

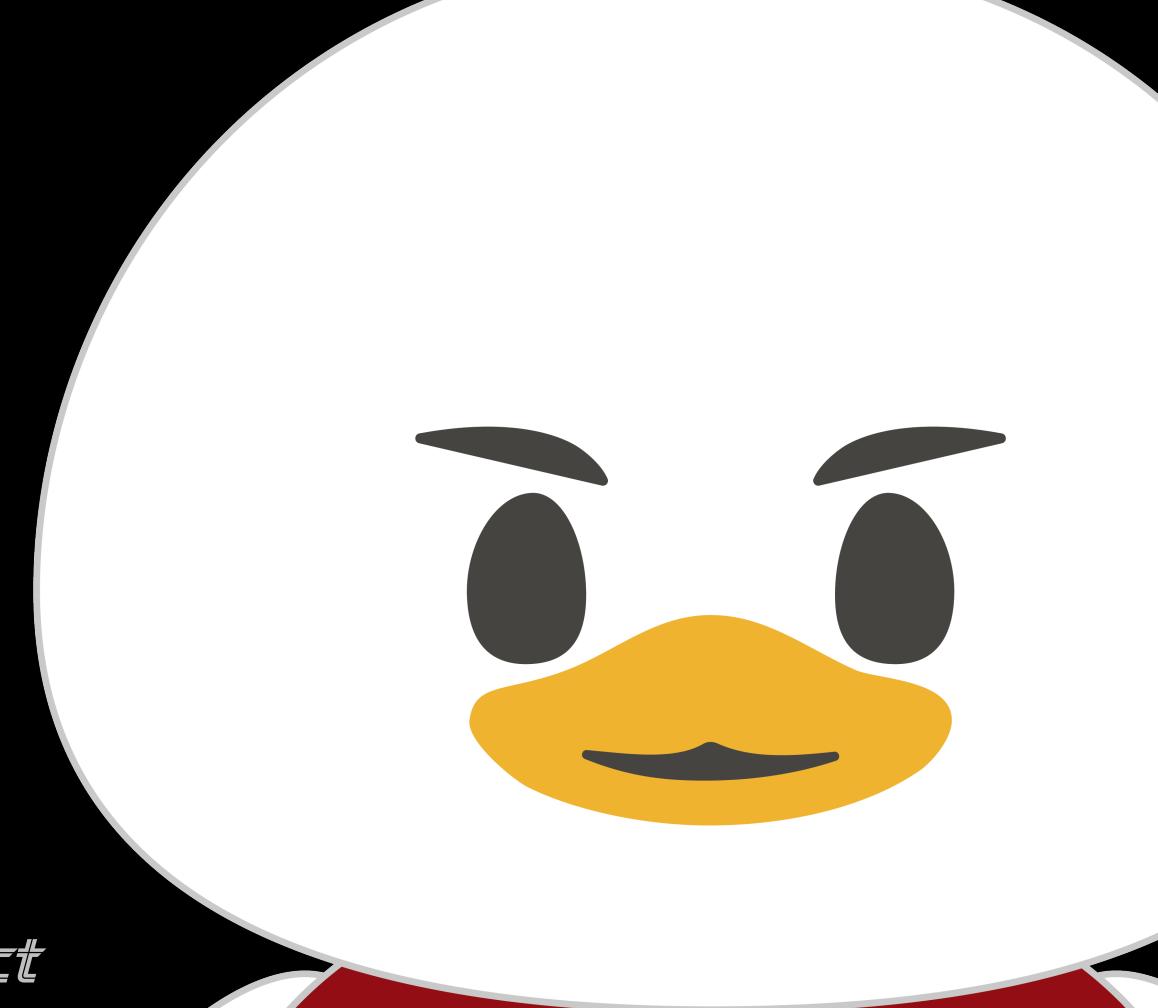
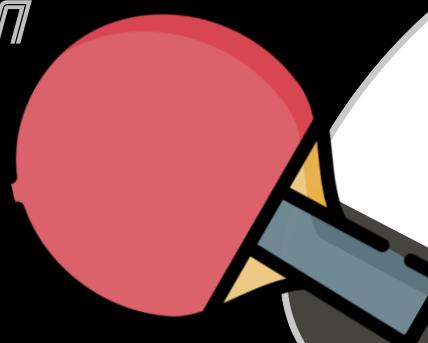
PINGPONG with ALLOS

