



윈도우프로그래밍 Windows Programming

Ch03. 제어메시지처리하기 과제

Prof. 윤영욱

과제

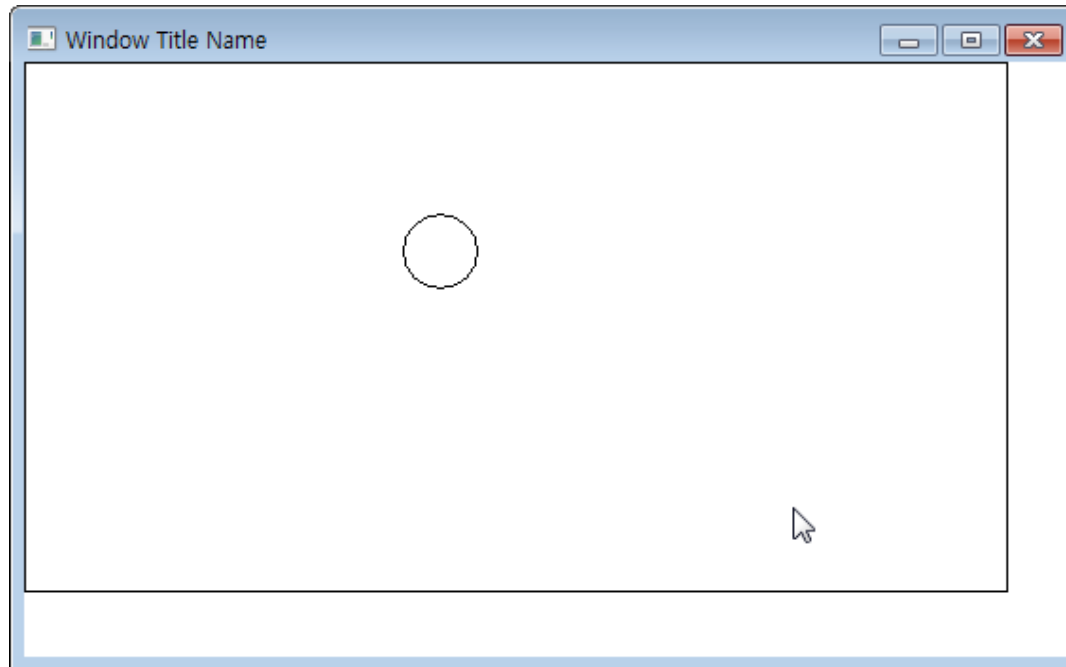
- ✓ 과제 제출시 파일명 : 원프_HW#(주차)_학번_이름
- ✓ 파일 예 : 원프_HW#1_202112345_윤영욱
- ✓ 제출 기한: 공지로부터 (2주간)
- ✓ 모든 과제의 타이틀 바에 본인의 이름 학번이 나타나게 한다.
- ✓ 과제 점수 10점 만점으로 환산 시
 - 기간 내 제출 시 최대 10점, 기간 이후 최대 5점
 - 제출기간 이후 게시판에 제출이 안되면 조교님 메일로 제출
 - 3문제이상 정답 10점, 2문제 정답 9점, 1문제 정답 8점, 소스코드가 있으나 출력이 전부 틀린 경우 7점

1. 방향화살표를 활용하여 윈도우 안에서만 원 이동 시키기

- 중심이 (20,20)이고 반지름이 20인 원을 그린다.
- 방향화살표를 활용하여 원을 4방향으로 이동 시킨다.
- VK_LEFT, VK_RIGHT, VK_UP, VK_DOWN 을 활용한다.
- 이동 단위는 40씩 움직인다.
- [Home] 버튼을 누르면 원이 (20,20)으로 이동한다.
- 단, 원이 만들어진 윈도우의 밖으로 나가지 않게 한다.
- 원이 움직일 수 있는 경계를 검은색 실선으로 그려준다.
- 경계는 아래 함수를 사용한다.
- `Rectangle(hdc, 0, 0, 40 * (int)(rectView.right/40), 40 * (int)(rectView.bottom/40));`

