**2022 Spring OOP Assignment Report**

과제 번호 : 4

학번 : 20210273

이름 : 하태혁

Povis ID : hth021002

**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**

이 프로그램은 기하 객체(rectangle, point)를 저장하고 출력이나 비교하는 기능을 구현하는 것을 목적으로 한다. 사용자는 stdin을 통해 문제 종류를 입력하고 문제 조건에 맞는 기하 객체를 입력한다. 이후, 조건에 맞게 stdout에 도형의 정보를 출력한다.

소스 파일에는 main.cpp파일이 존재하며 사용자의 입출력을 비롯한 함수의 호출이 해당 파일에서 일어난다. 헤더 파일에는 canvas2d.h와 geometry.h이 존재하다. geometryh에는 geometry, rectangle<T>, point<T, int N>가 선언, 정의되어 있고 canvas2d.h에는 canvas2d, parser가 선언, 정의되어 있다.

1. **프로그램의 구조 및 알고리즘**

* geometry class

rectangle<T>, point<T, int N> 클래스의 부모 클래스로 순수 가상 메소드인 draw2canvas가 선언되어 있다.

* rectangle<T>

입력된 좌표를 저장하는 변수 x1, y1, x2, y2가 선언되어 있다. Assignment4 보고서에 있는 메소드만 사용하여 해당 기능을 구현하였다.

is\_\*\*\*\* 함수를 구현할 때, 왼쪽 위 좌표와 오른쪽 아래 좌표를 기준으로 사용했기 때문에 STL algorithm에 저장되어 있는 min(), max(), minmax()를 사용하였다.

* point<T, int N>

point의 좌표를 저장하는 변수 x, y가 선언되어 있고 차원에 해당하는 숫자를 더한 값을 저장하기 위한 변수 sum이 저장되어 있다. 차원값이 없을 경우 기본 값은 1이기 때문에 sum을 선언할 때 1로 초기화 하였다.

그외 메소드는 모두 보고서에 저장되어 있는 방식과 설명을 따랐다.

* canvas2d

이차원 canvas의 크기를 저장하기 위한 int형 변수 x, y가 선언되어 있고 canvas의 기본 공간을 채우는 latter를 저장하기 위한 char형 변수 c가 선언되어 있다. 이차원 canvas는 char\*\* 형 변수 board로 선언되어 있다.

std::vector<geometry\*>를 상속받았기 때문에 draw2stdout()에서 Iterator를 선언하여 canvas2d에 저장되어 있는 각 객체 요소에 접근해 기하 객체를 board에 반영하였다.

* parser

geline을 통해 stdin으로 받은 입력 값을 반점 기준으로 자르고 vector<string> v에 저장하여 사용하였다.

1. **토론 및 개선**

STL vector를 상속받아 사용하면서 상속의 기능을 다시 한번 더 이해하게 되었다. 또한 STL algorithm을 사용해보고 pair<T,T> 개념도 공부해 보면서, C++의 활용 부분에 대한 이해도를 높일 수 있었다.

문제3 > STL vector를 상속받을 때 예상되는 단점

STL vector의 경우 소멸자가 virtual이 아니기 때문에 따로 정의하지 않으면 안된다. 만약 동적할당을 한다면 distructor에서 할당해제를 해줘야 하지만, 상위 클래스의 distructor를 불러올 수 없기 때문에 메모리 누수가 일어날 수 있다.

또한 copy constructor와 assignment operator를 구현할 때도 deep copy로 재정의하여 사용하지 않는면 동적할당 받은 변수가 copy, assignment 될 때 문제가 발생하게 된다.

1. **참고 문헌**

<https://blockdmask.tistory.com/70>

[https://notepad96.tistory.com/57 - :~:text=min, max, minmax는 algorithm,최대값, 최소값을 구한다.](https://notepad96.tistory.com/57#:~:text=min%2C%20max%2C%20minmax%EB%8A%94%20algorithm,%EC%B5%9C%EB%8C%80%EA%B0%92%2C%20%EC%B5%9C%EC%86%8C%EA%B0%92%EC%9D%84%20%EA%B5%AC%ED%95%9C%EB%8B%A4.)