

〈C프로그래밍 및 실습〉 2차 모의고사 문제지

※ [문제 1-1]에서 [문제 1-2]까지 연관된 문제이며, 배점이 각각 50, 100점이다. 즉, 1-1을 안 풀고 1-2만 풀어도 100점을 부여한다. 가장 높은 배점 하나만 반영한다. 합산하지 않는다.

[문제 1-1] (50점) 양의 정수 N 을 입력 받아, N 의 자리수를 역순으로 바꾼 정수 M 을 구하여 출력하고, 1부터 M 까지의 정수 중 2의 배수의 개수와 3의 배수의 개수를 출력하는 프로그램을 작성하시오.

- 만일 2의 배수나 3의 배수가 없으면 0을 출력한다.

입력 예시 1

출력 예시 1

12 ↪ 정수 N	21 ↪ 역순으로 바꾼 정수 M
	10 7 ↪ M 의 2의 배수 개수와 3의 배수 개수

1부터 자리수를 역순으로 바꾼 정수인 21까지의 2의 배수 개수, 3의 배수 개수 출력

입력 예시 1

출력 예시 1

10 ↪ 정수 N	1 ↪ 역순으로 바꾼 정수 M
	0 0 ↪ M 의 2의 배수 개수와 3의 배수 개수

[문제 1-2] (100점) 종료조건(0 입력)까지 양의 정수 N 을 반복해서 입력 받는다. 입력 받은 정수 N 뒤에, N 의 자리수를 역순으로 바꾼 정수를 붙여서, 대칭이 되는 정수 M 을 만들어 출력하고, 대칭이 되는 정수 M 의 각 자리수에서 3이 나오는 개수를 출력한다. 또한, 대칭되는 정수들 중 최소값과 최대값을 찾아 출력하는 프로그램을 작성하시오.

- 배열 사용 금지 (위반 시 100% 감점)

입력 예시 1

출력 예시 1

13 42 122 0 ↪ 정수	1331 2 4224 0 122221 0
	↪ 대칭정수 대칭정수에 포함된 3의 개수
	1331 122221 ↪ 최소값 최대값

1331에서 3이 두 개이므로 1331과 2를 출력한다.

입력 예시 2

출력 예시 2

1 5000 32 0 ↪ 정수	11 0 50000005 0 3223 2
	↪ 대칭정수 대칭정수에 포함된 3의 개수
	11 50000005

※ [문제 2-1]에서 [문제 2-2]까지 연관된 문제이며, 배점이 각각 50점, 100점이다. [문제 2-1]을 안 풀고 [문제 2-2]만 풀어도 100점을 부여한다. 가장 높은 배점 하나만 반영하며, 합산하지 않는다.

※ 소수는 1과 자신 이외의 수로 나누어떨어지지 않는 **1 보다 큰 양의 정수**이다.

[문제 2-1] (50점) 정수 N을 입력받고, N개의 양의 정수를 입력받아, 입력된 정수 중 **소수만을 한 줄에 3개씩** 출력하는 프로그램을 작성하시오.

- 마지막 줄에 출력되는 소수의 개수는 3개보다 작을 수 있다.
- 각 줄 마지막에 별표(*)를 출력하시오.
- N개의 양의 정수 중, 소수가 최소 1개 이상이라고 가정한다.

입력 예시 1

11 1 2 3 4 5 6 7 8 9 10 11	↦ N=11
-------------------------------	--------

출력 예시 1

2 3 5 *	
7 11 *	↦ 마지막 줄엔 소수 2개만 출력

입력 예시 2

15 11 13 17 19 71 73 74 2 3 5 58 97 89 79 45	↦ N=15
-------------------------------------------------	--------

출력 예시 2

11 13 17 *	
19 71 73 *	
2 3 5 *	
97 89 79 *	

[문제 2-2] (100점) 정수 N을 입력받고, N개의 양의 정수를 입력받아, 입력된 정수 중 소수만을 출력한다. 출력 방식은 **연속적으로 소수가 나오는 경우 같은 줄에 출력**하고, 소수가 아닌 정수가 입력되다가 다시 소수가 나오면 줄을 바꾸어 출력하도록 프로그램을 작성하시오.

- 연속적으로 나오는 소수는 개수에 관계없이 모두 한 줄에 출력한다.
- 각 줄 마지막에 별표(*)를 출력하시오.
- N개의 양의 정수 중, 소수가 최소 1개 이상이라고 가정한다.

입력 예시 1

15 11 13 17 19 71 73 74 2 3 5 58 97 89 79 45	↦ N=15
-------------------------------------------------	--------

출력 예시 1

11 13 17 19 71 73 *	
2 3 5 *	
97 89 79 *	

입력 예시 2

12 71 73 74 2 4 6 58 97 89 79 11 37	↦ N=12
----------------------------------------	--------

출력 예시 2

71 73 *	
2 *	
97 89 79 11 37 *	

※ [문제 3-1]에서 [문제 3-3]까지 연관된 문제이며, 배점이 각각 30, 60, 100점이다. 즉, 3-1, 3-2를 안 풀고 3-3만 풀어도 100점을 부여한다. 가장 높은 배점 하나만 반영한다. 합산하지 않는다.

