

컴퓨터 그래픽스

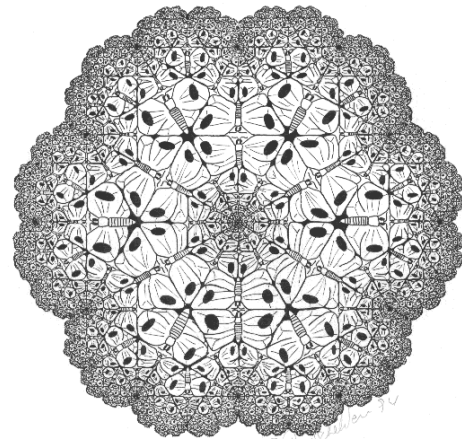
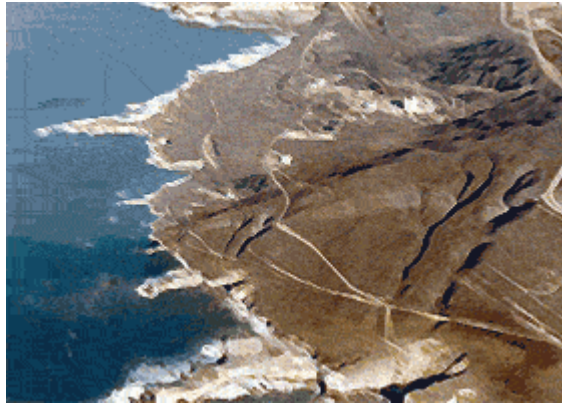
10. 절차적 그래픽스 기법

2022년 2학기

프랙탈

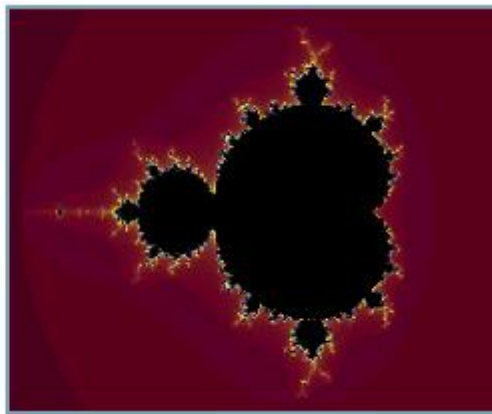
- [Fractal Model](#)

- 1974년 Mandelbrot이 프랙탈 기하이론을 통해 산, 구름, 해안선, 나무 등의 자연물 등을 표현
- 프랙탈: 불규칙해 보이는 자연물을 묘사하기에 좋은 수단
- 비정수 차원(Fractional Dimensional) 이라는 용어에서 유래



프랙탈

- 프랙탈의 특징
 - 비정수적 차원
 - 유클리드 기하학과는 다르게 비정수 차원으로 존재
 - 자기복제의 성질(Self-similarity Property)
 - 어떤 형태가 순환적으로 반복되어 전체 형상과 유사한 형태를 국부적인 부분에서도 발견
 - 무한대로 순환 반복(Infinite detail at every point)
 - 그림의 작은 부분을 확대하면 새로운 그림이 생성



나무의 자기 복제적 특징

무한대로 순환 반복하는 Mandelbrot 집합

- 프랙탈은 자기 반복적인 성질을 가지고 있는 대상물 묘사에 적합
 - 나무, 해안선 등 자연물 표시에 적절하다

프랙탈의 생성 원리

- 프랙탈 생성 원리
 - 임의적 반복 알고리즘(Random Iteration Algorithm)
 - 원칙 또는 규칙 몇 가지를 정한 후 무작위로 규칙을 반복 적용하면 프랙탈이 생성
 - 자기 순환적 복제
- 프랙탈 기하학은 간단한 절차(Fractal Generation Procedure)에 의해 정량적으로 표현
$$P_1 = F(P_0), P_2 = F(P_1), P_3 = F(P_2), \dots$$
이때, F : 변환 함수 (규칙적 혹은 임의 변이)

프랙탈의 생성 기법

- 프랙탈 생성 기법

- IFS (Iterated Function System)

- 자기복제적 성질을 가지는 함수나 기하변환을 통하여 초기 객체의 형태와 유사한 객체들을 반복적으로 복제해 나간다.
 - 대표 예) [Barnsley](#) 고사리, [IFS](#) 이용 애니메이션

