

Programming Assignment #3

Tetris

Due Date: 2020년 5월 15일 금요일 23:59

담당조교: 장원종 (wonjong@postech.ac.kr)

1. Objective

본 과제에서는 고전 오락게임인 tetris를 구현해보고, 이를 통해 inheritance 개념과 polymorphism 개념을 익혀보도록 한다.

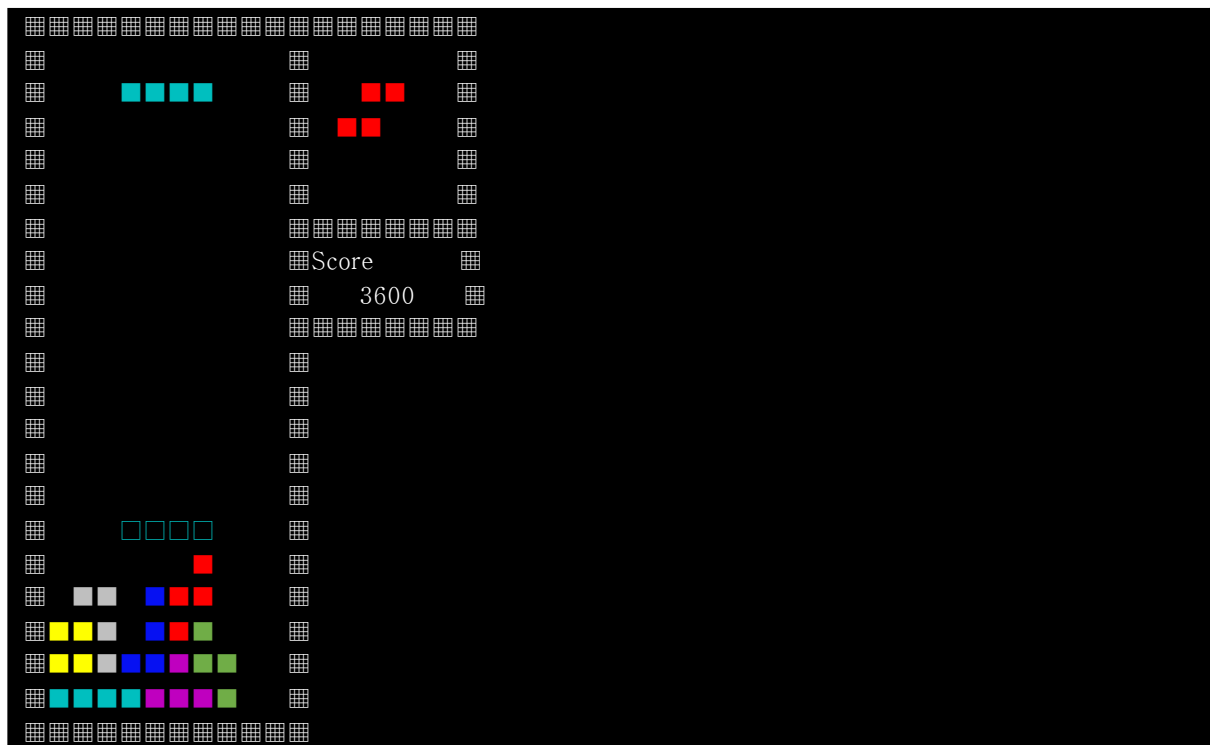


Figure 1. 게임 실행 예시

2. 과제 설명

본 과제 시작 전 tetris라는 게임에 익숙하지 않은 학생들은 과제 시작 전 구현할 프로그램에 대한 감을 잡기 위해 <https://tetris.com/play-tetris> 에서 실제 상용되어지고 있는 테트리스 게임을 직접 해보는 것을 권장한다.

2.1 블록



Figure 2. 테트리미노의 종류

테트리스의 각 블록인 테트리미노는 다음과 같이 7가지 종류로 구성되어 있다. 각각의 이름은 모양을 본 때 I, O, T, L, J, S, Z라 부른다 (Fig. 2). 이 때, L, J는 언뜻 이해가 안될 수 있는 데, L과 J는 세웠을 때의 모양을 기준으로 한다 (Fig. 3).



