calDistance

Write a C program that accepts four decimal values representing the coordinates of two points, i.e. (x1, y1) and (x2, y2), on a plane, and calculates and displays the distance between the points:

$$\text{distance} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Your program should be implemented using functions. Provide two versions of the function for calculating the distance: (a) one uses call by value only for passing parameters; and (b) the other uses call by reference to pass the result to the calling function.

The function prototypes are given below:

```
void inputXY(double *x1, double *y1, double *x2, double *y2);
void outputResult(double dist);
double calDistance1(double x1, double y1, double x2, double y2);
void calDistance2(double x1, double y1, double x2, double y2,
            double *dist);
```

A sample program template is given below to test the functions:

```
#include <stdio.h>
#include <math.h>
void inputXY(double *x1, double *y1, double *x2, double *y2);
void outputResult(double dist);
double calDistance1(double x1, double y1, double x2, double y2);
void calDistance2(double x1, double y1, double x2, double y2, double *dist);
int main()
{
  double x1, y1, x2, y2, distance=-1;

  inputXY(&x1, &y1, &x2, &y2);                  // call by reference
  distance = calDistance1(x1, y1, x2, y2);      // call by value
  printf("calDistance1(): ");
  outputResult(distance);
  calDistance2(x1, y1, x2, y2, &distance);      // call by reference
  printf("calDistance2(): ");
  outputResult(distance);                        // call by value
  return 0;
}
void inputXY(double *x1, double *y1, double *x2, double *y2)
{
  /* Write your code here */
}
void outputResult(double dist)
{
  /* Write your code here */
}
double calDistance1(double x1, double y1, double x2, double y2)
```

```
{
  /* Write your code here */
}
void calDistance2(double x1, double y1, double x2, double y2, double *dist)
{
  /* Write your code here */
}
```

Some sample input and output sessions are given below:

(1) Test Case 1:

```
Input x1 y1 x2 y2:
1 1 5 5
calDistance1(): 5.66
calDistance2(): 5.66
```

(2) Test Case 2:

```
Input x1 y1 x2 y2:
-1 -1 5 5
calDistance1(): 8.49
calDistance2(): 8.49
```