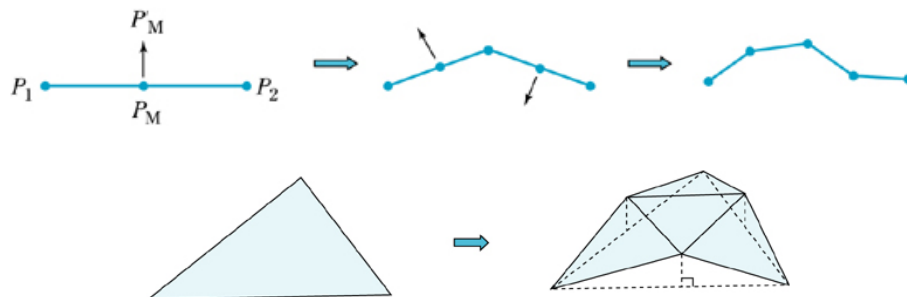


- 생성 문법 (Production Grammar) 이용 방법

- 생성 규칙을 반복적으로 적용하여 새로운 모양을 만들어가는 방법
 - 대표 예) dragon curve

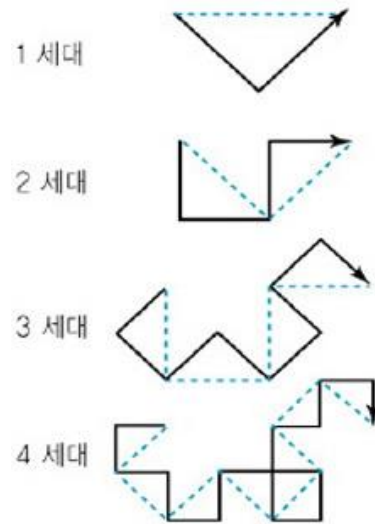
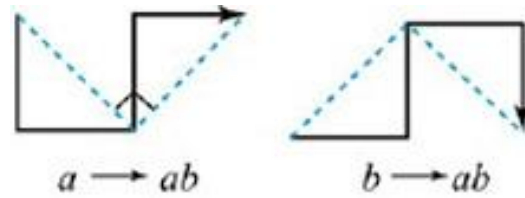
- 중점변위법 (Midpoint Displacement Method)

- 주어진 변의 중점을 정규분포로부터 생성된 난수만큼의 거리로 변위시켜 새로운 두 개의 변을 만들고 이러한 과정을 다시 변위시켜 새로운 변을 만들어가는 방법
 - 예) [산의 표면](#)



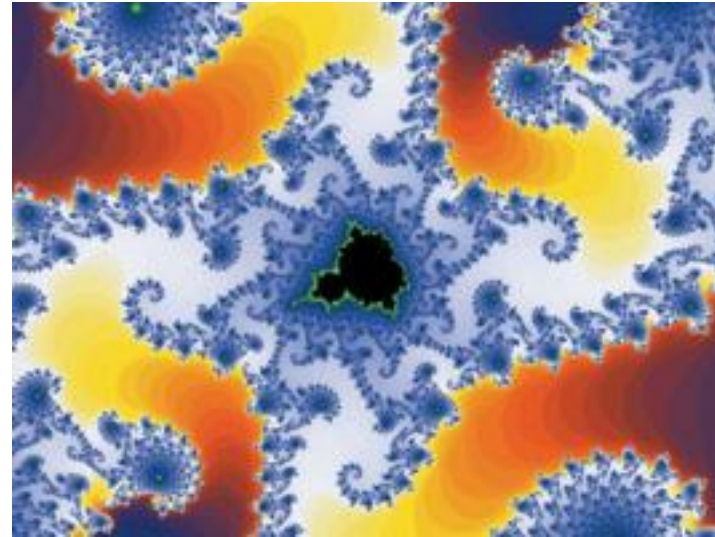
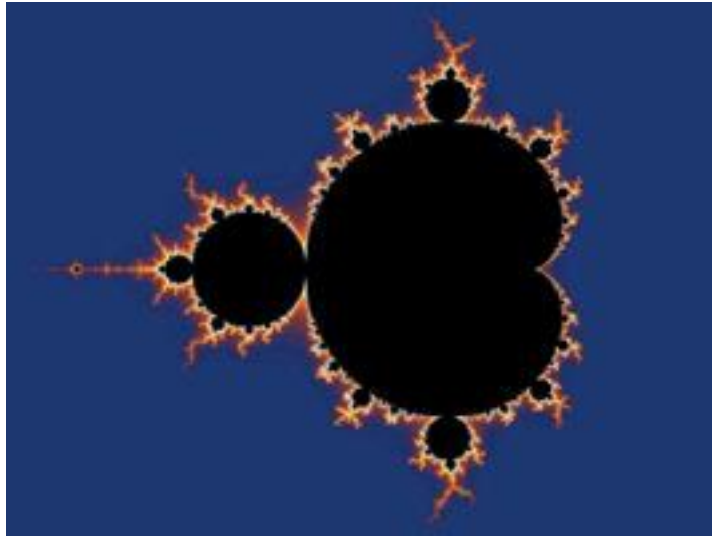
프랙탈의 생성 원리