Figure 3. L 미노와 J 미노의 구분

본 과제에서 color print 기능은 주황색을 지원하지 않으므로 주황색 L 블록은 Fig. 1에서 보이는 것처럼 하얀색으로 구현하도록 한다.

2.2 블록 생성



Figure 4. 테트리미노 생성 규칙

미노는 완전히 무작위로 나오는 것이 아니라 나오는 규칙이 존재한다. 7종의 미노는 위의 예시처럼 한 묶음으로 나와야 한다.

예를 들어, n 번째 나오는 미노를 n 번 미노라 하자. 그러면 1~7번 미노는 I, O, T, L, J, S, Z가 무작위

로 나온다. 그리고 또 8~14번 미노는 I, O, T, L, J, S, Z가 무작위로 나오게 된다.

이와 같은 규칙성을 바탕으로 구현하면 플레이어는 앞으로 어떤 블록이 나올 지 어느 정도 예측할 수 있고 원활하게 플레이할 수 있다.

2.3 필드(Playfield)

게임이 이루어진 평면을 의미한다. 기본적으로 필드의 크기는 10x20으로 구성되어 있다 (See Fig. 5). 테트리스미노가 playfield 위쪽으로 나가게 된다면, 게임은 player의 패배로 종료된다.

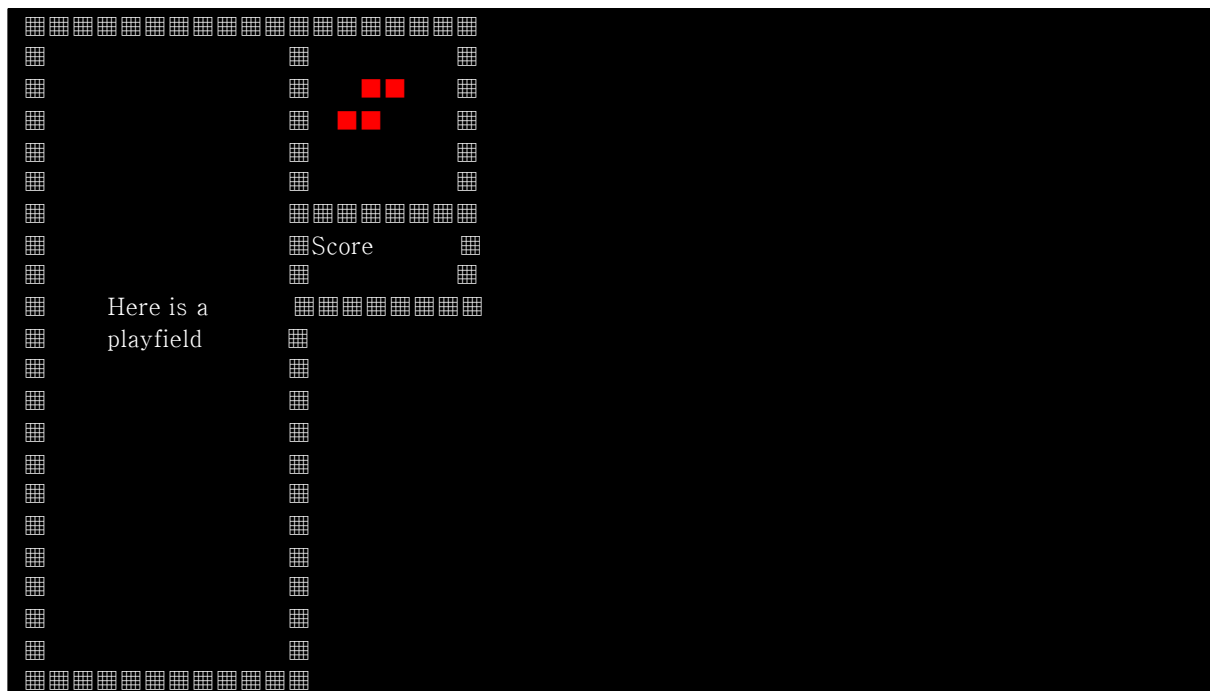


Figure 5. playfield 예시

2.4 T스핀(T-Spin)

T스핀(T-spin)은 T미노를 이용해 특별한 방법으로 줄을 없애는 기술을 의미한다. 기본적인 동작원리는 T미노가 바닥에 닿아 굳어지기 전에 회전을 하여 라인클리어를 하는 것이다. 2001년에 나온 테트리스 가이드라인을 지키는 게임들은 이 규칙이 적용되고, 2005년에 나온 Tetris DS 이후 지금의 형태로 유지되어 오고 있다. 자세한 설명은 <https://hse30.tistory.com/82> 에서 움직이는 그림과 함께 보면 이해가 쉽다. 본 과제를 구현할 시 T-spin이 가능하도록 구현하여야 한다.

