

추가 힌트

<동적할당>

1. 정수 N개를 입력받아 동적 배열에 저장하고, 배열의 순서를 거꾸로 뒤집어 출력하는 프로그램을 작성하시오.

입력: N개의 정수

출력: 역순 배열 출력

2. 사용자로부터 정수 N을 입력받아 동적배열에 저장한 뒤, 양수, 음수, 0의 개수를 각각 출력하는 프로그램을 작성하시오.

3. 사용자로부터 정수 N을 입력받아 동적배열에 저장한 뒤, 전체 평균을 구하고 평균이상인 값들만 출력하는 프로그램을 작성하시오.

<구조체 배열, 구조체 포인터 배열>

4. 영화 정보를 저장하는 구조체를 정의하여 보자. 구조체에는 영화(Movie)의 제목, 평점, 관객수 등이 저장되어야 한다. 3개 정도의 영화에 대한 정보를 저장할 수 있도록 구조체의 배열을 생성하고 구조체 포인터 배열을 매개변수로 하여 사용자로부터 값을 받아서 초기화하는 함수와, 영화를 관람한 총 관객수를 계산하는 함수를 작성하고 테스트하시오. 주어진 코드를 참고하여 프로그램을 완성하시오. (30점)

- 구조체 정의
 - title : 영화제목
 - rating : 영화 평점
 - viewer : 관객 수
- 함수 구현
 - 사용자로부터 SIZE 개수의 영화 정보를 입력받아 각 영화 정보를 초기화하는 함수 : input_movies - 영화 정보는 구조체 포인터 배열을 통해 관리
 - 모든 영화의 총 관객수를 계산하여 반환하는 함수 : total_viewers
- 메인 함수
 - SIZE 개수의 Movie 구조체와 그 포인터 배열을 선언하고 초기화
 - input_movies와 total_viewers 함수를 호출하여 영화 정보를 입력받고 영화의 총 관객수를 계산한 후, 결과를 출력

<실행결과 예시>

```
제목, 평점, 관객수(만명) 입력: 하얼빈 7.67 275
제목, 평점, 관객수(만명) 입력: 소방관 7.2 331
제목, 평점, 관객수(만명) 입력: 모아나2 7.1 337
총 관객 수: 943만명
```

```
#define SIZE 3 // 영화의 수를 정의하는 상수
```

```
// 구조체 정의
```

```
// 함수 선언
```

```
void input_movies(struct Movie* movies[], int size);
```

```
int total_viewers(struct Movie* movies[], int size);
```

```
int main(void) {
```

```
    struct Movie movies[SIZE]; // 영화정보 배열 선언
```

```
    struct Movie* m_ptrs[SIZE]; // 영화정보 포인터 배열 선언
```

```
// 포인터 배열 초기화

// 사용자로부터 영화정보 입력 받아 초기화하는 함수 호출

// 총 관객수 계산하는 함수 호출

printf("총 관객 수: %d명\n", total);
return 0;
}

// 함수 정의
```