```cpp
//James Rogers Jan 2022 (c) Plymouth University
#include<iostream>
#include <fstream>
#include<opencv2/opencv.hpp>

using namespace std;
using namespace cv;

#define PI 3.14159265
#define halfPI 1.570796325

int main()
{
    namedWindow("Task 2 - 10613591", WINDOW_NORMAL);
    VideoCapture InputStream("../Task2/Video.mp4");                    //Load in the video as an input stream
    const Point Pivot(592,52);                                         //Pivot position in the video

    //Open output file for angle data
    ofstream DataFile;
    DataFile.open ("../Task2/Data.csv");

    //loop through video frames
    while(true){

        //load next frame
        Mat Frame;
        InputStream.read(Frame);

        //if frame is empty then the video has ended, so break the loop
        if(Frame.empty()){
            break;
        }

        //video is very high resolution, reduce it to 720p to run faster
        resize(Frame,Frame,Size(1280,720));

        //=========================================================Your code goes here=====================================================
        //this code will run for each frame of the video. your task is to find the location of the swinging green target, and to find
        //its angle to the pivot. These angles will be saved to a .csv file where they can be plotted in Excel.
        int HueLower=70;
        int HueUpper=80;
        int SatLower=30;
        int SatUpper=255;
        int ValLower=30;
        int ValUpper=255;

        Mat FrameHSV;
        cvtColor(Frame,FrameHSV,COLOR_BGR2HSV); //convert to HSV
        Mat FrameFiltered;
        Vec3b LowerBound(HueLower, SatLower, ValLower);
        Vec3b UpperBound(HueUpper, SatUpper, ValUpper);
        inRange(FrameHSV, LowerBound, UpperBound, FrameFiltered);

        Moments m = moments(FrameFiltered, true);
        Point p(m.m10/m.m00, m.m01/m.m00);
        Point p_left(m.m10/m.m00, (m.m01/m.m00)-50);                   //left of crosshair
        Point p_right(m.m10/m.m00, (m.m01/m.m00)+50);                  //right
        Point p_up((m.m10/m.m00)+50, (m.m01/m.m00));                   //top
        Point p_down((m.m10/m.m00)-50, (m.m01/m.m00));                 //bottom
        //vertical line point
        Point vert_line_end(592,600);
        //text location
```

```
        Point p_text(650,70);
        Point p_text2(774,100);
        //vertical line
        line(Frame, Pivot, vert_line_end, Scalar(0,0,255),2);
        //
        circle(Frame, p, 20, Scalar(255,0,0), 3);
        circle(Frame, Pivot, 10, Scalar(255,0,0), -1);
        line(Frame, p,Pivot, Scalar(255,0,0), 2);
        line(Frame, p_left, p_right,Scalar(255,0,0), 2);
        line(Frame, p_up, p_down,Scalar(255,0,0), 2);

        //angle calculations
        double angle = (atan2(p.x,p.y))-M_PI_4;                                                    //
PI/4 offset since zero axis is in the negative y direction
        double angle_degrees = angle * (180/M_PI);                                                 //
Conversion to degrees for display purposes
        string text = "Angle ="+to_string(angle)+" Radians";                                       //
String construction
        string text_degrees = to_string(angle_degrees)+" Degrees";
        putText(Frame, text, p_text, FONT_HERSHEY_SIMPLEX, 1, Scalar(0,0,255),2);                  //
Put text on frame
        putText(Frame, text_degrees, p_text2, FONT_HERSHEY_SIMPLEX, 1, Scalar(0,0,255),2);         //
Roughly lines up with line above


        cout << angle <<endl;                                                                      //
Print angle to console
        DataFile << angle <<endl;                                                                  //
Send the output to the data file
        Mat3b combined,FilterRGB;
        cvtColor(FrameFiltered,FilterRGB,COLOR_GRAY2BGR);

        hconcat(Frame,FilterRGB,combined);                                                         //
combine the original and hue-shifted video feeds into one window
        //
============================================================================================================
=========================

        //display the frame
        imshow("Task 2 - 10613591", combined);
        //imshow("Task 2 - 10613591", Frame);
        waitKey(10);
    }

    DataFile.close(); //close output file
}
```