과제

1. 실행예



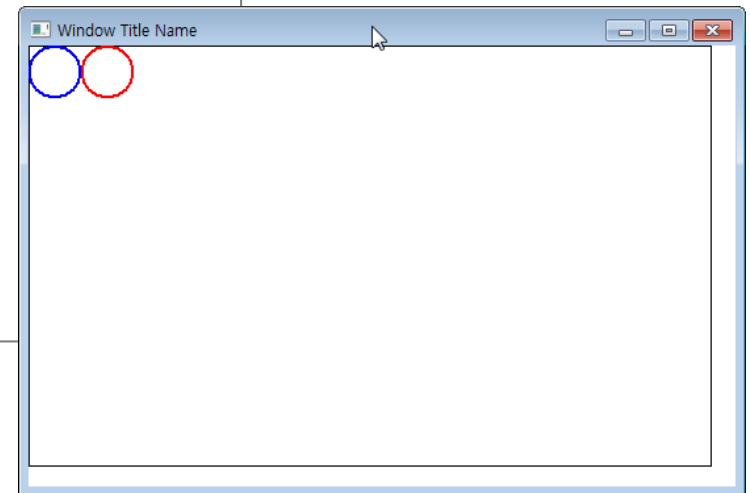
2. 방향화살표를 활용하여 애벌레 이동 프로그램

- 윈도우 화면에 애벌레 역할을 할 두 개의 원을 나타낸다.
- 첫 번째 그린 빨간 원 RGB(255,0,0)은 애벌레의 머리 역할을 하는 것이고 두 번째 그린 파란 원 RGB(0,0,255)은 애벌레의 꼬리 역할을 하는 것이다.
- 원은 실선 (PS_SOLID, 굵기 2)이다.
- 중심이 (20,20)이고 반지름이 20인 원을 그린다.
- 엔터키를 한번 누르면 오른쪽으로 움직이기 시작한다. 다시 누르면 멈추고 또다시 누르면 움직인다.
- 방향화살표를 활용하여 원의 이동 방향을 변경하며 방향이 변경되어도 계속 움직인다.
- 방향이 설정되면 원은 40씩 움직인다.
- 원은 만들어진 윈도우의 밖으로 나가지 않는다.
- 원이 움직일 수 있는 경계를 검은색 실선으로 그려준다.
- 움직임은 100ms으로 한다.
- tip
 - flag 변수를 사용해서 엔터키를 입력 받으면 변화를 준다.
 - case WM_TIMER: 는 다음과 같다. (다음 슬라이드)

과제

```
case WM_TIMER:
    GetClientRect(hwnd, &rectView);
    tmpx = x[0];
    tmpy = y[0];
    x[0] += xStep; y[0] += yStep;
    if (x[0] - 20 < rectView.left || x[0] + 20 > rectView.right)
    {
        x[0] = tmpx;
        y[0] = tmpy;
    }
    else
        x[1] = tmpx;
    if (y[0] - 20 < rectView.top || y[0] + 20 > rectView.bottom)
    {
        x[0] = tmpx;
        y[0] = tmpy;
    }
    else
        y[1] = tmpy;
    InvalidateRgn(hwnd, NULL, TRUE);
    break;
```

실행예



3. 선택한 원의 색상을 변경하기

- 중심이 (20,20)이고 반지름이 20인 원을 그리시오.
- 마우스 클릭으로 원이 선택되면 원의 색상을 변경하시오.
- WM_LBUTTONDOWN, WM_LBUTTONUP을 활용하시오.
- 원 선택 시 원은 실선, 굵기 2, 테두리 빨간색, RGB(255,0,0) 안쪽은 초록색, RGB(0,255,0) 으로 변경
- 원의 색상이 변하는 것은 왼쪽 마우스버튼이 눌러진 동안만 변하게 하시오.

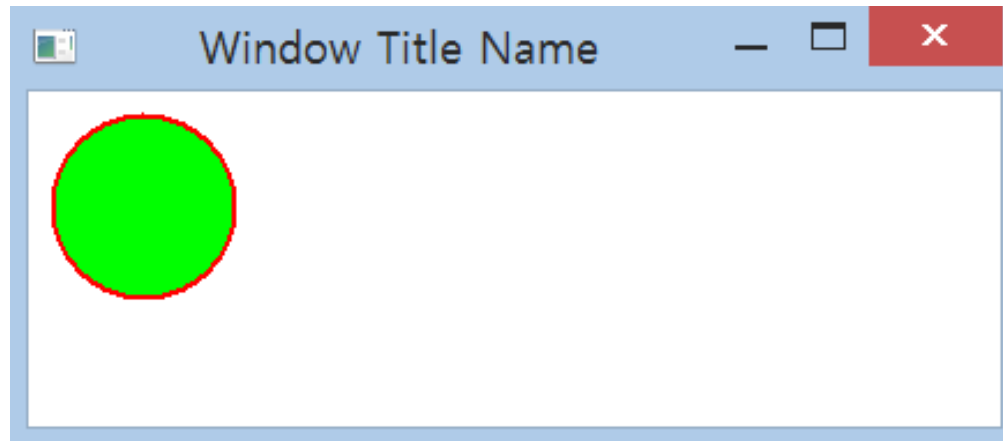
- tip. 선택, 비선택을 bool 형태의 Selection 변수를 활용한다.
- #include <math.h> 추가하고 원 안쪽의 경계를 계산한다.

```
float LengthPts(int x1, int y1, int x2, int y2)
{
    return(sqrt((float)((x2-x1)*(x2-x1) + (y2-y1)*(y2-y1))));
}
```

```
BOOL InCircle(int x, int y, int mx, int my)
{
    if (LengthPts(x, y, mx, my) < BSIZE) return TRUE;
    else return FALSE;
}
```