—

Fall 2023 Semester

[AAT3008] Creative Algorithms - Final Project

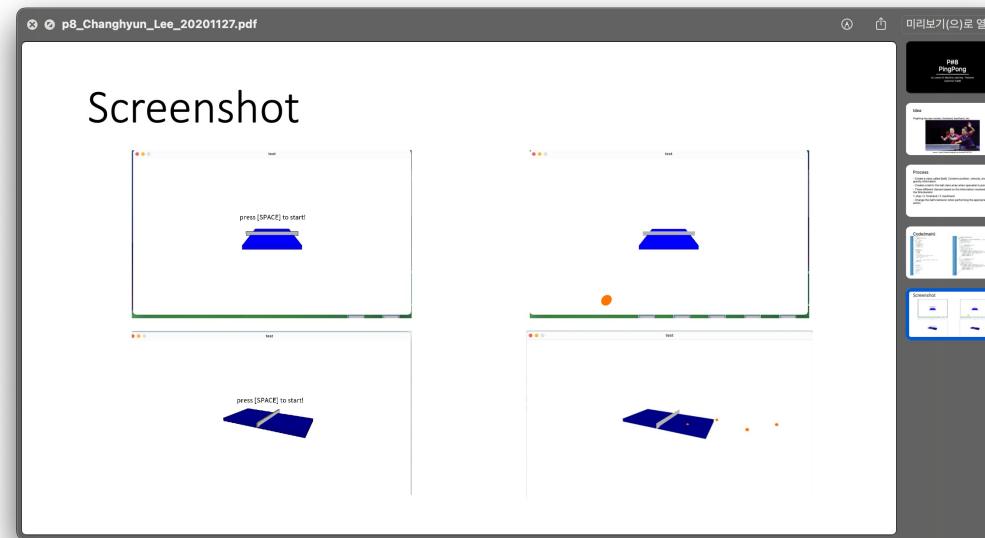
20201127 LEE CHANG HYUN



: SG

Motivation

In project 8,
I made basic of Ping-pong. So, I want to develop it.
Ping-pong moves are good to categorize as 'Wekinator'



Summary

This is a 3D ping pong game named "PINGPONG with ALLOS." It utilizes PeasyCam and Minim libraries for camera control and audio playback.

The game involves hitting a ball on a ping pong table, featuring coaching feedback, scoring, and a timer.

It interacts with a machine learning model through OSC messages, adjusting ball behavior.

Visual elements include a 3D ping pong table, timer display, coaching messages, and characters.

Success/failure screens prompt user restart or exit.

Process(just play)

- Create a class called [ball]. Contains position, velocity, and gravity information
- Creates a ball in the ball class array when spacebar is pressed.
- Three different classes based on the information received from the Wreckerator
 1. stop / 2. forehand / 3. backhand
- Change the ball's behavior when performing the appropriate action

Process(visual)

- Changes every time the character 'allos' throws a ball
- Various visual changes based on score situation
(facial expression of 'sogangi', color of 'stack'
character, etc.)
- Different endings depending on the final outcome