- 프랙탈의 예: Dragon curve



프랙탈의 예

- Mandelbrot 집합
 - 가장 대표적인 프랙탈 기하학
 - 자기 순환적인 복제가 무한히 반복



프랙탈 기하학의 차원

- 프랙탈 차원

S : 생성되는 객체의 크기(Scaling factor)

N : 생성되는 객체의 수(Number of subparts)

$$NS^D = 1 \quad \rightarrow \quad D = \ln \frac{1}{N} / \ln S = \ln N / \ln \frac{1}{S}$$

- 프랙탈의 차원을 유클리드 기하학에 적용

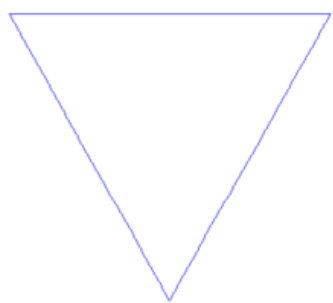
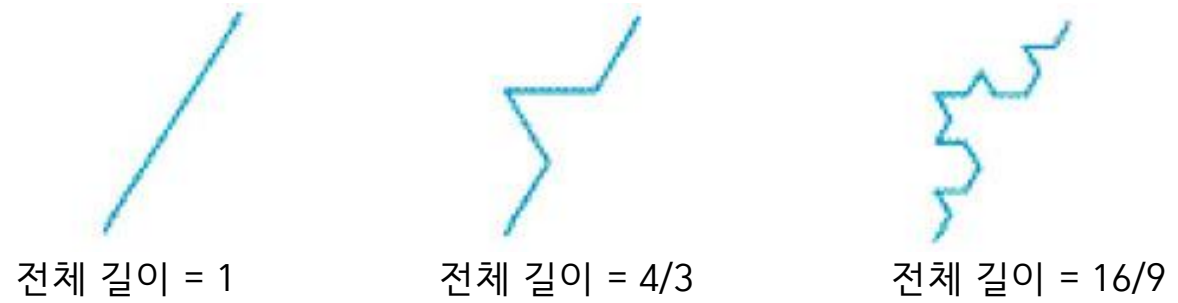
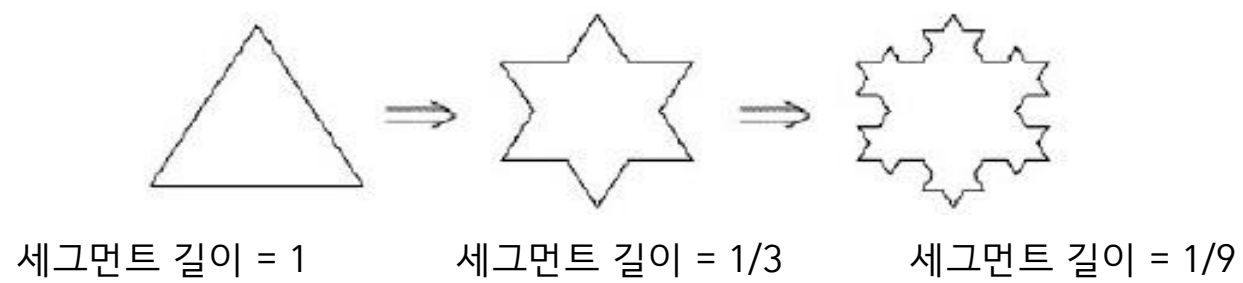
$$D=1 \quad \begin{array}{c} S=1 \\ \hline N=1 \end{array} \Rightarrow \begin{array}{c} S=1/2 \\ \hline N=2 \end{array} \Rightarrow \begin{array}{c} S=1/3 \\ \hline N=3 \end{array}$$

