

〈고급 C프로그래밍 및 실습〉 13장 고급 문법 실습 문제

※ 문제에 대한 안내

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

2절 [문제 1] 두 개의 양의 정수를 입력받고, 이 두 정수의 최대공약수를 출력하는 프로그램을 작성하시오. 반드시 **재귀함수**를 사용하시오.

입력 예시 1

출력 예시 1

366 60	6
--------	---

Hint: 유클리드 호제법으로 최대공약수 계산하기

(예) 입력 (12, 8) 1단계 (8, 12%8=4) 2단계 (4, 8%4=0) 최대공약수는 4

(작은 수 8, 큰 수에서 작은 수 나눈 나머지 12%8)

2절 [문제 2] 오일러의 수 e는 자연 로그의 밑수로 사용된다. 이 값은 다음과 같은 식에 의하여 근사치를 구할 수 있다. 오일러 수를 계산하는 프로그램을 작성하시오.

- 소수점 이하 **여섯째 자리**까지 화면에 출력 하시오.

$$= 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} + \dots + \frac{1}{n!}$$

입력 예시 1

출력 예시 1

30 \mapsto 계산할 수	2.718282
--------------------	----------

Hint: factorial() 함수 호출하여 사용

factorial() 함수는 **재귀함수**로 구현

2절 [문제 3] **재귀함수**를 이용하여 사용자가 입력한 N 번째 피보나치 수열의 값을 구하는 프로그램을 작성하시오. 여기서, 피보나치 수열은 다음과 같은 점화식으로 표현된다.

$$F_n = \begin{cases} 0 & \text{if } n = 0; \\ 1 & \text{if } n = 1; \\ F_{n-1} + F_{n-2} & \text{if } n > 1. \end{cases}$$

즉, 피보나치 수 F_n 은 0과 1로 시작되며, 다음 피보나치 수는 바로 앞의 두 피보나치 수의 합이 된다. (즉 $n=0$ 와 $n=1$ 일 때는 예외적이고 $n=2$ 부터 정상적으로 앞 두 수의 합이다)

$n = 0, 1, 2, 3, 4, 5, 6, 7, \dots$ 에 해당하는 피보나치 수는 $F_n = 0, 1, 1, 2, 3, 5, 8, 13, \dots$ 이 된다.

n	0	1	2	3	4	5	6	7	...
피보나치수	0	1	1	2	3	5	8	13	...

입력 예시 1

출력 예시 1

6 ↪ n	8
-------------------	---

응용문제

[문제 1] N 줄의 공백이 포함된 문자열을 입력받아, 각 문자열에 포함된 단어 수 순서대로 문자열을 정렬하는 프로그램을 동적 할당을 이용하여 다음의 조건들을 지켜 작성하시오.

(1) 구조체 struct string을 다음과 같이 선언하여 사용한다.

```
struct string{
    char *str;      // 문자열을 가리키는 포인터, 문자 배열 (char str[])로 선언한 경우 50% 감점
    int words;      // str이 가리키는 문자열에 포함된 단어 수
};
```

- (2) 정수 N을 입력받고, 크기 N의 struct string 타입의 구조체 배열을 동적으로 할당 받는다.
- (3) N 줄의 문자열을 입력받아, 문자열 크기에 맞는 문자 배열을 동적으로 할당 받아 str에 연결한다. 입력되는 문자열의 최대 크기는 100이다.
- (4) 문자열에 포함된 단어의 수를 세어 words에 저장한다. 공백과 공백 사이가 하나의 단어이다.
- (5) 문자열에 포함된 단어 수가 가장 많은 문자열부터 가장 적은 문자열까지 내림차순으로 정렬하고, 문자열과 단어 수를 출력한다. 단어 수가 같은 경우에는 입력순으로 출력한다.

입력 예시 1

출력 예시 1

5 why so serious icecream makes everything better you make me smile c language midterm and final see you	c language midterm and final 5 icecream makes everything better 4 you make me smile 4 why so serious 3 see you 2
---	--

[문제 2] N 줄의 공백이 포함된 문자열을 입력받아, 각 문자열에 포함된 단어 수 순서대로 문자열을 정렬하고, 각 문자열에 가장 많이 나타나는 단어를 출력하는 프로그램을 동적 할당을 이용하여 다음의 조건들을 지켜 작성하시오.

