**2022 Spring OOP Assignment Report**

과제 번호 : 4

학번 : 20210661

이름 : 오승준

Povis ID : seungjunoh

**명예서약 (Honor Code)**

나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.

I completed this programming task without the improper help of others.

1. **프로그램 개요**

* 이번에 제출한 프로그램은 class 관련 여러 기능들을 활용하여, 기하(geometry) 객체를 정의하고 이를 통한 기능을 구현하는 SimpleGeoWorld를 구현하는 것이다.

이 프로그램은 class 상속, class template, STL 상속 등 class에 관한 다양한 기능들을 사용하여 class에 익숙해지는 것을 목표로 한다.

프로그램을 실행하기 위해서는 내부 Visual Studio 프로그램(20210661.sln)을 실행하고, 로드를 수행하면 된다.

파일 구성에 대한 설명은 아래와 같다.

20210661.sln : VS project file

Canvas2d.h : canvas2d class 선언 및 정의 헤더파일

Geometry,h : geometry class, point class, rectangle class 선언 및 정의 헤더파일

Parser : parser class 선언 및 정의 헤더파일

Main.cpp : 프로그램 실행을 위한 메인 코드파일

1. **프로그램의 구조 및 알고리즘**

* 사용한 class의 설명은 다음과 같다.
* Geometry class : abstract class. 하위 클래스들의 draw2canvas를 수행하기 위한 class
* Point<T, int N> : N(2~5) 차원 점을 표현하는 class.
* Rectangle<T> : 2차원 사각형을 표현하는 class. 2차원 상 두 사각형의 위치 관계를 파악하는 함수가 포함되어있다.
* Parser : canvas, geometry에 대한 정보를 string으로 입력받고 이를 각 class에 해당하는 형식으로 변환하여 전달하는 class이다.
* Canvas2d class : canvas를 생성하고 입력된 geometry들을 출력하는 class이다.
* 전체적인 알고리즘에 대한 설명.

문제 1의 알고리즘은 아래와 같다.

* 1. Problem1을 입력받는다.
* 2. Canvas에 관한 정보를 입력받는다.
* 3. Parse\_canvas를 통하여 정보에 해당하는 canvas를 구현한다.
* 4. Geometry에 대한 정보를 여럿 입력받는다.
* 5. Parse\_geometry를 통하여 정보에 해당하는 geometry(point, rectangle)을 구현한다.
* 6. Draw2stdout에서 iterator를 통하여 이를 하나씩 순차적으로 draw2canvas를 수행하여, canvas에 해당하는 geometry에 따라 canvas의 해당 위치 값을 변화시킨다.
* 7. 이를 모두 반복하고, 값이 0인경우는 빈칸에 해당하는 문자를, 값이 존재하는 경우는 값의 10의 나머지를 출력한다.

문제 1에서 사용한 변수들에 대한 설명은 다음과 같다.

-canvas2d의 멤버변수 : width(너비), height(높이), c\_empty(빈칸에 해당하는 문자), \*\*canvas(2차원 캔버스)

-point의 멤버변수 : arr[N](point 차원 정보를 담는 사이즈 N의 1차원 배열)

-rectangle의 멤버변수 : l\_x,l\_y(1사분면에 좌표 표현시 왼쪽 아래 사각형의 x,y좌표), r\_x,r\_y(오른쪽 위 사각형의 좌표)

문제 2의 알고리즘은 아래와 같다.

* 1. Problem2를 입력받는다.
* 2. 자료형과, 사각형 2개의 좌표에 대한 정보를 입력받는다.
* 3. 생성된 두 rectangle에 대하여, 각각의 l\_x, l\_y를 비교하여 큰 것을 취하고, r\_x, r\_y 중 작은 것을 취하여, 교차지점을 파악하기 위한 점 4개 tl\_x,tl\_y,tr\_x,tr\_y를 각 함수에서 연산한다.
* 4. 4개의 변수의 값에 따라 5개의 위치 비교 함수의 값을 출력한다.

문제 2에서 사용하는 변수들은 문제 1에서와 동일하다.

1. **토론 및 개선**

문제3>canvas2d class 처럼 STL vector를 상속받을때 예상되는 단점

* STL vector를 상속하게 되는 경우, vector가 상속되는 것은 맞으나, vector의 함수들의 overloading이 제대로 이루어 지지 않는다. Overloading을 수행하려 하는 경우 overriding( 상속 관계에 있는 부모 클래스에서 이미 정의된 메소드를 자식 클래스에서 같은 시그니쳐를 갖는 메소드로 다시 정의하는 것)이 일어나며 제대로 vector의 함수를 불러오지 못하는 문제가 나타난다. Vector의 함수를 쓰고 싶으면 using-declaration을 사용해야하는 단점이 있다.
* 또한, STL vector 의 경우 소멸자가 가상소멸자가 아니라는 단점이 있다.

이번 ASSN4를 진행하며 class에 대한 다양한 기능들을 느낄 수 있었다. 문제3에선 단점이 있음을 다루었으나 vector를 상속받아 class를 구현할 수 있다는 것을 보며 class가 객체지향의 꽃이라 불리는 이유에 대해 느낄 수 있었다.

이번 프로그램을 제작하는 과정에서 개선하면 좋을 것으로 예상되는 점은, rectangle class의 함수들 중, 두 사각형의 위치관계를 파악하는 함수들에서 각 함수 내부에서 두 사각형의 위치관계를 파악하기 위한 좌표 4개를 파악하는데, 이 과정이 모든 함수에서 동일하다. 이를 rectangle의 멤버변수로 바꿔서 구현한다면 조금더 편하지 않을까 생각하였다.

1. **참고 문헌**
   * + https://kldp.org/node/42919- stl vector 상속에 대한 단점을 파악하기 위해 사용한 사이트이다.
     + https://blog.naver.com/PostView.nhn?blogId=xtelite&logNo=50173791450- rectangle 함수 구현 중 도형들 위치 관계를 파악하기 위한 알고리즘을 이 사이트에서 영감을 얻어 구현하였다.