$$D=2 \quad \begin{array}{c} \square \\ N=1 \end{array} \Rightarrow \begin{array}{c} \begin{array}{|c|c|} \hline \square & \square \\ \hline \square & \square \\ \hline \end{array} \\ N=4 \end{array} \Rightarrow \begin{array}{c} \begin{array}{|c|c|c|} \hline \square & \square & \square \\ \hline \square & \square & \square \\ \hline \square & \square & \square \\ \hline \end{array} \\ N=9 \end{array}$$

$$D=3 \quad \begin{array}{c} \text{cube} \\ N=1 \end{array} \Rightarrow \begin{array}{c} \begin{array}{|c|c|c|} \hline \text{cube} & \text{cube} & \text{cube} \\ \hline \text{cube} & \text{cube} & \text{cube} \\ \hline \text{cube} & \text{cube} & \text{cube} \\ \hline \end{array} \\ N=8 \end{array} \Rightarrow \begin{array}{c} \begin{array}{|c|c|c|c|} \hline \text{cube} & \text{cube} & \text{cube} & \text{cube} \\ \hline \text{cube} & \text{cube} & \text{cube} & \text{cube} \\ \hline \text{cube} & \text{cube} & \text{cube} & \text{cube} \\ \hline \text{cube} & \text{cube} & \text{cube} & \text{cube} \\ \hline \end{array} \\ N=27 \end{array}$$

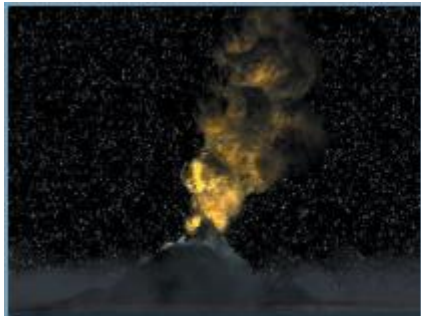
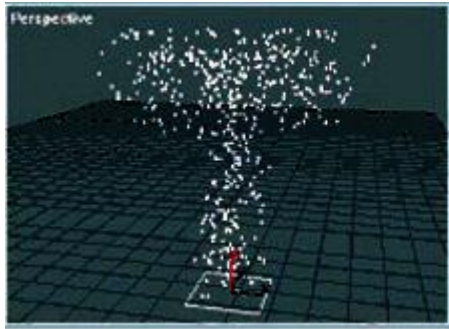
프랙탈 기하학의 차원

- 프랙탈 기하학의 차원의 예: Koch 의 Snowflake
 $D = \frac{\ln 4}{\ln 3} = 1.26...$ ➔ 즉, 2차원은 아니지만 1차원도 아닌 비정수 차원



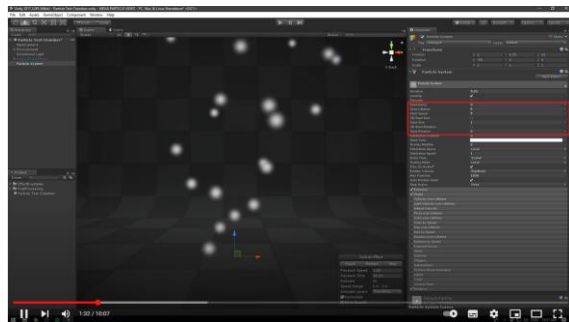
입자 시스템 (Particle system)

- 입자 시스템이란
 - 많은 개별적인 입자들의 움직임의 집합
 - 입자는 개별적인 특성을 갖고 있으며 입자들 서로간에 독립적으로 행동
 - 입자시스템의 각 입자들은 공통적인 특성을 공유하고 있기에 공통된 효과를 나타낸다.
 - 불꽃, 폭발, 연기, 액체, 눈, 비 먼지와 같은 자연현상을 효과적으로 표현



