

<고급 C프로그래밍 및 실습> 과제 #1 (9장 포인터)

※ 문제지의 무단 배포 및 사용을 원칙적으로 금지합니다.

- 특히, 커뮤니티, 개인 블로그 등 인터넷 사이트 게시를 절대 금지합니다.

※ 문제에 대한 안내

- 입출력 예시에서 \mapsto 이 후는 각 입력과 출력에 대한 설명이다.

- OJ에서 작성코드가 어떻게 실행되는지 확인하기 위하여 [Sample Submit](#) 기능사용 가능 합니다.

※ [문제 1-1]에서 [문제 1-2]까지는 연관된 문제이다. [문제 1-1]을 안 풀고 [문제 1-2]만 풀어도 100점을 부여한다. 가장 높은 점수 하나만 반영하며, 합산하지 않는다. [문제 1-1]을 풀고 [문제 1-2]를 풀면 쉽게 문제를 풀 수 있다.

[문제 1-1][레벨 1] (50점) 입력될 줄 수에 해당하는 정수 N을 입력받는다. 각 줄에서 0 이 입력 될 때까지 정수를 입력받는다. 각 줄에서 0 이후에는 숫자가 없다. 각 줄에서 0은 두 번째부터 입력될 수 있다. 각 줄에서 0을 제외한 최대값과 최소값을 출력하시오.

1) MAX() 함수 이용

- 인자: int형 배열 ar
- 배열 표기 [] 사용금지
- 배열에서 최대값의 주소를 반환한다.
- 반환 값: int형 포인터 즉 주소

2) MIN() 함수 이용

- 인자: int형 배열 ar
- 배열 표기 [] 사용금지
- 배열에서 최소값의 주소를 반환한다.
- 반환 값: **int형 포인터 즉 주소**

3) main 함수에서는

- 사용자로부터 정수 N을 입력 받는다. 아래 과정을 N번 반복한다.
 - 사용자로부터 정수를 입력 받는다. 0이 되면 종료한다. 배열 ar의 크기는 100.
 - MAX 함수를 호출한다. MIN 함수를 호출한다.
 - 두 함수에서 전달받은 정수를 화면에 출력한다.
 - MAX 함수에서 전달받은 주소가 가리키는, 배열의 원소를 화면에 출력한다.
 - MIN 함수에서 전달받은 주소가 가리키는, 배열의 원소를 화면에 출력한다.

4) 함수 원형 선언을 하고, 함수들을 main 함수 아래에 작성하시오.

▶ 배열 선언 이후, 배열 표기 [] 는 사용 금지 (포인터 표기 사용) (위반 시 50% 감점)

▶ 사용자로부터 정수들 입력하여 배열에 저장 시에 포인터 사용 (위반 시 20% 감점)

▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (위반 시 50% 감점)

(예: `for(p = ar ; p < ar+5 ; p++)`)

여기서 p는 정수형 포인터, ar은 배열의 이름)

- ▶ 함수 원형 선언을 하고, 함수들을 main 함수 아래에 작성하시오 (위반 시 함수 별 20% 감점)
- ▶ 전역변수 사용금지 (위반 시 100% 감점)

입력 예시 1

출력 예시 1

3 3 7 6 -2 0 1 7 6 13 2 0 2 0	↦ N	7 -2 ↦ 최대값 최소값 13 1 2 2
--	-----	-------------------------------

입력 예시 2

출력 예시 2

2 13 7 6 6 2 0 7 3 4 5 6 10 4 0		13 2 10 3
---------------------------------------	--	--------------

[문제 1-2][레벨 2] (100점) 입력될 줄 수에 해당하는 정수 N을 입력받는다. 각 줄에서 0 이 입력될 때까지만 정수를 입력받는다. 각 줄에서 0 이후에는 숫자가 없다. 각 줄에서 0은 두 번째부터 입력될 수 있다. 각 줄에서 0을 제외한 최대값과 최소값 사이에 위치한 정수를 사용자로부터 입력받은 순서대로 출력하시오. (입력받은 순서상 최대값과 최소값 사이에 위치한 정수를 출력)

- 크기가 아니고 입력된 위치가 두 수 사이에 있는 정수를 화면에 출력한다.
- 각 줄에서 최대값과 최소값은 1개씩만 있다고 가정한다.

