**1. HISC를 기록하고 불러오도록 합니다.**

1) HISC 텍스트 파일 생성 후 writeline readline 코드 작성하기

**2. 게임 공간을 8x10에서 10x12공간으로 바꿉니다. (이미지)**

2) 게임 공간 이미지 만들기

**3. 바뀐 공간에 대해서도 동일하게 게임이 동작하도록 합니다. (코드)**

3) line 199에 있는 neko\_bg.png 새로운 이미지 파일로 바꾸기 + 범위 지정 새로 하기

**4. 파괴된 블럭을 화면에 표현하고, 10개가 넘을 때 마다 보너스 10점씩 추가합니다.**

4) 파괴된 블록(이걸 어떻게 감지하지??)의 이미지를 화면의 특정 공간에 나타나도록 한다. (좌표를 정하고 이미지를 place 하면 될 것 같다.)

5) 파괴된 블록의 개수(이건 또 어떻게 측정함??)를 저장할 변수를 선언하고, while True로 무한 루프 만들어서 10개가 되면 변수의 값을 0으로 저장하고 보너스 점수를 score변수에 더한다. (다만 게임 플레이 코드 부분 전체에 반복문을 걸어야 할지 의문)

🡺 블록이 파괴되었음과 그 블록이 무엇인지 구별할 코드 필요, 반복문, 조건문?의 범위 조율

**5. 4개 블럭이 네모로 놓여 있을 때에도 블럭이 파괴됩니다.**

6) 2x2배열도 매치로 인식될 수 있게 check\_neko 함수에 코드를 추가한다. 기준점(왼쪽 위)을 [y][x]로 두고 오른쪽 위[y][x+1], 왼쪽 아래[y+1][x], 오른쪽 아래[y+1][x+1]의 neko가 같은 지 검사한다. 같다면 해당하는 4개의 블록 값을 모두 7로 바꾼다.(삭제 🡪 sweep\_neko 실행 조건)

6. 게임 시작 시 경기 시간을 화면에 표시합니다. (정확하지 않아도 되며 프레임 단위로 구현합니다.)

7. 5초 이상 블럭을 배치하지 않는 경우 블럭이 바로 떨어집니다.

8. 'esc'버튼을 누르는 경우 '게임을 종료하시겠습니까?' 라는 경고창이 뜨며, '예' 버튼을 누르는 경우 메인 화면으로 돌아갑니다.

9. 블럭 배치 5번 마다 조커 블럭을 배치할 수 있게 됩니다. 조커 블럭은 모든 블럭으로 해석 될 수 있습니다. (5번 횟수에 시간초과로 블럭이 배치되는 경우는 포함되지 않습니다.)

10. 1턴 이후에 조커 블럭이 랜덤한 일반 블럭으로 변경됩니다.