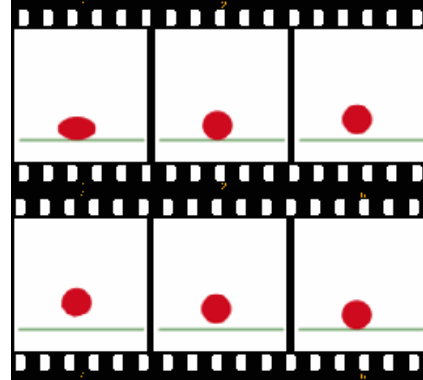
입자 시스템 (Particle system)

- 입자 시스템의 특성
 - 하나의 입자는 수명, 크기, 무게, 모양, 색상, 소유자 등과 같은 개별적 특성을 가짐
 - 입자 시스템은 이러한 입자들을 총괄
 - 입자들의 속성과 공통적인 특성을 저장, 관리
 - 입자들의 생성과 소멸, 움직임을 총괄
 - 입자의 모양
 - 단순한 점, 선 혹은 다각형 등의 Primitive Object
 - Texture가 적용된 빌보드(Billboard) 객체
 - 모델링 된 3차원 객체와 같은 복잡한 물체, 예) 철새들의 비행
 - 입자 시스템은 많은 메모리와 계산시간을 요하므로 단순화 된 입자를 많이 사용
 - 각 입자는 자신의 속성과 수명에 관한 정보를 데이터 구조 형태로 가짐
 - 때로는, 물리적 법칙과 같은 특정한 규칙에 제한을 받음
 - 입자는 하나의 객체(Object)로서 객체지향 프로그램 개념에 잘 부합



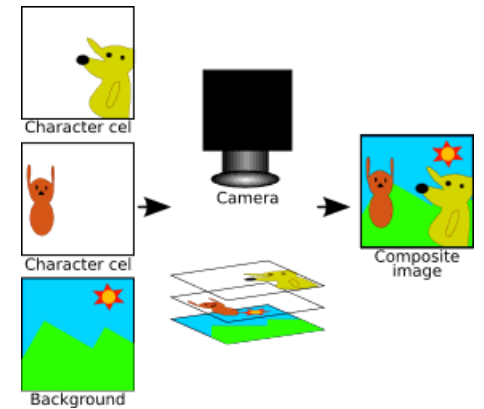
애니메이션 (Animation)

- 용어의 원래 의미: 생명이 없는 사물에 영혼이나 정신을 부여하는 행위
 - 일련의 정지된 그림을 빠른 속도로 재생
 - 사람 눈의 생리적 특성인 잔상현상을 이용: 약 1/16초
 - 프레임(Frame) 사이의 간격이 잔상의 지속 시간을 초과하지 않아야 함
 - 플리커(Flickering) 현상: 프레임 사이의 간격이 잔상 지속시간인 1/16초를 초과하면 화면이 끊어져 보인다.
 - 영화에서는 초당 24 프레임, NTSC 비디오 방식에서는 초당 30 프레임



애니메이션의 종류

- 플립북 애니메이션(Flip-book Animation)
 - 전통적인 애니메이션 기법으로 프레임(Frame-based) 애니메이션이라고도 함
 - 모든 프레임을 일일이 그려서 저장
 - 파일의 크기가 크기 때문에 인터넷과 같은 환경에서 데이터 전송 시간 많이 걸림
 - [예제\)](#)
- 셀 애니메이션(Cel Animation)
 - 1910년 존 랜돌프가 개발
 - "CEL"이란 단어는 투명한 종이를 뜻하는 Celluloid를 의미
 - 2차원 애니메이션을 제작할 때 자주 사용하는 기법
예) 디즈니 애니메이션 만화영화
 - 애니메이션을 만들기 위해서는 하나의 배경 셀과 여러 장의 전경 셀이 필요
 - 1990년 이후 컴퓨터를 이용하여 제작



애니메이션의 종류

- 키프레임 애니메이션(Key-Frame Animation)
 - 중요한 프레임(키프레임) → 그사이 장면(In-between Frames)들을 삽입
 - 트위닝(Tweening): 컴퓨터를 이용한 애니메이션에서도 이 개념을 그대로 적용
 - 애니메이터가 특정 프레임들을 키 프레임으로 지정하여 그리고, 컴퓨터가 중간 프레임들을 보간법(Interpolation)으로 자동 생성
 - 선형 보간법(Linear Interpolation) 적용



