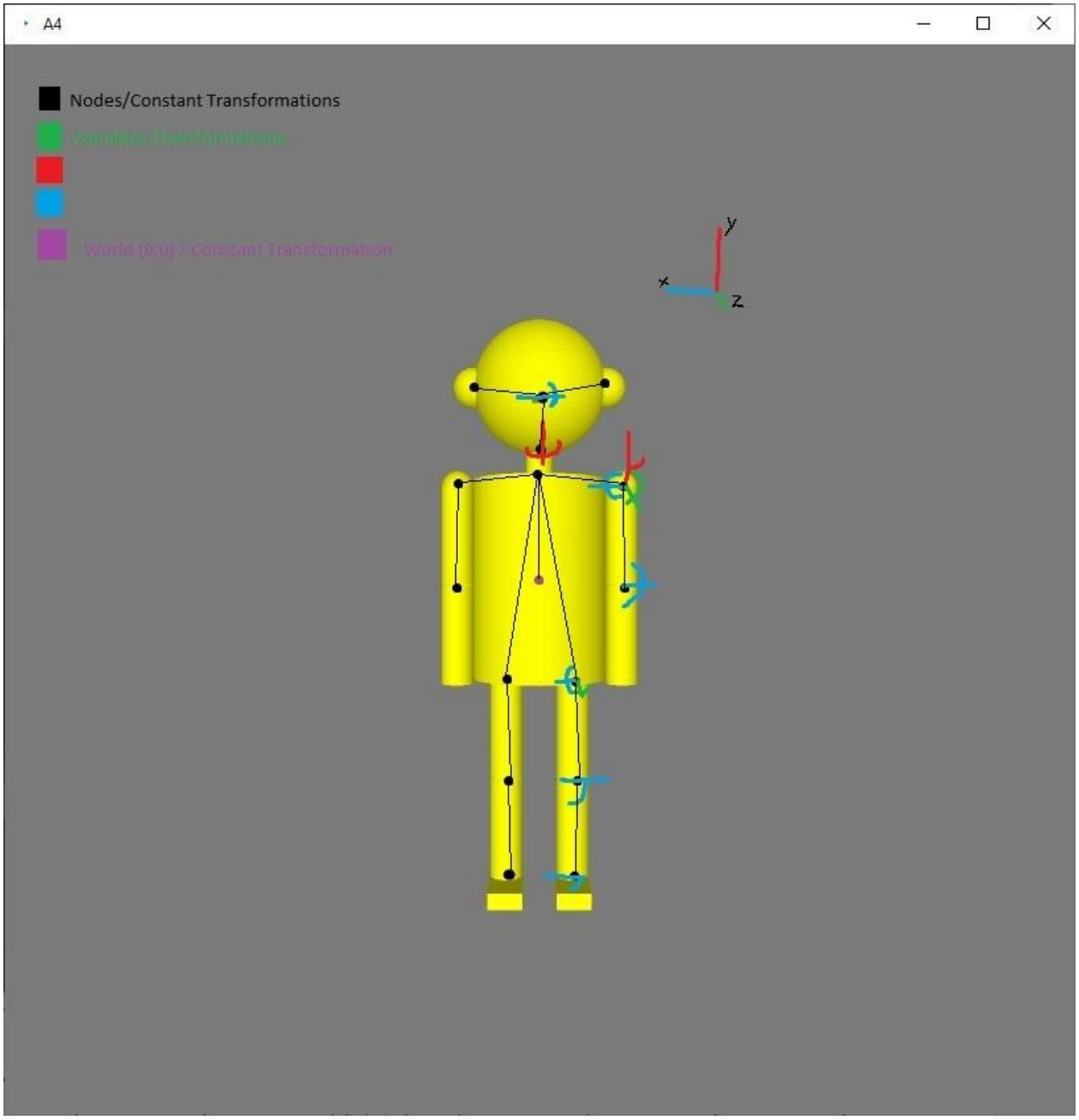
Nicholas Bolen
#200455709
CS408 Assignment 4
15 November 2022

Q1

My extra body parts were simply knees, elbows and shoulders, consisting of spheres. They don't add much for actual "actionable" or interesting body parts, however I feel they add a lot to smoothing out the character and movement, preventing some sharp edges and floating limbs whenever they bend.

Q2

Drawn model with:
Constant, variable, and node transformations



Q3

Note: When some of these rotations are executed, they are negated (in places where angles should be mirrored for left/right limb, or to ensure that the knees/elbows bend the correct direction). This is visible in my code.

