

프로세싱 중간과제

-내가 생각하는 숲-

201366229 유지연

넓다

외로움

슬

낭만적

고요함

외로움



forest151030

BuckAnimation

Drawin

```
public float mx=0, my=0;
public int direction=1;
public int buckframe=0, hideframe=0;
public void mousePressed(){
    mx = mouseX;
    my = mouseY;
}
```

```
void clickBuck(float x, float y){
    if(direction == 1){
```

약 12장의 프레임을 만들어
움직임을 주었다.

마우스로 사슴의 얼굴을 클릭 시
애니메이션이 시작된다.

Direction == 1 인 경우와
-1인 경우로 나누어
고개를 숙이고 다시 들어올리는 움직임을
제어했다.

넓다

고요함

얼은 호수

베지어 곡선을 이용하여
프로세싱 내에서 호수를 만들었다.

분위기를 살리기 위해
하늘에도 그라데이션을 주어,
윗부분은 더 캄캄하고 어두운 색을
아랫부분은 조금더 밝은 색을 입혔다.

```
//Sky*****
for(int i=0; i<600; i++){
  //stroke(#001E2C,255-i*0.5);
  stroke(#000203,255-i*0.5);
  stroke(#0C080D,255-i*0.4);

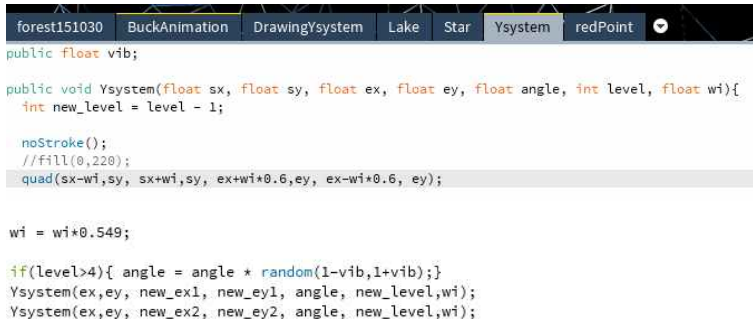
  line(0,i, 1024,i);
}
```

forest151030	BuckAnimation	DrawingYsystem	Lake	Star
--------------	---------------	----------------	------	------

```
void lake(){

  pushMatrix();
  translate(-530,-550);
  beginShape();
  //stroke(0);
  vertex(60, 620); //ver
  bezierVertex(60, 590, 30, 570, 70, 520); //a,a,ver
  bezierVertex(75, 510, 160, 450, 190, 450); //a,a,ver
  bezierVertex(250, 440, 320, 415, 350, 415); //a,a,ver
  bezierVertex(360, 415, 400, 415, 570, 415); //a,a,ver
  bezierVertex(660, 415, 770, 415, 770, 415); //a,a,ver
  bezierVertex(770, 415, 850, 415, 920, 470); //a,a,ver
  bezierVertex(935, 480, 986, 510, 996, 515); //a,a,ver
  bezierVertex(1000, 520, 1050, 515, 1024, 700); //a,a,ver
  bezierVertex(1000, 700, 500, 700, 500,700); //a,a,ver
  //bezierVertex(100, 700, 150, 700, 60,700); //a,a,ver
  endShape(CLOSE);
  popMatrix();
}
```

YSYSTEM



```
forest151030 BuckAnimation DrawingYsystem Lake Star Ysystem redPoint ▼  
public float vib;  
  
public void Ysystem(float sx, float sy, float ex, float ey, float angle, int level, float wi){  
    int new_level = level - 1;  
  
    noStroke();  
    //fill(0,220);  
    quad(sx-wi,sy, sx+wi,sy, ex+wi*0.6,ey, ex-wi*0.6, ey);  
  
    wi = wi*0.549;  
  
    if(level>4){ angle = angle * random(1-vib,1+vib);}  
    Ysystem(ex,ey, new_ex1, new_ey1, angle, new_level,wi);  
    Ysystem(ex,ey, new_ex2, new_ey2, angle, new_level,wi);  
}
```

수업시간에 이용했던 Ysystem 소스를 수정하여 이용하였다.
Line은 quad 함수로 바꾸고 나무 굵기를 조절하는 wi(width)변수를 추가하였다.

Vib(진동)을 주는 변수를 추가하여 나뭇가지가 조금씩 흔들리도록 하였다.
단, 자연스럽게 흔들리길 위하여 vib = 0.001; 가량의 아주 적은 수치를 주었다.

