**<Assign#1. Draw Initial> Code Report**

202126894 소프트웨어학과 박승연

**#1**. **Make a model** using all primitives supported by WebGL

: Park Seung Yeon = PSY라는 이니셜에는 직선의 요소와 곡선의 요소가 섞여 있기 때문에 우선 이 두 가지 요소를 분리하여 vertex에 필요한 좌표들을 정의하고자 했습니다. (어디에 사용되는 vertex인지 구분하기 쉽도록 이니셜과 primitive 형태를 포함하여 이름을 선언했습니다.) 텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진

자동 생성된 설명

(P: line 1개, semicircle 0개, S: line 0개, semicircle 2개, Y: line 3개, semicircle 0개)

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

dotVertices는 모든 primitives를 사용한다는 조건을 충족하기 위해 이니셜 P-S와 S-Y 사이에 point를 추가하는 목적으로 정의되었습니다.

텍스트, 스크린샷, 소프트웨어, 디스플레이이(가) 표시된 사진

자동 생성된 설명

P와 S를 그리는 데에 필요한 semicircle(반구)은 TRIANGLE\_STRIP으로 그릴 수 있는데, 극좌표계와 같이 coordinate을 *반지름의 길이\**와 *각\*\**으로 정의하여 반구의 원주를 따라 (x, y) 좌표들을 삼각함수로 쭉 찍어내 Float32Array로 정리하면 STRIP 작성에 필요한 vertices 정보가 수집됩니다. 이 때 origin에 대한 정보는 덧셈 또는 뺄셈으로, 전체 크기는 원본의 1/n로 축소하여 저장합니다.

이와 같은 반구가 총 3개 필요하므로 circleVertices\_P, \_S#1, \_S#2를 동일한 방법으로 생성합니다. (vert1 또는 vert2 각각이 x 또는 y 좌표이므로 vert1~4 중에 2개씩 concatenate 해줍니다)

\**(=sin과 cos에 곱해지는 값 or 계수)*

\*\**(=j=radian, pi를 180으로 나누면 1 radian이므로 i를 사용해서 180 radian까지 루프로 도달)*

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

vertex를 vec2(x\_coordinate, y\_coordinate)들이 아닌 Float32Array([x1, y1, x2, y2, …]) 형태로 정의한 또 다른 이유는 drawArray를 위해 매번 버퍼를 생성하고, bind하여, data를 넣어주는 과정을 반복하다 보니 공통된 operation을 하나로 묶어서 처리하는 것이 좋겠다고 생각했기 때문입니다.

이와 같은 이유로 정의된 drawA(type, vertices)는 필요한 vertex 정보를 Float32Array type의 vertices와 primitive 종류를 뜻하는 type이 결정되면 drawArray(type, 0, n\*\*\*)를 내부적으로 작성 완료하여 화면에 그려줍니다.

\*\*\*n = primitive type별로 사용하는 vertices 수 = vertices array length / 2

(x와 y 좌표를 vec2( )처럼 set으로 저장하지 않고 array에 나열하여 저장했기 때문입니다)

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

이렇게 하면 render( ) 함수는 위와 같이 정의됩니다.

추후에 animation으로 rotation과 color 변환을 추가하기 위해 theta와 fColor에 대한 uniform function을 drawing 전에 추가해 주었습니다.



viewport는 canvas와 동일하게 설정했습니다. (canvas 크기 : 폭 512, 높이 300)

**#2. Animations and Interactions**

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

애니메이션으로는 수업시간에 알려주신 회전을 추가했습니다. fragment마다 적용할 것이므로 uniform 형태로 theta를 선언하고(HTML), gl의 x와 y position을 수정하여 vertex shader를 작성했습니다.

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

이후 thetaLoc에 shader의 theta를 link하고, js file에서 theta를 새로 선언하여 App 내부에서 바뀌는 theta의 값을 업데이트 하고 위의 render( ) 내부에서 thetaLoc과 App의 theta value를 uniform하여 thetaLoc이 Position을 수정하도록 구성했습니다.

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

rotation과 관련된 interaction은 위와 같이 Start, Stop, Reset, Controls를 옵션으로 하였습니다. Start 버튼을 마우스 클릭하면 startRotation function을 callback하여 direction 정보에 따라 시계 또는 반시계 방향으로 0.1\*delay 간격만큼 회전하게 됩니다. 이후 변화된 theta의 값을 반영하기 위해 render( )를 호출하고, rotate operation을 animationFrame으로 호출하여 rotate( )를 반복하도록 구성했습니다. (animID로 animationFrame의 동작을 관리합니다)

이후 언제든 Stop 버튼이 눌리면 animID가 가리키는 rotate animationFrame을 종료합니다.

Reset은 animation이 시작되기 전, 즉 render( )만 된 상태로 되돌아가는 버튼 interaction을 담당합니다. Stop과 마찬가지로 animation이 수행되지 않는 상태이기 때문에 animID가 가리키는 animation Frame을 종료하고, resetCanvas( )를 불러서 shader에 부가적으로 정의되어 link되는 theta와 fColor를 초기값으로 되돌리고, render( )를 호출하여 저장합니다.

Controls는 direction, delay 값을 조절하여 회전 속도와 방향을 바꿔줍니다. 이 부분은 HTML에서button이 아닌 select로 구성하여 index value에 따라 서로 다른 control을 수행하도록 했습니다. 텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

Start, Stop, 그리고 Reset은 버튼 device로 생성하였고, button을 눌렀을 때의 action을 표현하기 위해 style로 button:hover 부분을 추가했습니다. (클릭하면 버튼 내부 색상 변경)

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

다음으로는 rotation에 이어서 r, g, b 중에 키보드 입력(keydown)이 들어오면 이니셜의 색상을 바꾸는 기능을 추가했습니다. 위의 rotation을 vertex shader에서 수정했다면, 색상은 fragment shader에서 4 dimensional vector인 fColor를 uniform 형태로 선언했습니다. 마찬가지로 fColorLoc에 fColor를 link하고, 키보드 입력이 없는 default 상태의 경우에는 검정색으로 색상을 표시하다가 키보드 입력이 들어왔을 때 그에 따른 색상으로 fColor에 color vector를 저장하도록 switch문을 작성했습니다. 이후 render( )를 호출하면 uniform4fv( )를 통해 fColor의 4 dimensional vector values가 fColorLoc에 전달되어 색상 정보가 업데이트 됩니다.

폰트, 텍스트, 스크린샷, 그래픽이(가) 표시된 사진

자동 생성된 설명

키보드 입력을 통해 색상을 변경할 수 있음을 notify하기 위해 color-change-info를 추가하고, 첫 번째 assignment라는 타이틀까지 추가하여 모든 device을 html이라는 이름의 하나의 device class로 묶어서 위치를 조정해 주고 나면 완성입니다.

텍스트, 스크린샷, 폰트, 메뉴이(가) 표시된 사진

자동 생성된 설명

**<결과물: Window 화면>**

텍스트, 스크린샷, 소프트웨어, 컴퓨터 아이콘이(가) 표시된 사진

자동 생성된 설명