Node #	Object	Parent #	Constant	Node	Children
0	torso	-	(0,-75,0)	-	1 6 11 15 19
1	neck	0	-	(0,-95,0)	2
2	head	1	-	(0, -50, 0)	3 4 5
3	ear-l	2	-	(0, -50, 0)	-
4	ear-r	2	-	(0, 50, 0)	-
5	nose	2	-	(0, 15, 50)	-
6	shoulder-l	0	-	(-62, -70, 0)	8
8	upper arm-l	6	-	(0, 0, 0)	9
9	elbow-l	8	-	(0, 75, 0)	10
10	lower arm-l	9	-	(0, 0, 0)	-
11	shoulder-r	0	-	(62, -70, 0)	12
12	upper arm-r	11	-	(0, 0, 0)	13
13	elbow-r	12	-	(0, 75, 0)	14
14	lower arm-r	13	-	(0, 0, 0)	-
15	upper leg-l	0	-	(-25, 75, 0)	16
16	knee-l	15	-	(0, 75, 0)	17
17	lower leg-l	16	-	(0, 0, 0)	18

18	foot-l	17	-	(0, 75, 0)	-
19	upper leg-r	0	-	(25, 75, 0)	20
20	knee-r	19	-	(0, 75, 0)	21
21	lower leg-r	20	-	(0, 0, 0)	22
22	foot-r	21	-	(0, 75, 0)	-

Node #	Object	Variable X (min, max)	Variable Y	Variable Z
0	torso	-	-	-
1	neck	-	-PI/3, PI/3	-
2	head	-PI/6, PI/4	-	-
3	ear	-	-	-
4	nose	-	-	-
5	shoulder	-	-	-
6	upper arm	-PI/4, PI	-PI/2, PI/2	0, PI
7	elbow	0, 5*PI/6	-	-
8	lower arm	-	-	-
9	upper leg	-PI/6, PI/2	-PI/2, 0	0, PI/3
10	knee	-5*PI/6, 0	-	-

11	lower leg	-	-	-
12	foot	-PI/3, 0	-	-

Here are the raw min & max vector arrays I used:

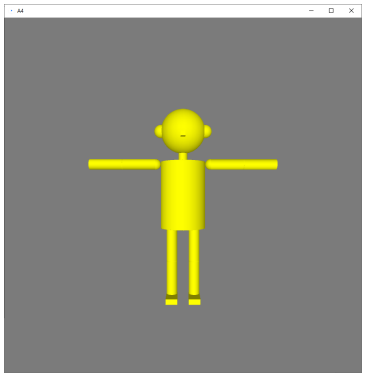
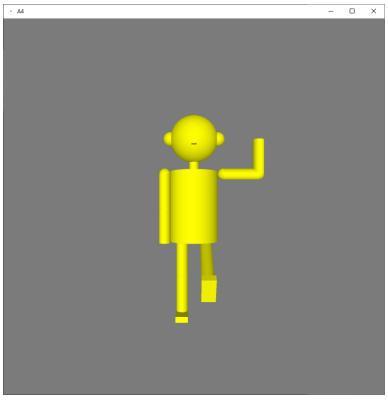
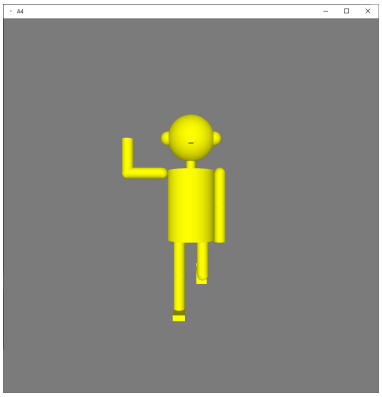
```
PVector[] min = {
   new PVector(-PI/6, -PI/3, 0), // head
   new PVector(-PI/4, -PI/2, 0), // left upper arm
   new PVector(0, 0, 0), // left lower arm
   new PVector(-PI/4, -PI/2, 0), // right upper arm
   new PVector(0, 0, 0), // right lower arm
   new PVector(-PI/6, -PI/2, 0), // left upper leg
   new PVector(-5*PI/6, 0, 0), // left lower leg
   new PVector(-PI/3, 0, 0), // left foot
   new PVector(-PI/6, -PI/2, 0), // right upper leg
   new PVector(-5*PI/6, 0, 0), // right lower leg
   new PVector(-PI/3, 0, 0), // right foot
};

PVector[] max = {
   new PVector(PI/4, PI/3, 0), // head
   new PVector(PI, PI/2, PI), // left upper arm
   new PVector(5*PI/6, 0, 0), // left lower arm
   new PVector(PI, PI/2, PI), // right upper arm
   new PVector(5*PI/6, 0, 0), // right lower arm
   new PVector(PI/2, 0, PI/3), // left upper leg
   new PVector(0, 0, 0), // left lower leg
   new PVector(0, 0, 0), // left foot
   new PVector(PI/2, 0, PI/3), // right upper leg
   new PVector(0, 0, 0), // right lower leg
   new PVector(0, 0, 0), // right foot
};
```

Q4

Active when the program is launched for the first time, and will continue looping until the user chooses one of the pose modes with f, b, or n.

Q5

Pose 0	Pose 1	Pose 2
		
L upper-arm = (0, 0, PI/2) R upper-arm = (0, 0, PI/2)	R upper-arm = (PI/2, 0, PI/2) R lower-arm = (PI/2, 0, 0) R upper-leg = (PI/3, 0, 0)	L upper-arm=(PI/2, 0, PI/2) L lower-arm = (PI/2, 0, 0) R lower-leg = (-PI/2, 0, 0)

Q6

‘f’ - Interpolate between pose 0 and pose 1
‘b’ - Interpolate between pose 0 and pose 2
‘n’ - Interpolate between pose 3 and pose 4 (custom, cheering march?)

‘X/x’ - Increase/decrease X axis spin speed
‘Y/y’ - Increase/decrease Y axis spin speed
‘Z/z’ - Increase/decrease Z axis spin speed