1) [문제 1-1]에서 작성한 MAX() 함수와 MIN() 함수를 이용한다.

2) main 함수에서는

- 사용자로부터 정수 N을 입력 받는다. 아래 과정을 N번 반복한다.
 - 사용자로부터 정수를 입력 받는다. 0이 되면 종료한다. 배열 ar의 크기는 100.
 - MAX 함수를 호출한다. MIN 함수를 호출한다.
 - 최대값과 최소값 사이에 위치한 정수를 사용자로부터 입력받은 순서대로 출력한다.
 - 각 줄에서 최대값과 최소값은 한 개씩만 있다.
 - 화면에 출력할 정수가 없으면 none을 출력한다.

3) 함수 원형 선언을 하고, 함수들을 main 함수 아래에 작성하시오.

▶ 배열 선언 이후, 배열 표기 [] 는 사용 금지 (포인터 표기 사용) (위반 시 50% 감점)

▶ 사용자로부터 정수들 입력하여 배열에 저장 시에 포인터 사용 (위반 시 20% 감점)

▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (위반 시 50% 감점)

(예: for(p = ar ; p < ar+5 ; p++)

여기서 p는 정수형 포인터, ar은 배열의 이름)

▶ 함수 원형 선언을 하고, 함수들을 main 함수 아래에 작성하시오 (위반 시 함수 별 20% 감점)

▶ 전역변수 사용금지 (위반 시 100% 감점)

입력 예시 1

3	↦ N
3 7 6 5 5 -2 0	
1 7 6 2 3 13 2 0	
2 0	

출력 예시 1

6 5 5
7 6 2 3
none

입력 예시 2

2
13 7 6 6 2 0
7 3 4 5 6 7 10 4 0

출력 예시 2

7 6 6
4 5 6 7

※ [문제 2-1]에서 [문제 2-2]까지는 연관된 문제이다. [문제 2-1]을 안 풀고 [문제 2-2]만 풀어도 100점을 부여한다. 가장 높은 점수 하나만 반영하며, 합산하지 않는다.

[문제 2-1][레벨 1] (50점) 21인승 프리미엄 고속버스의 좌석 중복 여부를 확인하는 프로그램을 작성하려고 한다. 버스 좌석 번호는 1~21까지 차례로 부여한다. 승객의 수는 21명을 넘지 않으며, 최소 5명이상 예약을 해야 버스가 운행을 한다. 승객들은 여객터미널에서 좌석 예약을 위해 한명씩 1~21사이의 숫자를 임의로 입력하여 좌석을 예약하였다. 이 때, 좌석 예약에 중복이 있는 경우, 중복 예약된 좌석 번호와 예약 중복 횟수를 알려주는 프로그램을 작성하시오.

(우선 승객의 수인 N을 입력하며, N은 5~21사이의 숫자를 입력한다. 그 다음 줄에 N개의 1~21사이의 숫자를 입력하며, 이는 승객들이 예약한 좌석 번호를 의미한다. 입력하는 숫자의 순서가 승객들이 예약하는 순서와 동일하며, 중복 예약이 없는 경우에는 0을 출력한다. 승객의 수(N)가 5보다 적은 경우, 승객 수 부족의 의미로 -1을 출력한다.)

