 알파벳과 공백을 제외한 문자"는 숫자, 특수문자(!@#\$%^&\*(){}?.,<>`~=-+ 등)이다.
- "영어 알파벳과 공백을 제외한 문자"의 최대 개수가 같은 문자열이 두 개 이상 있는 경우 먼저 입력된 문자열을 출력한다.
- count\_c 함수
  - 함수원형 : `int count_c (char *p, char *q)`
  - 문자열의 첫 번째 문자의 주소(p)와 마지막 문자의 주소(q)를 인자로 받아, "영어 알파벳과 공백을 제외한 문자의 개수"를 세어 반환한다.
  - count\_c 함수에서 반복문으로 배열을 훑어볼 시, 배열 표기 [ ]를 사용하지 않고, 포인터를 이동시키며 반복문을 구현한다. (count\_c 함수에서는 p[i] 또는 p+i 표기 사용금지)
- 단, main 함수에서의 주소 표현 방식은 제한 사항이 없다.

입력 예시 1

5	↳ N=문자열 줄 수
1 year is 365 days.	↳ 알파벳, 공백 제외 5개(1365.)
How are you doing?	↳ 알파벳, 공백 제외 1개(?)
**Notice**	↳ 알파벳, 공백 제외 4개(****)
Winter is coming.	↳ 알파벳, 공백 제외 1개(.)
1+1=2	↳ 알파벳, 공백 제외 5개

출력 예시 1

1 year is 365 days.
↳ 알파벳, 공백 제외 문자 최대 개수가 같은 "1 year is 365 days."와 "1+1=2" 중, 먼저 입력된 문자열 출력