[문제 3-1] (30점) 양의 정수 N과 M을 입력받고, N개의 **양의 정수**를 입력받아 이 중 **M번째 소수까지**만 출력하는 프로그램을 작성하시오.

- 만일 M개의 소수보다 적으면 있는 소수를 모두 출력하고, 소수가 하나도 없으면 0을 출력한다.

입력 예시 1	출력 예시 1
10 3 1 2 3 4 5 6 7 8 9 10	→ 정수 10개를 읽어 소수 3개 출력 □ 2 3 5

입력 예시 2	출력 예시 2
10 5 1 2 3 4 5 6 7 8 9 10	□ 2 3 5 7

입력 예시 3	출력 예시 3
10 5 32 33 34 35 36 42 44 45 46 54	0

[문제 3-2] (60점) 양의 정수 N과 N개의 **양의 정수**를 입력받아 연속적으로 소수가 나온 최대 회수를 카운트하고 그 구간의 마지막 소수를 출력 하는 프로그램을 작성하시오.

- 단 **100 이상인 수는 제외** 한다 (소수로 생각하지 않음).
- 소수가 6번 연속 나온 구간이 제일 길다면 6을 출력한다. (입력 예시 1)
- 입력 중간에 100이상의 소수가 나와도 100이상의 수이므로 소수 구간은 새로 시작해야한다. (입력 예시 2)
- 소수연속구간 최대 회수를 출력하고, 다음 줄에 그 구간의 소수들 중 가장 마지막 소수를 출력한다.
- 한 번도 소수가 없었다면 0을 출력한다. (입력 예시 3)
- 최대 길이가 같은 구간이 있으면 제일 처음 나온 구간의 정보를 출력한다.

입력 예시 1	출력 예시 1
15 11 13 17 19 71 73 74 2 3 5 58 97 89 79 47	6 → 11 13 17 19 71 73 73 → 그 구간의 마지막 소수

입력 예시 2	출력 예시 2
17 11 13 17 19 101 71 73 74 2 3 5 58 97 89 79 11 37	5 → 97 89 79 11 37 37 → 그 구간의 마지막 소수

입력 예시 3	출력 예시 3
10 32 33 34 35 36 42 44 45 46 54	0

[문제 3-3] (100점) 양의 정수 N과 N개의 정수(음수도 가능) 를 입력받아 연속적으로 소수가 나온 최대 회수를 카운트하고 그 구간의 마지막 소수를 출력 하는 프로그램을 작성하여라. 또한 마지막 줄에는 전체 입력값들 중 최대값과 최소값을 출력한다.

- 단 100 이상인 수는 제외 한다 (소수로 생각하지 않음). 음수는 소수가 아니다.
 - 소수가 6번 연속 나온 구간이 제일 길다면 6을 출력한다. (입력예시 1)
 - 만일 한 번도 소수가 없었다면 0 만을 출력한다. 최대값 최소값을 출력하지 않는다. (문제 4-2의 입력예시 3)
 - 최대 길이가 같은 구간이 있으면 제일 처음 나온 구간의 정보를 출력한다. (입력예시 3)
- N은 1 이상이다. ($N \geq 1$)

입력 예시 1

15 11 13 17 19 71 73 74 2 3 5 -58 97 89 79 -47	출력 예시 1 6 ↦ 11 13 17 19 71 73 73 ↦ 그 구간의 마지막 소수 97 -58 ↦ 최대 최소값
---------------------------------------------------	--------------------------------------------------------------------------

입력 예시 2

19 -100 11 13 17 19 101 71 73 74 -99 3 5 58 97 89 79 11 37 351	출력 예시 2 5 ↦ 97 89 79 11 37 37 ↦ 구간 마지막 소수 351 -100 ↦ 최대 최소값
-------------------------------------------------------------------	----------------------------------------------------------------------

입력 예시 3

11 13 17 99 2 3 5 100 23 29 31 10	출력 예시 3 3 ↦ 2 3 5 (23 29 31도 있지만 처음 것) 5 ↦ 그 구간의 마지막 소수 100 2 ↦ 최대 최소값
--------------------------------------	---------------------------------------------------------------------------------

※ [문제 4-1]에서 [문제 4-2]까지 연관된 문제이며, 배점이 각각 50점, 100점이다. [문제 4-1]을 안 풀고 [문제 4-2]만 풀어도 100점을 부여한다. 가장 높은 배점 하나만 반영하며, 합산하지 않는다.

[문제 4-1] (50점) 영문 소문자 C1과 C2를 입력받아, C1과 C2를 포함한 C1과 C2 사이의 소문자 세 개로 이루어진 단어 중 다음의 조건을 만족하는 단어를 출력하는 프로그램을 작성하시오.

