

<고급C 프로그래밍 및 실습> 1차 인증시험대비 문제지

2018.10.17 (수)

학번	학과	이름

※ 문제에 대한 안내

- 문제지는 총 5페이지이고, 총 4문제 400점 만점이고, 문제의 순서는 난이도와 관계없다.
- 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.

[문제 1] 정수 N을 입력받는다. 알파벳과 숫자로만 구성된 N개 문자를 입력받는다. 입력 받은 N개의 문자들에 대해서 대문자와 소문자를 출력하고, 숫자 문자의 경우 숫자 문자를 연결하여 만든 정수에 2를 곱하여 출력하시오.

- 모든 함수에서 (즉 input 함수와 main 함수에서) 반복문으로 배열을 훑어볼 시, 주소를 이용하여 반복문을 구현한다. 즉, 포인터가 배열의 각 원소를 순차적으로 가리키도록 하며, 포인터가 가리키는 위치의 원소에 대해 필요한 작업을 수행한다.
- 배열 선언을 제외하고 배열 기호 []를 사용하지 않는다.
- 입력되는 N개 문자 안에 대문자 최소 1개, 소문자 최소 1개, 숫자 문자 최소 1개를 포함한다.
- 여기서 $5 \leq N \leq 20$

1) input 함수

- 함수 원형: `void input(char *p, int N)`
- p는 배열의 시작 주소이며, 문자 n개를 사용자로부터 %c를 이용하여 입력받아 p가 가리키는 배열에 저장한다.
- 반환 값 : 없음

2) upper 함수

- 함수 원형: `void upper(char *p, int N)`
- 입력받은 포인터 변수가 가리키고 있는 곳에 저장된 문자 중에서 대문자만 화면으로 출력한다.
- 반환 값 : 없음

3) lower 함수

- 함수 원형: `void lower(char *p, int N)`
- 입력받은 포인터 변수가 가리키고 있는 곳에 저장된 문자 중에서 소문자만 화면으로 출

력한다.

- 반환 값 : 없음

4) double 함수

- 함수 원형: **int DOUBLE(char *p, int N)**
- 입력받은 포인터 변수가 가리키고 있는 곳에 저장된 숫자 문자를 차례로 연결하여 만든 정수에 2를 곱하여 반환한다. 이 정수는 main 함수에서 화면으로 출력한다.
- 반환 값 : 정수

- ▶ 함수의 원형(인자, 반환값)을 지키지 않거나 목적에 맞게 함수를 사용하지 않거나 위에서 설명된 기능을 정확하게 구현하지 않은 경우 (**input, upper, lower, DOUBLE** 각 20% 감점)
- ▶ 배열 선언 이외에 배열 표기 []를 사용하는 경우 (**main, input, upper, lower, DOUBLE** 각 20% 감점)
- ▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현을 하지 않은 경우 (**main, input, upper, lower, DOUBLE** 각 20% 감점)
- ▶ 전역변수를 사용하는 경우 (**100% 감점**)

입력 예시 1

출력 예시 1

12 FIV12grandE5	↪ N = 12	FIVE grand 250	↪ 125 × 2 = 250
--------------------	----------	----------------------	-----------------

입력 예시 2

출력 예시 2

8 offC5U5T	CUT off 110
---------------	-------------------

[문제 2] 문자열을 입력 받아 문자열에 포함된 모든 공백 문자를 삭제하는 프로그램을 작성하시오.

- del 함수 작성
 - 인자는 char형 포인터
 - 문자열에 포함된 모든 공백 삭제하는 del 함수 작성
 - 반환 값은 없음
- main 함수에서, 사용자로부터 문자열을 입력받고 del 함수 호출하고 문자열을 화면으로 출력

입력 예시 1

출력 예시 1

I am a boy	lamaboy
------------	---------

입력 예시 2

출력 예시 2

C programming language	Cprogramminglanguage
------------------------	----------------------

- ▶ [문제 3-1]에서 [문제 3-3]까지 연관된 문제이다.
- ▶ [문제 3-1] 만 풀면 30점, [문제 3-2] 만 풀어도 60점이다.
- ▶ 문제 푸는 시간을 줄이기 위하여 [문제 3-3] 만 풀어도 100점이고, 세 개의 부분문제 다 풀어도 100점이다.

[문제 3-1] (30점) 문자열을 입력받아, 숫자만 실수형 배열에 저장 후 화면에 출력하시오.

- 1) 사용자로부터 공백이 포함된 문자열을 한 번에 입력받는다.
- 2) 문자열의 전체 길이는 100을 넘지 않도록 하고, 연산에 사용하는 숫자의 개수는 10개 이내로 한다.
- 3) 화면에 출력할 때 "%.0lf" 서식 지정자를 사용한다.