YSYSTEM 적용

forest151030 BuckAnimation DrawingYsystem Lake St

```
void drawYsystem_layer1(){  
    for(int i=0; i<8; i++){  
        //L4  
        //fill(#d0d0d0,180-i*20);  
        fill(0,200-i*20);  
        vib = 0.005;  
        Ysystem(230-i*1.2,470,230-i*1.2,330,15,6,10); //  
        Ysystem(230+i*1.2,470,230+i*1.2,330,15,6,10); //  
  
        //L5  
        //fill(#d0d0d0,180-i*20);  
        fill(0,200-i*20);  
        vib = 0.005;  
        Ysystem(250-i*1.1,440,255-i*1.1,320,15,6,10); //  
        Ysystem(250+i*1.2,440,255+i*1.2,320,15,6,10); //  
        /*  
        //L4_shadow  
        fill(0,15);
```

가독성을 위해 DrawingYsystem탭에
나무를 그리는 소스를 정리하였다.

그림의 미를 위해 연하게 겹쳐지도록 하는 효과를 주었다.
(변수 i를 활용하여 Ysystem 반복사용, opacit조절)



낭만적



```
forest151030  BuckAnimation  DrawingYsystem  Lake  Star

void star(){
  int px, py;

  px = (int)random(0,1024);
  py = (int)random(0,400);

  for(int i=1; i<5; i++){
    noStroke();
    fill(#DBFFEF,130-i*30);
    ellipse(px, py, i*random(0,1.5),i*random(0,1.5));
  }
}


imageMode(CENTER);
blend(ton1,0, 0, 2000, 1000, 0, 0, 1024, 600, DODGE); /
blend(ton2,0, 0, 2000, 1000, 0, 0, 1024, 600, SCREEN);
```

낭만적인 분위기 연출을 위해 별을 뿌렸다.

그리고 두 가지의 톤을 블렌딩 시켰다.

Ton1과 ton2는 좌측의 이미지 소스를 포토샵으로 수정하여 이용하였다.





중간발표후 수정사항

1. 호수

호수가 얼어있는 것처럼 보이도록 색상을 좀더 하얗게 올렸습니다.



2. 사슴

사슴의 마지막프레임을 물을 마시는 모습이 아닌 얼은 호수에 얼굴을 비춰보는 모습으로 변경하였습니다.



'녹은 호수와 물마시는 사슴' 이 아닌
'얼은 호수와 얼굴을 비춰보는 사슴'
을 고집한 것은 이유가 있다.

내가 생각하는 숲은 '외로움'이 있는 장소이기
때문이다.

얼어붙은 호수위에서
비춰지는 자기 자신을 외로이 바라보는
사슴을 넣고자 했다.

3. 밤하늘의 별

기존의 별은 위치가 고정되었지 않았으며 지정된 범위내에서 움직이며 깜박였습니다.

좀 더 밤하늘처럼 디테일을 살리기 위해

밤하늘의 별들의 위치를 고정시켜 주었으며 카시오페이아, 안드로메다, 트라이앵글럼 자리의 별자리는 제자리에서 빛나도록 하였습니다. (소스 첨부)

[기존 소스 코드]

```
void star(){
    int px, py;

    px = (int)random(0,1024);
    py = (int)random(0,400);

    for(int i=1; i<5; i++){
        noStroke();
        fill(#DBFFEF,130-i*30);
        ellipse(px, py, i*random(0,1.5),i*random(0,1.5));
    }
}
```

[변경 후 소스 코드]

```
color basic = #DBFFEF;
int [] px = new int[500];
int [] py = new int[500];

public int [] seedFrame = new int [2];

//draw the point. == like ellipse.
void star(float px, float py, float lineLength, float maxSize, color col){
    for(int i=1; i<6; i++){
        fill(col,130-i*20);
        if(i>2){
            stroke(col,50-i*8);
            line(px+random(-lineLength,0), py, px+random(0,lineLength), py);
            line(px, py+random(-lineLength,0), px, py+random(0,lineLength));
        }
        noStroke();
        ellipse(px, py, i*random(0,maxSize),i*random(0,maxSize));
    }
}
```

fill 값이 일정한 ellipse 한개로는 그저 원일 뿐, 별과 같은 느낌이 들지 않는다.

