

```

#include <Servo.h>

Servo motor1;
Servo motor2;
Servo motor3;
Servo motor4;
Servo servArray[] = {motor1, motor2, motor3, motor4};

int servAng1 = 90;
int servAng2 = 90;
int servAng3 = 90;
int servAng4 = 90;
int servAngArray[] = {servAng1, servAng2, servAng3, servAng4};

int servPin1 = 3; //avoiding using the same numbers as we used for the
pressure sensors
int servPin2 = 5;
int servPin3 = 6;
int servPin4 = 11;

int fsrPin1 = 0; //defining pins we are using for the pressure sensors
int fsrPin2 = 1; //these must be analogs (A0, A1, A2, A3)
int fsrPin3 = 2;
int fsrPin4 = 4;

int fsr; //fsr reading
double V; //voltage (mV)
double R; //resistance (ohms)
double C; //conductance (micromhos)
double F; //force (N)
double W; //weight (lbs)

double W1, W2, W3, W4; //4 weights read in
double diff1, diff2; //diff1 = maxW - avgW //diff2 = avgW - minW
int maxPosition = 0;
int minPosition = 0;

void setup()
{
  Serial.begin(9600);
}

double calculate(int fsrRead)
{
  if(fsrRead == 0)
  {
    W = 0;
  }
  else
  {
    V = fsrRead * (5000.0 / 1023.0);
    R = ((5000 - V) * 10000) / V;
  }
}

```

```

    C = 1000000 / R;

    if(C <= 1000)
    {
        F = C / 80;
    }
    else
    {
        F = (C - 1000) / 30;
    }
    W = 0.224808943 * F;
}
return W;
}

void loop()
{
    double maxW, minW, sumW, avgW; //calculated by loops

    W1 = calculate(analogRead(fsrPin1));
    W2 = calculate(analogRead(fsrPin2));
    W3 = calculate(analogRead(fsrPin3));
    W4 = calculate(analogRead(fsrPin4));

    double weightArray[] = {W1, W2, W3, W4};

    maxW = W1;
    for(int x = 1; x <= 3; x++) //finds maxW
    {
        if(weightArray[x] > maxW)
        {
            maxW = weightArray[x];
            maxPosition = x;
        }
    }

    minW = W1;
    for(int x = 1; x <= 3; x++) //finds minW
    {
        if(weightArray[x] < minW)
        {
            minW = weightArray[x];
            minPosition = x;
        }
    }

    sumW = 0;
    for(int x = 0; x <= 3; x++) //finds sumW
    {
        sumW += weightArray[x];
    }
}

```

```

avgW = sumW / 4.0; //finds avgW
diff1 = maxW - avgW;
diff2 = avgW - minW;

//Here is where we add the code for what the motors need to need
//first finds a 'neutral' zone in which nothing will happen
if((diff1 - diff2) < (0.01 + (0.2 * avgW)))
{
    //do nothing
}
else if(diff1 > diff2)
{
    //lower the cell with the maxW
    if(servAngArray[maxPosition] >= 10)
    {
        servAngArray[maxPosition] -= 10;
        servArray[maxPosition].write(servAngArray[maxPosition]);
    }
    else
    {
        for(int i = 0; i <= 3; i++)
        {
            if (servAngArray[i] <= 169 && servAngArray[i] >= 10)
            {
                servAngArray[i] += 10;
                servArray[i].write(servAngArray[i]);
            }
        }
    }
}
else //the case where diff2 > diff1
{
    //raise the cell with the minW
    if(servAngArray[minPosition] <= 169)
    {
        servAngArray[minPosition] += 10;
        servArray[minPosition].write(servAngArray[minPosition]);
    }
    else
    {
        for(int i = 0; i <= 3; i++)
        {
            if (servAngArray[i] <= 169 && servAngArray[i] >= 10)
            {
                servAngArray[i] -= 10;
                servArray[i].write(servAngArray[i]);
            }
        }
    }
}
}

```