2.5 요구 사항

(0) 테트리미노

다음 사항들은 테트리스라는 게임을 구성하기 위한 기본적인 요소들이지만 설명이 필요한 학생들을 위해 정리하였다.

- 모양별로 다른 색의 테트리미노가 등장하여야 한다.
- 한 줄이 블록들로 꽉차게 되면 그 줄을 지우고 그 위의 테트리미노들을 아래로 떨어뜨려야 한다.
- 필드(Playfield) 우측에는 다음에 나올 테트리미노를 미리 표시하여 준다.
- 테트리미노가 필드 위쪽으로 나가게 된다면, 게임은 종료된다.
- 테트리미노는 필드 바깥으로 나가거나 서로 겹칠 수 없다.
- 테트리미노의 회전은 위의 예시 게임과 동일한 회전축을 사용하여 구현한다.
- 그 외 자세한 사항은 위의 상용 게임을 참고한다.

(1) 점수 체계

- 동시에 1줄 삭제 시 100점, 2줄 삭제 시 300점, 3줄 삭제 시 500점, 4줄 삭제 시 1000점을 부여한다.

(2) 속도 조절

- 1000점 단위로 테트리미노가 떨어지는 속도가 +10%가 된다. 즉 3000점이면 초기 속도의 +30%한 1.3배로 적용한다.

(3) 그림자

- 그림자란 테트리미노가 떨어질 위치를 사영하여 미리 표시해주는 것을 의미한다.
- 그림자 모드는 키고 끌 수 있어야 한다.

(4) 조작키

- a, d : 테트리미노의 좌우 이동
- s : 테트리미노의 아래 이동
- w : 테트리미노의 시계방향 회전
- x : 테트리미노의 반시계방향 회전
- m : 그림자 모드 키고 끄기

- space : 테트리미노 바로 내리기

(5) 게임 종료 및 재시작

```
Do you want to save your score? (y/n) y
Do you want to restart? (y/n) n
```

Figure 6. 게임 종료 시 동작

- 게임이 종료되면, 점수를 저장할 것인지 물어보고 점수를 저장한다면 score.txt에 점수를 기록한다. 그 후 게임을 다시 시작할 것인지 물어보고 다시 시작한다면 게임을 처음부터 다시 진행하면 되고, 아니라면 프로그램을 종료한다.

- 점수는 score.txt 에 시간순으로(오래된 것이 아래에) 차례차례 적어둔다.

```
3600
7000
6500
10000
```

Figure 7. score.txt 예시

(6) 추가 구현 및 추가 점수

- 이번 과제에는 추가 구현에 따른 추가 점수를 받을 수 있는 항목이 존재한다.
- 아래의 항목들은 필수 구현항목은 아니나 구현할 경우 채점 시 +10점까지의 추가점수를 부여한다.

1) T-spin 에 따른 점수체계

- T-spin 미니 : 100점
- T-spin 싱글 : 500점
- T-spin 더블 : 1000점
- T-spin 트리플 : 1500점

T-spin으로 얻은 점수와 라인클리어로 얻은 점수는 중복되지 않는다. 즉, T-spin 싱글로 500점을 얻었다면, 라인클리어로 100점을 얻지 않아야 한다.

2) Back-to-back에 따른 점수체계

- 200점의 추가 점수 부여

3) 콤보에 따른 점수 체계

- 2 콤보부터 100*(콤보수-1)점의 추가 점수 부여

- 추가 구현을 한 경우 추가 점수를 받기 위해 반드시 보고서에 구현 방법과 내용을 명시한다.

3. Tips

3.1 Block.cpp

MinGW 환경에서 color print 에 익숙하지 않은 학생들을 위하여 본 과제에 사용할 수 있는 single block class에 대한 예시 코드를 제공한다.