How the program works [code]

```
64 void setup() {
65   size(960, 540, P3D);
66
67 // load all sources-----
68 myFont = createFont("Giants-Inline.otf", 32);
69   textFont(myFont);
70   minim = new Minim(this);
71   bgm = minim.loadFile("bgm.mp3");
72   ping = minim.loadFile("ping.mp3");
73   pong = minim.loadFile("pong.mp3");
74   bgm.loop();
75   for(int i=0;i<4;i++){
76     allos[i] = loadImage("al"+i+".png");
77     allos[i].resize(1000,1500);
78   }
79   for(int i=0;i<4;i++){
80     sogangi[i] = loadImage("so"+i+".png");
81     sogangi[i].resize(1000,1500);
82   }
83   logo = loadImage("logo.png");
84   logo.resize(238,342);
85 //-----
86
87 // set the frame rate
88 frameRate(60);
89
90 oscP5 = new OscP5(this, 12000);
91 dest = new NetAddress("127.0.0.1", 6448);
92
93 // PeasyCam 초기화
94 cam = new PeasyCam(this, width / 2);
95 cam.setRotations(PI / 15, 0, 0); // X축을 기준으로 시점 설정
96
97 balls = new ArrayList<Ball>(); // ArrayList 초기화
98
99 }
```

```
101 void draw() {
102   background(255);
103
104   draw_sogangi();
105   draw_object();
106
107   if (gameStarted) { // 게임이 시작되면 모든 Ball을 업데이트하고 시작
108     draw_timer();
109     for (Ball ball : balls) {
110       ball.update();
111       ball.display();
112     }
113     motion();
114     score();
115     coaching();
116     draw_stacks();
117     draw_allos();
118     if (remainingTime == 0) {
119       if (score > 0) {
120         success();
121       } else {
122         fail();
123       }
124     } else { // 게임이 시작되지 않았으면 메시지 표시
125       displayStartMessage();
126     }
127   }
128 }
```

What is the ball class?

```
136 Ball(float speedX) {
137     this.x = 0;
138     this.y = -30;
139     this.z = -150;
140     this.speedX = speedX;
141     this.speedY = 0;
142     this.speedZ = 10;
143     this.gravity = 0.1;
144 }
145
146 void update() {
147     speedY += gravity;
148     x += speedX;
149     y += speedY;
150     z += speedZ;
151
152     // 탁구대와의 충돌 체크
153     if (y > -5 && y < 5 && z > -137 && z < 137 && x > -76 && x < 76) {
154         allos_img = 0;
155         // 충돌 발생 시 y 속도를 반대로 변경
156         speedY = -speedY;
157         if(z > 0 && z < 137){
158
159             ping.rewind();
160             ping.play();
161             pongs = 0;
162             coach = 3;
163
164         } else { // 잘 받아쳐서 상대 탁구대에 공이 닿았을 때
165
166             pong.rewind();
167             pong.play();
168             if(stacks < 10){ // 스택은 최대 10개까지
169                 stacks += 1;
170                 stacks_bad = 0;
171             }
172             pongs = 1;
173             if(stacks > 5){
174                 score += 3;
175             }else if(stacks > 3){
176
177             score += 2;
178         }else{
179             score += 1;
180         }
181         coach = 1;
182     }
183
184
185     // 알로스 리시브
186     if (z < -151 && y < 0 && y > -50 && x > -76 && x < 76) {
187         if(pongs == 1){
188             x = 0;
189             y = -30;
190             z = -150;
191             speedX = (random(2) > 1) ? 1 : -1;
192             speedY = 0;
193             speedZ = 10;
194             allos_img = 1;
195         } else {
196             coach = 2;
197         }
198
199         if (y > 76) {
200             allos_img = 1;
201             x = 0;
202             y = -30;
203             z = -150;
204             speedX = (random(2) > 1) ? 1 : -1;
205             speedY = 0;
206             speedZ = 10;
207             stacks = 0;
208             pongs = 0;
209             stacks_bad +=1;
210         }
211     }
212
213
214 void display() {
215     pushMatrix(); // 현재 변환 상태 저장
216
217     translate(x, y, z);
218
219     fill(255, 127, 0);
220     noStroke();
221     sphere(4); // 지름이 15인 3D 구
222
223     popMatrix(); // 이전 변환 상태로 복원
224 }
225
226
227 void displayStartMessage() {
228     cam.beginHUD(); // HUD 모드 시작
229     fill(255,255,255,160);
230     rect(0,0,960,540);
231     textAlign(CENTER, CENTER);
232     fill(0);
233     textSize(50);
234     text("PINGPONG \nwithALLOS",width / 2, 200);
235     textSize(24);
236     text("press [SPACE] to start!", width / 2, 400);
237     textSize(15);
238     text("It can be moved around the screen with the i");
239     text("Fall 2023 Semester [AAT3008] Creative Algor
240
241 }
```

