

다음 내용을 보고 간이 그림판 프로그램을 작성하시오.

main 부분(60점)

간이 그림판 클래스

멤버 변수

- 그림의 각 픽셀 정보를 char로 담은 2차원 nXn 동적 배열을 할당할 2차원 char형 포인터

멤버 함수

- 점 그리기: x, y 좌표와 int 색상 값을 입력 받으면 해당 위치의 픽셀 정보를 수정.
- 픽셀 정보 변경: 픽셀 정보를 실제로 변경하는 함수. 내부 함수로 inline으로 구현.
- 생성자: n을 입력 받아 nXn 크기 동적 배열을 멤버 포인터 변수에 할당
- 픽셀 정보 출력: 저장하고 있는 그림의 픽셀 정보를 화면에 출력.

추가 기능 - 각 함수로 구현

1. 색채우기 기능(60점)

- 가. 함수는 점 그리기처럼 x, y 좌표와 int 색상 값을 입력 받음.
- 나. 만약 해당 좌표의 기존 색상이 입력 받은 색상값과 같다면 함수 종료.
- 다. x y좌표를 저장하는 que를 이용하여 구현.
- 라. 초기 입력 좌표 que에 삽입. 해당 좌표의 이전 색상값은 따로 변수에 저장.
- 마. que가 비어있지 않다면
 - 1) que에서 아이템을 꺼냄.
 - 2) 해당 좌표의 색상과 이전 색상값이 일치하는지 확인.
 - 3) 일치하면 색상 변경 후 인접 픽셀 좌표들을 que에 삽입.
- 바. 마 과정이 완료되면 색상 변경 완료.
- 사. 큐 대신 스택을 사용할 경우 어떤 문제가 발생할지 주석으로 함수 윗부분에 답을 하시오.(20점)

4	8	10	2	7
69	31	77	9	12
54	60	3	7	7
60	60	77	60	7
60	60	4	7	77

4	8	10	2	7
69	31	77	9	12
54	50	3	7	7
50	50	77	60	7
50	50	4	7	77

예시) 좌측 그림 (4,0) 위치에 50 색채우기 사용 시 우측처럼 변경

2. 되돌리기 기능(40점) - 보너스

- 가. 각 편집 명령이 내려질 때마다 변경된 픽셀의 이전 픽셀 정보를 list 형태로 저장되도록 픽셀 정보 변경 함수 및 점 그리기와 색채우기 함수 수정
- 나. 해당 list들과 편집 명령들로 데이터 클래스를 정의하고 해당 데이터 객체를 관리하는 deque 구현
- 다. deque의 크기는 10으로 설정.

테스트 드라이버

모든 입력과 출력은 cout, cin을 기반으로 구현할 것.

모든 편집 명령의 수행 전과 후에 각각 그림 화면 출력 기능을 사용할 것.

```
int main()
{
    //사용자로부터 그림 크기 입력 받음. 그림은 항상 n X n 사이즈 정사각형
    //간이 그림판 클래스 객체 생성 및 n 사이즈 초기화
    //2중 반복문을 통하여 사용자로부터 각 픽셀 정보 입력 받음.
    //입력 받은 그림 화면 출력
    //두 군데 점 찍기 실행
    //입력 받은 그림 화면 출력
    //색채우기 기능 두 번 실행
    //입력 받은 그림 화면 출력
    //되돌리기 기능 실행
    //입력 받은 그림 화면 출력
    //한 군데 점 찍기 실행
    //입력 받은 그림 화면 출력
    //되돌리기 기능 두 번 실행
    //입력 받은 그림 화면 출력
    return 0;
}
```