※ 단어 생성 조건

- (1) 세 문자가 모두 같은 문자인 경우는 없다. 즉, 세 문자는 모두 다른 문자이거나, 세 문자 중 두 문자만 같은 문자일 수 있다.
- (2) 두 번째 문자는 모음('a', 'e', 'i', 'o', 'u' 중 하나)이다.
- (3) 두 번째 문자는 세 문자의 중간값이다. 즉, 세 문자를 알파벳 순서로 줄을 세운 경우, 두 번째 문자가 중간 순서에 위치한다. 단, aab 또는 baa처럼 같은 문자가 중복된 경우에도 알파벳 순서로 줄을 세우면 a-a-b순으로 문자 a가 중간값이 된다.

- 출력 순서는 단어가 사전에 나오는 순서이다. 사전에 나오는 순서란 알파벳 순서가 빠른 문자로 이루어진 단어가 먼저 출력된다는 의미이다. 예를 들어, aab와 abc를 비교해보면, 첫 번째 문자는 'a'로 동일하지만, 두 번째 문자인 aab의 'a'가 abc의 'b'보다 빠르기 때문에 aab가 abc보다 사전 순서상 빠른 단어이다.

- C1과 C2 사이의 문자로 조건에 맞는 단어가 만들어지지 않는 경우에는 "none"을 출력한다.
- C1은 C2 보다 알파벳 순서 상 빠른 문자라고 가정한다.

입력 예시 1

출력 예시 1

dg	<div> <div>dee</div> <div>def</div> <div>deg</div> <div>eed</div> <div>eef</div> <div>eeg</div> <div>fed</div> <div>fee</div> <div>ged</div> <div>gee</div> </div> <div> ↳ 'd', 'e', 'f', 'g'로 만들 수 있는 64개의 단어 중 조건을 만족하는 단어만 출력 </div>
----	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

입력 예시 2

출력 예시 2

jn	<div>none</div> <div> ↳ 'j'에서 'n' 사이에는 모음이 존재하지 않아 조건에 맞는 단어가 만들어지지 않아 "none" 출력 </div>
----	---------------------------------------------------------------------------------------------------------------

[문제 4-2] (100점) 영문 소문자 C1, C2와 영문 소문자 세 개로 이루어진 단어 T를 C1C2-T의 형식으로 입력받아,

- (1) C1과 C2를 포함한 C1과 C2 사이의 소문자 세 개로 이루어진 단어 중 [문제 4-1]의 단어 생성 조건을 만족하는 단어를 사전에 나오는 순서로 출력한다.
- (2) (1)에서 출력된 단어 중에서, 단어 T보다 사전순서상 빠른 가장 마지막으로 출력된 단어를 출력한다.

- (1)의 출력 순서는 단어가 사전에 나오는 순서이다. 사전에 나오는 순서란 알파벳 순서가 빠른 문자로 이루어진 단어가 먼저 출력된다는 의미이다. 예를 들어, aab와 abc를 비교해보면, 첫 번째 문자는 'a'로 동일하지만, 두 번째 문자인 aab의 'a'가 abc의 'b'보다 빠르기 때문에 aab가 abc보다 사전 순서 상 빠른 단어이다.
- C1과 C2 사이의 문자로 조건에 맞는 단어가 만들어지지 않는 경우에는 "none"을 출력한다.
- 사전순서상 T보다 빠른 단어를 찾을 수 없는 경우에도 "none"을 출력한다. (출력 예시 2와 3 참고)
- C1은 C2 보다 알파벳 순서 상 빠른 문자라고 가정한다.

입력 예시 1

출력 예시 1

<p>dg-fff</p> <p>↳ C1C2-T 형태 입력</p> <p>↳ C1은 'd', C2는 'g'</p> <p>↳ T는 "fff"</p>	<p>dee</p> <p>def</p> <p>deg</p> <p>eed</p> <p>eef</p> <p>eeg</p> <p>fed</p> <p>fee</p> <p>ged</p> <p>gee</p> <p>fee</p>	<p>↳ 'd', 'e', 'f', 'g'로 만들 수 있는 64개의 단어 중 조건을 만족하는 단어만 출력</p> <p>↳ 출력된 단어 중 "fff"보다 빠른 가장 마지막 단어 출력</p>
---------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------

입력 예시 2

출력 예시 2

<p>np-nno</p> <p>↳ C1C2-T 형태 입력</p> <p>↳ C1은 'n', C2는 'p'</p> <p>↳ T는 "nno"</p>	<p>noo</p> <p>nop</p> <p>oon</p> <p>oop</p> <p>pon</p> <p>poo</p> <p>none</p>	<p>↳ 'n', 'o', 'p'로 만들 수 있는 27개의 단어 중 조건을 만족하는 단어만 출력</p> <p>↳ 출력된 단어 중 "nno"보다 빠른 단어가 없어서 "none" 출력</p>
---------------------------------------------------------------------------------	-------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------

입력 예시 3

출력 예시 3

<p>jn-mmm</p>	<p>none</p> <p>none</p>	<p>↳ 'j'에서 'n' 사이에는 모음이 존재하지 않아 조건에 맞는 단어가 만들어지지 않아 "none" 출력</p> <p>↳ 출력된 단어 중 "mmm"보다 빠른 단어가 없어서 "none" 출력</p>
---------------	-------------------------	------------------------------------------------------------------------------------------------------------------