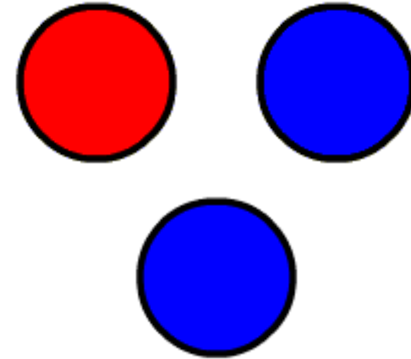
시작 키프레임



끝 키프레임



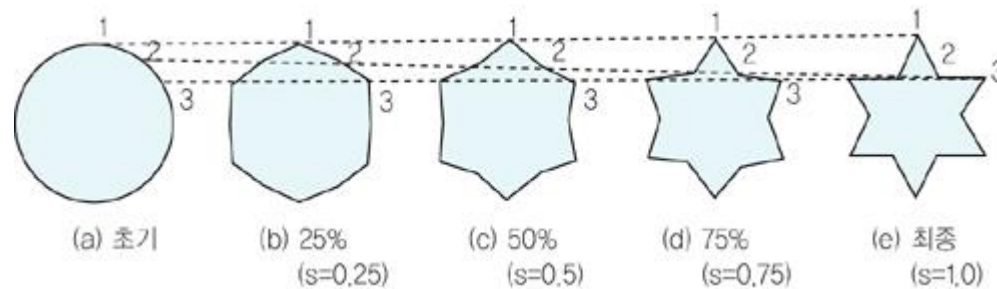
최종 키프레임



색상변환 애니메이션

애니메이션의 종류

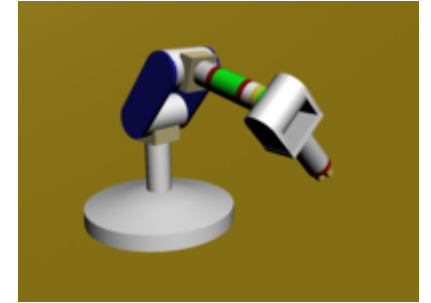
- 모핑 (Morphing)
 - 모핑은 "변형"을 의미하는 "Metamorphosis" 에서 유래
 - 2개의 서로 다른 이미지나 3차원 모델 사이의 변화하는 과정을 표현
 - 모핑은 초기 형상과 최종 형상이 완전히 다른 물체인 경우에 적용
 - 두 프레임 간에 대응점들을 지정해 주고 중간 프레임들은 컴퓨터가 자동 생성
 - 2차원 그림에서는 물론 3차원 그림에도 적용
 - [모핑 예제](#)



캐릭터 애니메이션

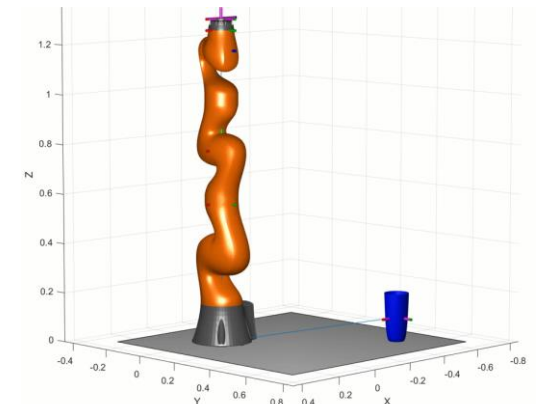
- 운동학(Forward Kinematics)

- 세그먼트(Segment)들이 계층구조에 의해 연결된 모델(Hierarchical model)에서 움직임의 시작점인 루트 관절로부터 각 관절의 각도와 위치를 누진적으로 계산
- 기계적인 물체(예, 로봇)의 비교적 단순한 움직임이라면 객체 세그먼트의 각도와 위치를 일일이 수동적으로 조절 가능
- 사람이나 동물의 관절 움직임을 애니메이션 하고자 할 때 운동학을 이용하면 대단히 번거로운 작업이 될 뿐만 아니라 직관적이지 않음



- 역운동학(Inverse Kinematics)

- 계층구조(Hierarchy) 제일 말단 부분의 객체(Effector)의 위치를 지정
- 나머지 중간 관절들의 위치와 각도를 역으로 계산해 나가는 방식
- 역운동학은 캐릭터 애니메이션에서 움직임을 효과적으로 표현



이번 주에는

- 절차적 그래픽스
 - 프렉탈 생성
- 애니메이션