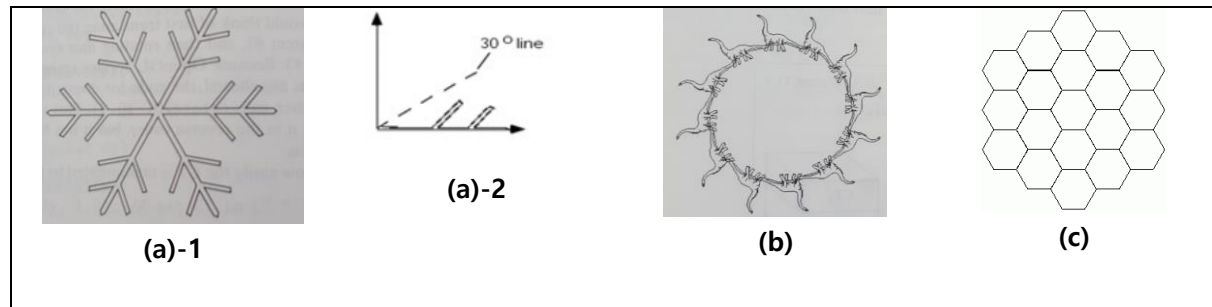


1. 코드 실행 결과를 볼 수 있는 실행 화면 캡처, 어떻게 코드를 짰는지와 이에 대한 설명 및 토의 사항을 적은 리포트 (pdf 포맷)와 실행 코드 (.cpp)들을 하나의 파일로 압축하여 제출
2. 모든 문제에 어떻게 가시 공간, 변환, CTM등을 사용하였는지 간단히 설명하고 물체나 polyline의 색은 여러 분이 보기 편하도록 적절히 선택한다.

1. 물체의 변환을 이용하여 아래 그림 (a)-1과 같은 눈송이 모양을 만들고자 한다. 이 눈송이는 대칭형태로 각 spoke (살)는 6개가 있고 각각은 같은 모양으로 60° 씩 떨어져 있다. 각 spoke를 이루는 축과 가지는 축을 기준으로 60° 의 각도를 이루고 있다. 눈송이 모양을 만들기 위하여 기본이 되는 하나의 spoke를 완성한 후 이를 60° 씩 회전하면서 `glRotatef` 함수를 6번 반복하여 그려보고자 한다. 먼저 기본이 되는 하나의 spoke를 polyline을 이용하여 만들어 보자. (a)-2를 참조하여 만들어 보자. spoke는 또다시 각 spoke를 이루는 축과 대칭으로 되어 있으므로 위쪽 반만 만든 후 `glScalef` 함수를 이용하여 반사 (reflection)시켜서 완성하자.

2. 수업시간에 배운 CTM과 복합 변환을 이용하여 아래 그림 (b)와 같은 모양을 만들어 보자. 각각의 dinosaur는 회전된 형태로 각각의 발은 원점을 향하는 모양이다.

3. 보드 게임을 보면 아래 그림 (d)와 같은 hexagonal grid를 자주 보게 된다. Hexagonal grid를 이루는 각각의 hexagon은 6개의 같은 길이의 변으로 이루어진 6-gon으로 각 내각은 120° 총 내각의 합은 720° 이다. . 가시 공간 안에 모든 hexagon들이 들어오도록 변의 길이는 적절히 주자. CTM과 복합 변환을 이용하여 총 19개의 hexagon이 붙어 있는 (d) 형태의 hexagonal grid를 만들어 보자.



4. 아래 Dropbox 링크에 있는 코드와 같이 `gluLookAt()` 함수에서 eye와 center 위치를 정하였다. Eye의 위치는 고정하고 center의 위치만 아래 4개와 같이 변경하였을 때 물체의 출력 결과를 보여주고 왜 그렇게 보이는지 설명해 보자. 종이를 이용하여 frustum과 eye, center의 위치와 LOS를 표시해 보고 이를 이용하여 설명해 보자 (종이에 그린 것을 스크린 캡처). 이해하기 쉽게 frustum을 위에서 내려다 본것과 같이 z축과 x축만을 이용하여 2D로 설명해보자.

https://www.dropbox.com/s/uy5tnsas80p7ro4/HW2_4.txt?dl=0

(1) 0.0, 0.0, -10.0 (2) 0.0, 0.0, 20.0 (3) 0.0, 0.0, 15.0 (4) 5.0, 0.0, 0.0