▶ 배열 선언 이후, 배열 표기 [] 는 사용 금지 (포인터 표기 사용) (50% 감점)

▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (50% 감점)

(예: for(p = ar ; p < ar+5 ; p++) 여기서 p는 정수형 포인터, ar은 배열의 이름)

입력 예시 1

8	7 2 ↦ 7이 2번 입력됨
7 2 5 5 20 5 8 7	5 3 ↦ 5가 3번 입력됨
	↦ 7이 먼저 입력되었으므로 5보다 먼저 출력됨

출력 예시 1

입력 예시 2

7	0 ↦ 중복된 숫자가 없는 경우 0을 출력함
21 15 20 9 16 1 3	

출력 예시 2

입력 예시 3

3	-1 ↦ 승객수가 5보다 작은 경우,
10 5 5	좌석 예약 중복유무와 상관없이 -1을 출력함

출력 예시 3

[문제 2-2][레벨 2] (100점) [문제 2-1]과 같이 승객들이 좌석 예약을 완료한 후, 승객들의 수가 충분한지 확인하고, 중복된 예약 좌석 번호의 유무를 확인하여 중복된 경우 빈 좌석으로 변경해주는 프로그램을 작성하려고 한다.

- 승객 수가 5명 미만인 경우, 승객부족의 의미로 -1 만을 출력한다. (중복 예약 여부와는 무관)
- 승객 수가 5명 이상인 경우에는 승객이 충분하여 버스가 운행한다는 의미로 0을 출력한다. 그리고 다음 줄에 승객들이 예약한 좌석번호를 순서대로 출력하여 보여준다. 좌석 번호가 중복된 경우, 나중에 예약한 승객의 중복된 좌석 번호를 빠른 번호의 빈 좌석부터 차례로 채우는 방식으로 변경시켜준다.

1) main 함수

- 승객의 숫자를 입력 받는다.
- 승객들이 차례로 버스 좌석 번호를 예약한다. (중복 가능)
- passengerN 함수를 호출하여 승객 수가 부족한지 확인한다.
- 승객 수가 부족한지 여부를 출력한다. (부족: -1, 충분: 0)
- 승객 수가 부족하면 이후 프로그램을 실행시키지 않는다.
- 승객 수가 충분하면 최종 좌석 예약 결과를 출력한다. (중복된 좌석 예약이 없으면 그대로 출력, 중복된 좌석 예약이 있으면 수정된 좌석 예약 결과를 출력)

2) passengerN(int n) 함수

- n: 승객 수
- 반환 값: 승객 수가 부족하면 -1, 승객 수가 충분하면 0을 반환한다.

3) rebooking(int *ar)함수

- ar: 처음 예약된 버스 좌석 번호가 저장되어 있는 배열의 이름
- 중복된 좌석 번호가 존재하는지 확인하고, 중복된 좌석이 존재하는 경우, changeseat 함수를 호출하여 중복된 좌석 번호를 빠른 번호의 빈 좌석 번호로 대체한다.

단, 중복된 좌석들 중 나중에 예약한 좌석 번호를 빈 좌석으로 대체한다.

예) 8 3 3 3 1 4 가 입력된 경우, 3 이 중복되므로 이들 중 두 번째의 3 이 먼저 2로 변경 된다(1번 좌석은 이미 예약되었기 때문에 그 다음 번호로 예약). 그 다음으로 세 번째 3 이 5로 변경 된다(1번부터 4번까지의 좌석이 이미 예약되어 5번으로 변경). 최종 결과는 8 3 2 5 1 4 가 됨.

즉, 아래와 같은 순서로 변경 됨. 8 3 3 3 1 4 --> 8 3 2 3 1 4 --> 8 3 2 5 1 4

4) changeseat(int *ar) 함수

- ar: 처음 예약된 버스 좌석 번호가 저장되어 있는 배열의 이름
- 1~21 사이의 숫자들 중 배열에 존재하지 않으면서 가장 작은 숫자를 찾아준다.
- 반환 값 : 찾아낸 가장 작은 숫자를 반환한다.