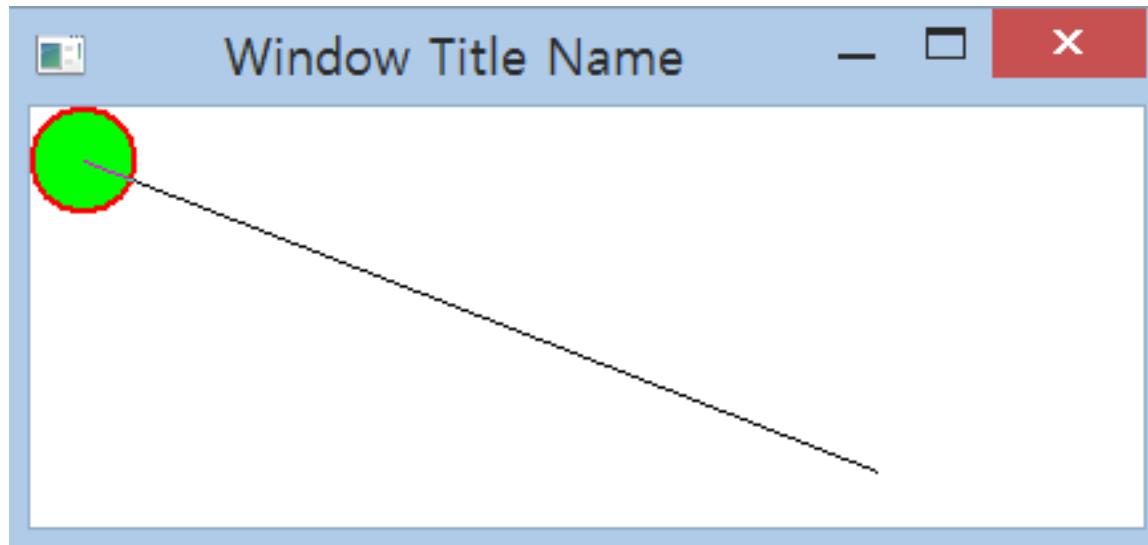
과제

3. 실행 예



4. 선택한 원의 중심에서 마우스 커서까지 직선 그리기

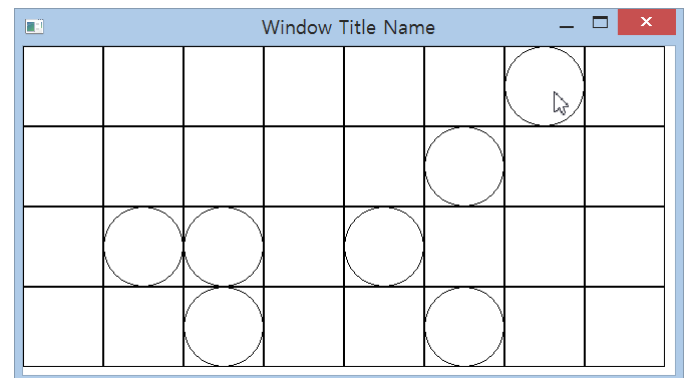
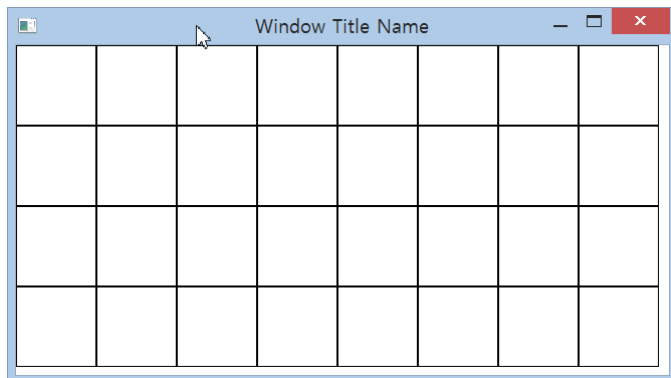
- 중심이 (20,20)이고 반지름이 20인 원을 그리시오.
- 왼쪽 마우스버튼을 눌러서 원이 선택되면 원의 중심부터 드래그 중인 마우스의 커서까지 직선을 그리시오.
- 레스터 연산으로 직선을 그린다. SetROP2(hdc, R2_XORPEN)
- 색정보는 3번 문제와 같다
- 마우스버튼에서 손을 놓으면 직선은 사라지게 하시오.



5. 격자 속에 클릭한 위치마다 원 나타내기

- 프로그램을 실행하면 가로 8줄, 세로 4줄짜리 격자가 나타난다.
- 정사각형 변의 길이는 80이다.
- 격자 내부에서 마우스 왼쪽버튼을 클릭하면 마우스 커서가 포함된 격자에 원이 나타난다.
- tip. 아래와 같이 마우스 좌표를 박스의 사이즈로 나누고 해당 좌표에 TRUE 값을 넣는다.

```
case WM_LBUTTONDOWN :  
    x = LOWORD(lParam) / BSIZE;  
    y = HIWORD(lParam) / BSIZE;  
    if ( x < 8 && y < 4 )  
        rect[x][y] = 1;
```





END of document