(1) 구조체 struct string을 다음과 같이 선언하여 사용한다.

```
struct string{
    char *str;      // 문자열을 가리키는 포인터, 문자 배열 (char str[])로 선언한 경우 50% 감점
    int words;      // str이 가리키는 문자열에 포함된 단어 수
};
```

- (2) 정수 N을 입력받고, 크기 N의 struct string 타입의 구조체 배열을 동적으로 할당 받는다.
- (3) N 줄의 문자열을 입력받아, 문자열 크기에 맞는 문자 배열을 동적으로 할당 받아 str에 연결한다. 입력되는 문자열의 최대 크기는 100이다.
- (4) 문자열에 포함된 단어의 수를 세어 words에 저장한다. 공백과 공백 사이가 하나의 단어이다.
- (5) 문자열에 포함된 단어 수가 가장 많은 문자열부터 가장 적은 문자열까지 내림차순으로 정렬한다. 단어 수가 같은 경우에는 입력순으로 정렬한다.
- (6) 각 문자열을 단어로 나누어 동적으로 할당받은 2차원 배열에 저장한다. 먼저, 단어 수와 같은 크기의 문자 포인터 배열을 동적으로 할당받고, 문자열을 단어로 나누어 각 단어가 저장될 문자 배열을 동적으로 할당받아 포인터 배열에 연결한다.
- (7) 각 문자열 마다 저장된 단어 중 중복 횟수가 가장 많은 단어를 찾아, 해당 문자열, 문자열의 단어 수, 중복 횟수가 가장 많은 단어, 해당 단어의 중복 횟수를 출력한다. 중복 횟수가 가장 많은 단어가 둘 이상인 경우, 사전 순서상 빠른 단어를 출력한다.
- (8) 각 문자열에 대한 출력을 마친 후, 단어 저장을 위해 동적으로 할당받은 2차원 배열은 해제 (free)한다.

입력 예시 1

```
5
why why why so so so serious
icecream makes everything better and better
you make me smile
c language exam and exam and exam and exam
see you later later later later
```

출력 예시 1

```
c language exam and exam and exam and exam 9 exam 4
why why why so so so serious 7 so 3
icecream makes everything better and better 6 better 2
see you later later later later 6 later 4
you make me smile 4 make 1
↳ 문자열, 단어수, 빈도수 높은 단어, 빈도수 출력
↳ 빈도수 높은 단어가 둘 이상인 경우, 사전 순서상 빠른 단어 출력
```

[문제 3] 공연 예약 정보를 출력하는 프로그램을 함수와 동적 할당을 이용하여 다음의 조건들을 지켜 작성하시오.

(1) 구조체 struct reserve를 다음과 같이 선언하여 사용한다.

```
struct reserve{
    int id;           // 고객 id
    char grade[4];    // 구매한 티켓의 등급
    int price;        // 구매한 티켓의 가격
};
```

(2) 정수 N을 입력받고, 고객의 예매 정보를 저장할 크기 N의 struct reserve 타입의 배열을 동적으로 할당 받는다.

(3) N개의 예매 정보(고객 id와 구매한 티켓의 등급)를 입력받아, 티켓 가격과 함께 구조체 배열에 저장한다. 한 공연에는 4개 등급의 티켓이 있고, 각 티켓의 등급과 가격은 아래 표와 같다.

티켓 등급	가격
VIP	140,000원
R	110,000원
S	85,000원
A	55,000원

(4) 예매 정보 입력을 위해 input 함수를 사용한다.

- 함수 원형: **void input(struct reserve *p, int N)**
- 구조체 배열의 시작 주소와 배열의 크기를 인자로 받아, 입력받은 고객 id, 티켓 등급과 티켓 가격을 저장한다.
- input 함수에서는 배열 표기 []를 사용하지 않고, 포인터를 움직이며 반복문을 구현한다.

(5) 예매 정보를 출력할 고객의 id를 입력받아, 해당 고객이 구매한 티켓의 등급과 티켓 가격을 입력순서대로 출력하고, 티켓 구매에 사용된 총 금액을 출력한다. 티켓 구매에 사용된 총 금액을 계산하기 위해 cal_sum 함수를 사용한다.

- 함수 원형: **int cal_sum(struct reserve *p, int N, int id)**
- 구조체 배열의 시작 주소와 배열의 크기, 그리고 고객 id를 인자로 받아, 해당 고객이 구매한 티켓의 총 구매 금액을 계산하여 반환한다.
- cal_sum 함수에서는 배열 표기 []를 사용하지 않고, 포인터를 움직이며 반복문을 구현한다.

(6) 단, **main 함수에서의 주소 표현 방식은 제한 사항 없다.**

입력 예시 1

출력 예시 1

10	↳ N=10명	A 55000
123 A		A 55000
210 VIP		R 110000
123 A		A 55000
123 R		R 110000
210 S		385000
210 VIP		
123 A		↳ 123번 고객이 구매한 티켓의 종류와 티켓 한 장의
523 A		↳ 가격을 입력순서대로 출력하고, 티켓 구매 금액의
123 R		↳ 총 합 출력
523 VIP		
123	↳ id=123, 123번 고객의 정보 출력	