- ▶ 배열 선언 이후, 배열 표기 [] 는 사용 금지 (포인터 표기 사용) (50% 감점)
- ▶ 배열에 저장 된 정수들 화면에 출력 시에 포인터 사용 (20% 감점)
- ▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (50% 감점)
(예: for(p = ar ; p < ar+5 ; p++) 여기서 p는 정수형 포인터, ar은 배열의 이름)
- ▶ 전역변수 사용금지 (100% 감점)

입력 예시 1

출력 예시 1

7 1 2 3 3 4 5 6	0 ↳ 승객 수가 충분함으로 0을 출력함 1 2 3 7 4 5 6 ↳ 중복되는 3번 좌석 중 나중에 등장하는 4번 승객의 좌석을 예약되지 않은 가장 빠른 좌석 번호인 7로 변경함
--------------------	--

입력 예시 2

출력 예시 2

6 8 8 6 8 6 1	0 ↳ 승객 수가 충분함으로 0을 출력함 8 2 6 3 4 1 ↳ 중복되는 좌석 번호인 6과 8번들 중에서 먼저 예약되었던 8 번 중복 좌석들부터 빈 좌석으로 변경(2, 3으로 변경) 후, 중복된 6번 좌석을 4로 변경
------------------	--

입력 예시 3

출력 예시 3

3 1 2 2	-1 ↳ 승객 수가 부족함으로 -1을 출력함. 중복된 좌석 예약이 있 지만 승객이 부족함으로 이후 아무 것도 출력하지 않음.
------------	---

입력 예시 4

출력 예시 4

10 8 3 8 3 8 11 21 13 5 2	0 8 3 1 6 4 11 21 13 5 2
------------------------------	-----------------------------

※ [문제 3-1]에서 [문제 3-2]까지는 연관된 문제이다. [문제 3-1]을 안 풀고 [문제 3-2]만 풀어도 100점을 부여한다. 가장 높은 점수 하나만 반영하며, 합산하지 않는다.

[문제 3-1][레벨 2] (50점) '#' 문자가 입력될 때까지 문자를 입력받는다. 소문자로 구성된 단어들을 출력하시오. 각 단어는 소문자가 아닌 문자로 구분된다. 마지막에 길이가 가장 긴 단어의 길이를 출력하시오.

- '#' 문자를 포함하여 최대 100개 문자가 입력된다.
- 첫 글자는 항상 영어 소문자이다.

- ▶ 배열 선언 이후, 배열 표기 [] 는 사용 금지 (포인터 표기 사용) (50% 감점)
- ▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (50% 감점)

입력 예시 1

banana25apple#

출력 예시 1

banana
apple
6 ↳ 가장 긴 단어의 길이

입력 예시 2

roll out electric334car334#

출력 예시 2

roll
out
electric
car
8 ↳ 가장 긴 단어의 길이

[문제 3-2][레벨 2] (100점) [문제 3-1]에 추가하여 길이가 가장 긴 단어의 첫 번째 문자를 화면에 출력한다. 함수 PRT를 작성하여, 가장 긴 단어를 화면에 출력한다.

1) PRT(int n, char *p) 함수

- n: 가장 긴 단어의 길이
- p: 단어의 시작 주소
- 반환 값: 없음

2) main 함수 위에, PRT 함수 원형 선언을 하고, main 함수 다음에, PRT 함수를 작성한다.

- ▶ 배열 선언 이후, 배열 표기 [] 는 사용 금지 (포인터 표기 사용) (50% 감점)
- ▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (50% 감점)
- ▶ PRT 함수 미작성 시 (50% 감점)
- ▶ 함수원형선언 위반 시 (30% 감점)

입력 예시 1

banana25apple#

출력 예시 1

banana
apple
6 b ↳ 가장 긴 단어의 길이와, 그 단어의 첫 번째 문자
banana ↳ 가장 긴 단어

입력 예시 2

roll out electric334car334#

출력 예시 2

roll
out
electric
car
8 e ↳ 가장 긴 단어의 길이와, 그 단어의 첫 번째 문자
electric ↳ 가장 긴 단어

※ [문제 4-1]에서 [문제 4-2]까지는 연관된 문제이다. [문제 4-1]을 안 풀고 [문제 4-2]만 풀어도 100점을 부여한다. 가장 높은 점수 하나만 반영하며, 합산하지 않는다.