How to end the game[Code]

```
349 void fail() {
350     cam.beginHUD();
351     fill(255,255,255,160);
352     rect(0,0,960,540);
353     image(allos[3],-100,-250, 600,900);
354     fill(0);
355     textSize(80);
356     textAlign(CENTER, CENTER);
357     text("FAIL", width*2/3, height / 3);
358     textSize(24);
359     text("The timer has exceeded 100 seconds!\nPress [R] to restart or [Q] to exit.", width*2/3, height*2/3);
360     cam.endHUD();
361
362     // 사용자 입력 확인
363     if (keyPressed) {
364         if (key == 'R' || key == 'r') {
365             resetGame();
366             println("reset");
367         } else if (key == 'Q' || key == 'q') {
368             exit();
369         }
370     }
371 }
372
373 void resetGame() {
374     score = 0;
375     balls.clear(); // 모든 공 제거
376     gameStarted = false;
377     timerStartTime = millis(); // 타이머 재설정
378 }
```

How to implement interactivity(code)

```
396 void oscEvent(OscMessage msg) {
397     if (msg.checkAddrPattern("/wek/outputs"))
398     {
399         msg.print();
400         type = msg.get(0).floatValue();
401         currentTime = millis();
402     } else {
403         //msg.print();
404     }
405 }
406
407 void motion(){
408     if (type == 1) {
409         // 클래스 0에 대한 동작 수행
410     } else if (type == 2 && currentTime - lastTime > 800) {
411         // 클래스 1에 대한 동작 수행
412         // 마지막으로 생성된 공의 조작
413         if (balls.size() > 0) {
414             Ball lastBall = balls.get(balls.size() - 1);
415             if (lastBall.speedX > 0 && lastBall.speedY < 0) {
416                 // speedX가 양수이고 speedY가 음수인 경우
417                 lastBall.speedX *= -1;
418                 lastBall.speedZ *= -1;
419             }
420         }
421         lastTime = currentTime;
422     } else if (type == 3 && currentTime - lastTime > 800) {
423         // 클래스 10에 대한 동작 수행
424         // 마지막으로 생성된 공의 조작
425         if (balls.size() > 0) {
426             Ball lastBall = balls.get(balls.size() - 1);
427             if (lastBall.speedX < 0 && lastBall.speedY < 0) {
428                 // speedX가 음수이고 speedY가 음수인 경우
429                 lastBall.speedX *= -1;
430                 lastBall.speedZ *= -1;
431             }
432         }
433         lastTime = currentTime;
434     }
435 }
```

Other various function(code)

```
243 void keyPressed() {
244     if (key == ' ') {
245         // 스페이스바를 누르면 새로운 공 추가
246         int direction = (balls.size() % 2 == 0) ? 1 : -1;
247         Ball newBall = new Ball(direction);
248         balls.add(newBall);
249         gameStarted = true;
250
251         timerStartTime = millis();
252     } else if (key == ENTER) {
253         // Enter 키를 누르면 초기 각도로 카메라를 설정
254         cam.setRotations(PI / 15, 0, 0);
255     }
256 }
257
258 void score(){
259     cam.beginHUD(); // HUD 모드 시작
260     fill(0);
261     textSize(24);
262     textAlign(CENTER,CENTER);
263     text("SCORE",width/2,20);
264     textSize(50);
265     text(score,width/2,70);
266     cam.endHUD(); // HUD 모드 종료
267 }
268
269 void coaching(){
270     cam.beginHUD();
271     if(coach == 1){
272         text("Good!",width*5/8,height*2/5);
273     } else if (coach == 2){
274         text("FAST!",width*5/8,height*2/5);
275     } else if (coach == 3){
276         text("SLOW!",width*5/8,height*2/5);
277     } else{
278     }
279     cam.endHUD();
280 }
281
283 void draw_stacks(){
284     cam.beginHUD(); // HUD 모드 시작
285     fill(0);
286     textSize(24);
287     textAlign(CENTER,CENTER);
288     text("STACKS",width/4,20);
289     textSize(50);
290     if(stacks > 7){
291         fill(255,0,0);
292     } else if (stacks > 5){
293         fill(200,0,0);
294     } else if (stacks > 3){
295         fill(150,0,0);
296     }
297     text(stacks,width/4,70);
298     cam.endHUD(); // HUD 모드 종료
299 }
300
301 void draw_allos(){
302     translate(10,-20,-150);
303     image(allos[allos_img],-25,-37.5,50,75);
304     translate(-10,20,150);
305 }
306
307 void draw_timer(){
308
309     elapsedTime = millis() - timerStartTime;
310     remainingTime = max(0, timerDuration - int(elapsedTime / 1000));
311
312     cam.beginHUD();
313     fill(0);
314     textSize(24);
315     textAlign(CENTER, CENTER);
316     text("TIMER", width*3/4,20);
317     textSize(50);
318     if(remainingTime < 11){
319         fill(255,0,0);
320     }
321     text(nf(remainingTime, 2) + "s", width*3/4, 70);
322     cam.endHUD();
323 }
```

