**1문제)**

텍스트, 스크린샷, 정보이(가) 표시된 사진

자동 생성된 설명

**문제분석)**

먼저 유저에게 입력 받는 것이 아니라 주어진 값으로 배열을 생성하기 때문에 생성할 때 따로 for문을 사용하지 않고 그냥 선언하면 될 것 같다. 각 열과 행의 합계를 구할 때는 2중 for문을 이용해 구하면 될 것 같고 행과 열의 합계 sum 변수를 초반에 선언해두고 2중 for문에서 각 행과 열의 인덱스를 증감연산자를 이용해 더하고 한 열과 행을 각각 더하고 프린트하고 0으로 초기화하면 될 것 같다. 아니면 그냥 2중 for문의 사이에 sum 변수를 0으로 선언해도 될 것 같긴 하다. 그리고 문제에서 0행이라고 나오는데 보통 일상생활에서는 1행부터 말하기 때문에 i+행 또는 j+1열로 바꿔 줘야겠다.

**소스코드)**

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

2중 for문을 이용해 각 행과 열의 인덱스를 증감연산자로 더해주고 출력하는 프로그램이다. 만들어 놓고 보니 그냥 2중 포문사이에 sum = 0으로 선언했어도 될 것 같다는 생각이 들었다.

**실행결과)**

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

**2문제)**

텍스트, 스크린샷, 디스플레이, 폰트이(가) 표시된 사진

자동 생성된 설명

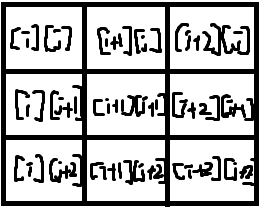
**문제분석)**

먼저 매장량을 난수로 생성하기 때문에 stdlib.h와 time.h를 import해주고 srand(time(NULL))을 사용해 난수를 재정비해주고 rand()%10을 이용해 1~9의 난수를 생성해줘야 할 것 같다. 가장 많은 매장량을 저장해줄 max\_oil을 정수형으로 선언해준 뒤 3x3 구역으로 찾아가면서 높은 매장량을 구하는 이중 포문을 만들어 준다. 3x3구역으로 찾아주므로 for문의 I, j모두 0~4까지로 반복해준 뒤 인덱스를 +1, +2해줘서 그 값이 가장 높은 총 값을 max\_oil에 저장하는 알고리즘으로 가면 될 것 같다. 이번 기회에 if문이 아닌 함축 조건문을 이용해 봐야겠다.

**소스코드)**

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

srand(time(NULL))을 이용해 난수를 재정비해주고 oil\_field[6][6]의 각각의 인덱스를 rand()%10을 이용해 0~9의 난수로 만들어준다. 2중 for문을 이용해 난수로 만든 필드를 출력해준다. 3x3으로 묶어서 가장 높은 매장량을 구하기 때문에 행과 열을 0~4로 반복해주고  이 값을 모두 더해 가장 매장량이 많은 이 값들이 나온 합이 max\_oil이 되어 이 값을 출력하는 프로그램이다. 함축 비교문을 이용하는 것이 익숙하지 않았는데 이번 기회에 익숙해진 것 같다.

**실행결과)**

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명