```cpp
#include <Servo.h>
Servo motor1;
Servo motor2;
// angles in degree and radian

// length of links of robot arm
double L1;
double L2;
// end effector


double pi = 3.14159265359;

double getUserInputBlocking(const char * message) {
  double result = 0;
  bool receiving = true;
  bool acquisitionStarted = false;
  Serial.println(message);
  do {
    int r = Serial.peek();
    if (r != -1) { // got something
      if (isdigit(r)) {
        acquisitionStarted = true;
        Serial.read(); // remove the byte from the
incoming stream
        result = 10 * result + (r - '0'); // do the math, might
overflow.
      } else {
        if (acquisitionStarted) {
          receiving = false; // we are done
        } else {
```

```
      Serial.read(); // ignore that byte and remove it
from the incoming stream
        }
      }
    }
  } while (receiving);
  return result;
}

void setup() {
  motor1.attach(11);
  motor2.attach(3);
  Serial.begin(9600);
  motor1.write(0);
  motor2.write(0);
// Serial.println("Enter the length of first arm ");
L1 = getUserInputBlocking("Enter the length of first
arm ");
Serial.println(L1);
// Serial.println("Enter the length of second arm ");
L2 = getUserInputBlocking("Enter the length of
second arm ");
Serial.println(L2);
digitalWrite(5,HIGH);

Serial.println("Enter Number 1 for
ForwardKinematics");
Serial.println("Enter Number 0 for
inverseKinematics");

  }
```

```arduino
void loop() {

 if (Serial.available())  {

   char choice = Serial.read();
   if (choice == '1'){         //Forwad kinematics Case

    ForwardKinematics();
 }

else if(choice == '0'){

  inverseKinematics();          //inverse kinematics
case

 }

 }
 }

 void ForwardKinematics(){
double angle1 ;
double angle2 ;
double rad_angle1;
double rad_angle2;
double x;
double y;
// Serial.println("Enter the angle1 in degree ");
//  while(Serial.available()==0){}
// angle1=Serial.parsedouble();
```

```
angle1 = getUserInputBlocking("Enter the angle1 in
degree ");
Serial.println(angle1);

//  Serial.println("Enter the angle2 in degree ");
//  while(Serial.available()==0){}
//  angle2=Serial.parsedouble();
angle2 = getUserInputBlocking("Enter the angle2 in
degree ");
Serial.println(angle2);

  rad_angle1 = (angle1*pi)/180;
  rad_angle2 = (angle2*pi)/180;

  motor1.write(angle1);
  delay(1000);
  motor2.write(angle2);

  x = L1 * cos(rad_angle1) +L2 * cos(rad_angle1 +
rad_angle2);
  y = L1 * sin(rad_angle1) +L2 * sin (rad_angle1 +
rad_angle2);
  delay(1000);
 Serial.print("L1 =  ");
 Serial.println(L1);
  Serial.print("L2 = ");
 Serial.println(L2);
 Serial.print("x = ");
 Serial.println(x);
 Serial.print("y = ");
 Serial.println(y);
```

```cpp
   Serial.print("angle1 is ");
   Serial.println(angle1);
   Serial.print("angle2 is ");
   Serial.println(angle2);


   Serial.println("Enter Number 1 for
   ForwardKinematics");
   Serial.println("Enter Number 0 for
   inverseKinematics");

   }

   void inverseKinematics(){
   double angle1 ;
   double angle2 ;
   double rad_angle1;
   double rad_angle2;
   double x;
   double y;
   //   Serial.println("Enter the value x ");
   //     while(Serial.available()==0){}
   //     x=Serial.parsedouble();
   x = getUserInputBlocking("Enter the value x ");

   //     Serial.println("Enter the value y ");
   //     while(Serial.available()==0){}
   //     y=Serial.parsedouble();
   y = getUserInputBlocking("Enter the value y ");

      rad_angle2 = acos((sq(x)+ sq(y) - sq(L1) - sq(L2)) /
```

```
(2*L1*L2));
    rad_angle1= atan(y / x) -
atan((L2*sin(rad_angle2)) / (L1+
L2*cos(rad_angle2)));
    delay(1000);

    angle1= (rad_angle1*180)/pi;
    angle2= (rad_angle2*180)/pi;


 Serial.print("x is ");
 Serial.println(x);
 Serial.print("y is ");
 Serial.println(y);
 Serial.print("angle1 is  ");
 Serial.println(angle1);
 Serial.print("angle2 is ");
 Serial.println(angle2);
 Serial.println("Enter Number 1 for
ForwardKinematics");
 Serial.println("Enter Number 0 for
inverseKinematics");
motor1.write(angle1);
motor2.write(angle2);
delay(2000);
 }
```