때문에 실제 별과 같은 디테일을 위해 for문을 통해 line과 ellipse를 반복적으로 호출하였으며, 빛나는 효과를 위해 여러 변수를 사용하였다. 함수 호출시, 실인자값을 조정하여 별의 디테일 수정이 용이하다.

```
void etc(int seedIndex, int range, int num){
    seedFrame[seedIndex]++;
    if(seedFrame[seedIndex]>100){
        seedFrame[seedIndex] = 0;
    }
    randomSeed(0);
    for(int i=0; i<80+num; i++){
        star(random(0,1024),random(0,range), 2, 0.4, #BAACC6);
    }
    randomSeed(seedFrame[seedIndex]); //random initialize!
}
```

- 밤하늘의 별들을 뿌리는 구문이다.

수많은 별들의 위치를 고정시키기 위하여 seed값을 주었다.

- 일부 별자리들 (카시오페아,안드로메다,트라이앵글럼)은 더 돋보이도록 제자리에서 반짝이고 싶었다.

그래서 별을 80개 가량 뿌린뒤,

seed값에 규칙적으로 변화하는 수 (seedFrame[i])를 넣어, 위의 시드값을 제거하는 효과를 주었다.

설명:

별 한개를 그리는 star() 함수 내부에 random()함수가 쓰인 이유는 별을 반짝이게 하기 위함이었다.

시드가 0으로 고정되면 별들이 반짝이지 않기 때문에 시드를 반드시 초기화 시켜야 한다.

[변경 후 소스 코드]

```
void cassiopeia(){
    star(-10,-81, 5, 1.3, #FFE9CE); //caf
    star(45,-47, 5.5, 1.5, basic);
    star(0,0, 6, 1.6, #CBE0FF); //gamma
    star(20,45,4, 1.2, basic);
    star(-29,87,4, 1.15, basic);
}

void andromeda(){
    star(0,100, 5, 1.3, basic);
    star(64,0, 5, 1.3, basic);
    star(30,-30, 5, 1.1, basic);
    star(0,-40, 3, 0.7, basic);
    star(108,-70, 4, 0.8, basic);
    star(112,-145, 5.5, 1.5, basic);
    star(254,-163, 4, 1, basic);
}

void triangulum(){
    star(0,0, 4, 1.15, basic);
    star(60,-40, 4, 1.05, basic);
    star(20,20, 4, 0.6, basic);
}
```

카시오페이아, 안드로메다,
트라이앵글럼 별자리를 만들었다.
스텔라리움이라는 프로그램을 이용하여
실제 북쪽지역에서 지평선 위로 보이는
밤하늘이 되도록 그렸다.
star()에 쓰인 인자는
(위치x, 위치y, line길이, star크기, 색)
이다.
:line길이는 반짝이는 효과에쓰임.

```
//drawing a night sky
pushMatrix();
translate(400,200);
cassiopeia();
translate(200,0);
andromeda();
translate(80,110);
triangulum();
popMatrix();
etc(0,380,0);
etc(1,240,80);
```

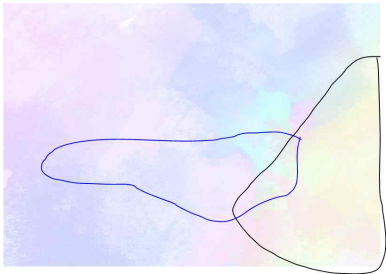
draw함수 내부의 코드는 이와
같이 간결하다.
적절한 변수를 넣어 디테일을
조정한다.

4. 피드백 받았으나 수정하지 못한 사항.

밤하늘에 쓰인 ton(이하 톤, 텍스처)와 호수에 쓰인 ton이 같은데,
하늘과 호수에서 같은 ton이 쓰이고 있어서 이를 구분하면 좋겠다는 피드백을 받았었다.

하지만 아래의 톤을 포토샵으로 수정하는 것이 어려웠다.
파랗게 표시한 부분에서 밤하늘과 호수에 톤이 이어졌기에 이부분을 없애야 했지만,
검은색으로 표시한 부분은 아기 사슴에 초록색 빛을 비춰주는 효과여서 꼭 남기고 싶었다.

한쪽은 살리고 한쪽은 없애는 작업에 실패하여 톤은 수정하지 못하였다.



5. 결과.

아주 큰 차이는 없으나 변경된 디테일 부분에 있어서 결과물이 마음에 든다.

시간을 더 투자하여 원하는 결과를 내기위해 코딩을 하였고, 원하는 결과에 거의 가까워 질 수 있었다. 값진 경험이었다.

