Arduino Code

```cpp
#include <Braccio.h> // Include the Braccio library
#include<Servo.h>

Servo base;
Servo shoulder;
Servo elbow;
Servo wrist_rot;
Servo wrist_ver;
Servo gripper;

// Define constants for link lengths (in cm) for the Braccio arm
const float L1 = 12.5; // Length of link 1 (shoulder to elbow)
const float L2 = 12.5; // Length of link 2 (elbow to wrist)
const float L3 = 7.15; // Length of link 3 (wrist to end-effector)

void setup() {
    Serial.begin(9600); // Initialize serial communication
    delay(1000);
    Braccio.begin(); // Initialize Braccio
}

void loop() {
    //  user to enter end-effector coordinates and orientation randomly
    float x = 10, y = 20, gamma=90;

    // Call inverse kinematics function and move the Braccio arm
    moveBraccio(x, y, gamma);

    delay(1000); // Wait for 1 second before next iteration
}

void moveBraccio(float x, float y, float gamma) {
    //we did the same thing mentioned into the picture finding point (a,b)
    float a = x - (L3 * cos(radians(gamma)));
    float b = y - (L3 * sin(radians(gamma)));
    float C = sqrt(pow(a, 2) + pow(b, 2));

  // Check if position is within reachable workspace
  if ((L1 + L2) > C) {
    // Calculate angles
        float alpha = degrees(acos((pow(L1, 2) + pow(C, 2) - pow(L2, 2)) / (2
* L1 * C)));
```

```
        float Beta = degrees(acos((pow(L1, 2) + pow(L2, 2) - pow(C, 2)) / (2 *
L1 * L2)));

        // finding the final theta1, theta2 and theta3 angles of shoulder,
elbow and wrist respectively.
        float theta1 = degrees(atan2(b, a)) - alpha;

        float theta2 = 180 - Beta;
        float theta3 = gamma - theta1 - theta2;

        // Set servo angles and move the Braccio arm
    //why we added 90 degrees angle in M3 and M4 mentioned into the detains
into the explanation of code
        Braccio.ServoMovement(20,0,int(theta1),90+int(theta2),90+int(theta
3),0,73);

    }
}
```