입력 예시 1

출력 예시 1

123 + 2*3 - 4 / 2	123 2 3 4 2
-------------------	-------------

123□+□2*3□-□4□/□2

□123□2□3□4□2

[문제 3-2] (60점) 문자열을 입력받아, 숫자만 실수형 배열에 저장 후 화면에 출력하시오. 이때 숫자 앞에 뿔샘 기호가 있으면 음수 실수로 저장하시오.

- 1) 사용자로부터 공백이 포함된 문자열을 한 번에 입력받는다.
- 2) 문자열의 전체 길이는 100을 넘지 않도록 하고, 연산에 사용하는 숫자의 개수는 10개 이내로 한다.
- 3) 화면에 출력할 때 "%.0lf" 서식 지정자를 사용한다.

입력 예시 1

출력 예시 1

123 + 2*3 - 4 / 2	123 2 3 -4 2
-------------------	--------------

123□+□2*3□-□4□/□2

□123□2□3□-4□2

[문제 3-3] (100점) '+', '-', '*', '/' 기호를 사용한 사칙연산 계산기 프로그램을 작성하시오.

- 1) 사용자로부터 공백이 포함된 문자열을 한 번에 입력받는다. 문자열에는 숫자, 사칙연산 기호, 공백만 포함될 수 있다. (괄호 '(', ')'는 사용하지 않음.)
- 2) 연산에 사용하는 숫자는 음이 아닌 정수이다. (단, 출력은 계산에 따라 실수값이 나올 수 있음. 예를 들면, 3/2=1.500000.)
- 3) 문자열의 전체 길이는 100을 넘지 않도록 하고, 연산에 사용하는 숫자의 개수는 10개 이내로

한다.

- 4) '*'와 '/'는 사칙연산의 규칙에 따라 '+'와 '-'보다 우선적으로 계산되어야 한다. (예를 들면, $2+4*3=14$.)
- 5) 출력에 사용하는 서식 지정자로는 '%lf'를 사용한다.

입력 예시 1

출력 예시 1

1 + 2*3 - 4	3.000000
-------------	----------

1□+□2*3□-□4

입력 예시 2

출력 예시 2

4/2*3 + 10	16.000000
------------	-----------

입력 예시 3

출력 예시 3

2 + 5 + 3 / 2	8.500000
---------------	----------

- ▶ [문제 4-1]에서 [문제 4-2]까지 연관된 문제이다.
- ▶ [문제 4-1] 만 풀면 50점, [문제 4-2] 까지 모두 풀면 100점이다.

[문제 4-1] 3명 학생의 이름과 각 학생이 좋아하는 학생의 이름 3명을 입력받는다. 그다음 찾고자 하는 학생의 이름을 입력하면, 이 학생이 좋아하는 3 학생의 이름을 출력하시오.

- 한 학생의 정보는 다음과 같다. (struct student)
 - 이름 : 공백을 포함하지 않고 길이가 최대 9인 문자열
 - 좋아하는 3 학생의 이름 : 공백을 포함하지 않고 길이가 최대 9인 문자열 3개를 저장하기 위한 문자형 2차원 배열
- typedef 를 사용하시오.
- friend 변수명은 문법에서 이미 사용하고 있는 예약어 이므로 사용하지 마시오.

입력 예시1

출력 예시1

kim -> lee park choi lee -> choi park cha park -> lee choi woo park	□lee□choi□woo
--	---------------

[문제 4-2] 3명 학생의 이름과 각 학생이 좋아하는 학생의 이름 3명을 입력받는다. 서로 좋아하는 관계인 학생들을 (lee가 park을 좋아하고, park도 lee를 좋아하는 관계) 차례로 모두 출력하시오.

- 예시 1 처럼 1쌍이 서로 좋아하는 관계인 경우, 1번이 아닌 2번 출력된다.

- 입력된 학생 순서로, 서로 좋아하는 관계인 학생들을 조사하여, 화면에 출력한다. (예시 2)
- main 함수에서 perA 학생의 목록에 있는 perB 학생이 서로 좋아하는 관계인지 조사하기 위해 서 다음 함수를 만들어서 호출한다.
- ABC(STUDENT stu[], char* perA, char* perB) 함수
 - 1단계) stu[] 배열에서 perB 학생을 찾는다.
 - 2단계) perB 학생의 목록에 perA 학생이 있으면 1을 반환하고, 없으면 0을 반환한다.
 - 반환값: 1 혹은 0 을 가지는 정수형
- friend 변수명은 문법에서 이미 사용하고 있는 예약어 이므로 사용하지 마시오.

입력 예시1

```
kim -> lee park choi
lee -> choi park cha
park -> lee choi gu
```

출력 예시1

```
lee park
park lee
```

입력 예시2

```
kim -> lee park choi
lee -> choi park cha
park -> lee choi kim
```

출력 예시2

```
kim park
lee park
park lee
park kim
```