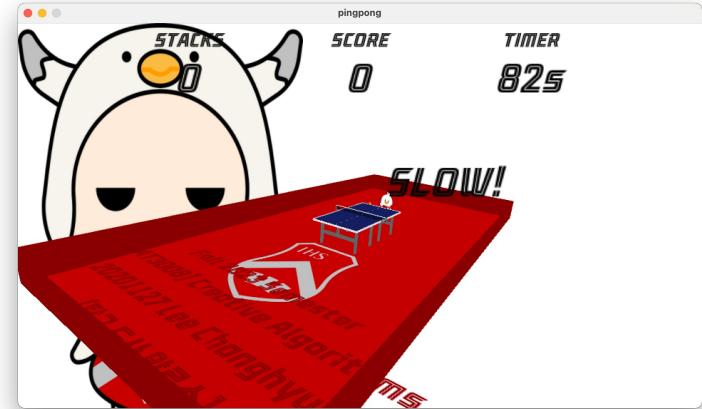
Screenshot



Start



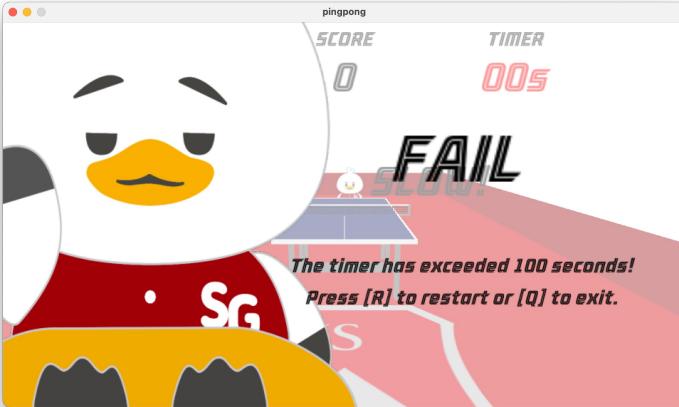
Composition1(normal)



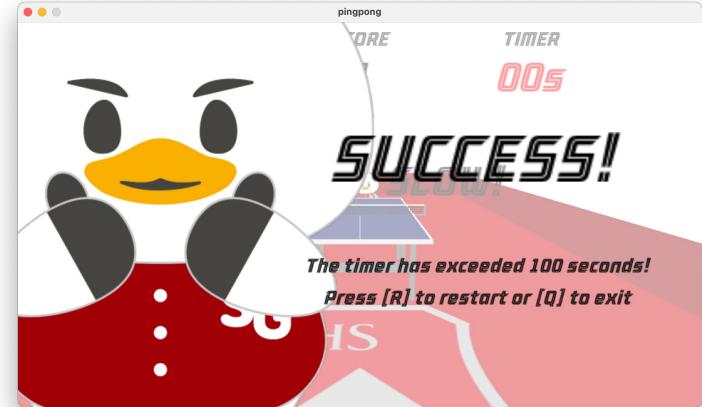
Composition2



Composition3



Ending(fail)



Ending(success)

Link & Source

VIDEO :

<https://youtu.be/VcDjfYEumgY>

CODE :

<https://github.com/lee-chh/Pingpong>

Font : <https://www.giantsclub.com/html/?pcode=1007>

Character : https://www.sogang.ac.kr/intro/symbol/s_character.html

Symbol : https://www.sogang.ac.kr/intro/symbol/s_symbol.html