```
/**
 * @author: Caleb Spradlin
 * @version: 04/11/2019
 * Class: NM420
 */

#include <Servo.h>
Servo servo;
const int trigPin = A0;
const int echoPin = A1;
const int flexPinOne = A2;
const int flexPinTwo = A3;
int valueOne, valueTwo;
float duration, distance;

void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  servo.attach(8);
  servo.write(0);
}

void pickThrow(int topFinger, int botFinger){
  //test case for paper
  boolean topFingerBent = false;
  boolean botFingerBent = false;
  if (topFinger < 300) { topFingerBent = true; }
  if (botFinger < 100) { botFingerBent = true; }

  if (topFingerBent && botFingerBent){

    servo.write(90);
    Serial.println("Rock thrown");
    delay(2000);

  } else if (topFingerBent && !botFingerBent){

    servo.write(65);
    Serial.println("Scissors throws");
    delay(2000);

  } else if (!topFingerBent && !botFingerBent){
```

```
    servo.write(30);
    Serial.println("Paper thrown");
    delay(2000);

  }

  delay(2000);
  servo.write(0);
}
void distanceMonitor(){
  valueOne = analogRead(flexPinOne);
  Serial.println(valueOne);
  valueTwo = analogRead(flexPinTwo);
  Serial.println(valueTwo);

  delay(2000);

  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);
  distance = (duration*.0343)/2;
  Serial.print("Distance: ");
  Serial.println(distance);
  if (distance < 50) { pickThrow(valueOne, valueTwo); }
}

void loop() {
  // put your main code here, to run repeatedly:
  distanceMonitor();
}
```