[문제 4-1][레벨 1] (50점) 세종대학교 101호와 102호에서 코딩 시험을 보기로 하였다. 코딩 시험은 100점 만점으로 치러진다. 각 강의실에는 최소 1명에서 최대 10명의 응시생이 시험을 볼 수 있지만 몇 명이 시험을 치를 지는 정해진바 없다. 101호와 102호 각 방에서 본 시험 점수 n 개와 m 개를 2개의 배열 a 와 b 에 각각 입력받는다. a 와 b 배열의 시험점수를 입력받을 때 -1이 들어오기 전까지만 입력받는다.

이 때 101호에서 응시한 학생 수와 102호에서 응시한 학생 수를 출력하시오. 그리고 그 다음 줄에 101호와 102호 응시 학생들 중 최고 점수와 최저 점수를 출력하시오. 단, 배열 a 와 b 이외의 다른 배열은 사용할 수 없다.

입력 예시 1

출력 예시 1

10 50 70 -1	3 2
20 100 -1	100 10

입력 예시 2

출력 예시 2

5 20 23 80 80 -1	5 10
70 50 50 45 5 15 28 46 76 11 -1	80 5

[문제 4-2][레벨 2] (100점) 세종대학교 101호와 102호에서 코딩 시험을 보기로 하였다. 코딩 시험은 100점 만점으로 치러진다. 각 강의실에는 최소 1명에서 최대 10명의 응시생이 시험을 볼 수 있지만 몇 명이 시험을 치를 지는 정해진바 없다. 101호와 102호 각 방에서 본 시험 점수 n 개와 m 개를 2개의 배열 a 와 b 에 각각 입력받는다. a 와 b 배열의 시험점수를 입력 받을 때 -1이 들어오기 전까지만 입력받는다.

이 때 101호와 102호에서 응시한 학생들의 점수 중 가장 높은 n 개의 점수를 내림차순으로 정렬하여 출력한다. 그 다음 줄에는 101호와 102호에서 응시한 학생들의 점수 중 가장 낮은 m 개의 점수를 오름차순으로 정렬하여 출력한다. 단, 배열 a 와 b 이외의 다른 배열은 사용할 수 없다.

▶ 배열 선언 이후, 배열 표기 [] 는 사용 금지 (포인터 표기 사용) (50% 감점)

▶ 반복문으로 배열 훑어볼 시, 주소를 이용하여 반복문 구현 (50% 감점)

[예: for($p = ar$; $p < ar+5$; $p++$) : p 는 정수형 포인터, ar 은 정수형 배열의 이름]

입력 예시 1

출력 예시 1

10 50 70 -1	100 70 50
20 100 -1	10 20

입력 예시 2

출력 예시 2

10 20 40 5 10 -1	100 100 90 80 75
100 100 50 20 5 90 80 75 75 60 -1	5 5 10 10 20 20 40 50 60 75

제출기한 및 방법

- * OJ시스템(<https://ex-oj.sejong.ac.kr/index.php/auth/login>) 내의 과제1을 이용하여 제출
- * 제출 마감: **9월 X일(일요일)** 밤 12시 까지 제출
- * 과제 점수는 위 마감일 전에 제출된 가장 마지막 코드를 기준으로 부여
(마감일 이후에 제출되는 코드는 채점에서 제외됨)
- * OJ시스템의 점수는 참고로만 사용
 - 문제의 조건을 만족시켜서 코딩했는지 조사 후 최종 점수 부여
 - 코드 2줄당 1줄의 비율로 반드시 주석을 적을 것(주석이 없을 경우 감점)
 - 보고서 등 기타 제출물 없음