Single block class는 테트리미노 및 필드의 작은 하나의 블록을 구성하는 단위이다. 블록을 출력할 수도 있으며, 원하는 경우 색깔과 메시지를 지정하여 원하는 색깔의 메시지를 출력할 수도 있다.

또한 코드의 예시처럼 single block class 간의 연산도 정의하여 편리하게 이용할 수도 있다.

Block.cpp는 예제 코드이므로 수정을 원한다면 멤버 변수 및 함수나 operator overloading을 추가하는 등 자유롭게 수정하여 사용하여도 좋다.

참고로 위의 코드의 color print는 system dependent한 방법으로 구현되어 있으므로 MinGW가 아닌 환경에서는 정상적으로 동작하지 않을 수 있다.

3.2 화면 지우기

<stdlib.h>의 system("clear") 를 통해 지울 수 있다.

4. Design

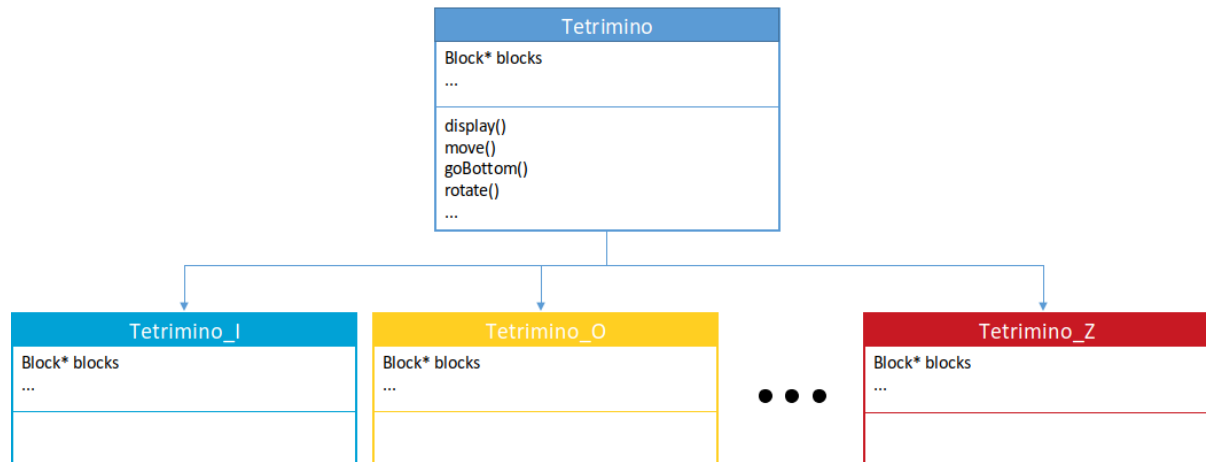


Figure 8. Tetrimino class 설계도

본 과제는 inheritance와 polymorphism의 개념을 익히기 위한 과제이므로 테트리미노 구현 시 위의 Fig. 8 처럼 반드시 상속을 이용하여 구현한다.

- blocks는 tetrimino를 구성하는 block의 집합으로 이에 대한 자료구조는 배열이나 링크드리스트를 이용하여 구현한다.

- display(...)는 tetrimino를 화면에 출력하는 함수이다.

- move(...)는 키 입력을 받아 충돌 검사를 해가며 입력된 키에 알맞은 이동을 시키는 함수이다.

- goBottom(...)은 space를 눌렀을 때의 동작에 해당하는 함수로 테트리미노를 바로 아래로 내린다.

- rotate(...)는 테트리미노의 회전과 관련된 함수이다.

그 외의 게임 구성을 위한 객체들은 자유롭게 설계하도록 한다. 이 때, assingment 2와 마찬가지로 본인이 한 설계에 대한 그림과 설명을 반드시 보고서에 명료하고 보기 쉽게 정리하여 제출한다.

5. 과제 제출

1) 제출 파일은 **assn3.zip, assn3_report.pdf** 형태로 **두가지 파일을 따로 업로드**할 것

2) 실행 파일 파일을 제외한 나머지 **makefile, 소스 및 헤더 파일**만을 제출하고, 해당 압축파일을 풀었을 때 **make** 커맨드만으로 컴파일이 가능하도록 제출할 것