Unit 6: Coordinates | **Skill Builder 2: The Distance Formula**

In this lesson, you will use the TI-Innovator Rover to mark two points on its coordinate plane and determine the distance between the points using the distance formula.

Objectives:

- Move to two different points
- MARK the points with SOUND and COLOR and a marker
- Compute and display the distance between the two points
- Measure the distance between the points to confirm the calculation
- Estimate the error in the movement vs. the computation

This activity requires that a marker be used in the Rover's marker holder to leave a trail on the paper.

We'll write a program that marks two points on the plane and computes the distance between the two points. You will then measure the distance between the two points and compare the measurement with the calculation.

```
NORMAL FLOAT AUTO REAL DEGREE MP
EDIT MENU: [alpha] [f5]
PROGRAM:ROVER62
:ClrHome
:Disp "ROVER UNIT6 SB2"
:Send("CONNECT RV")
:Send("SET RV.GRID.M/UNIT
.0254")
:Input "X1?",X
:Input "Y1?",Y
:Input "X2?",Z
:Input "Y2?",W
```

1. Start your program with **CONNECT RV** and include the Rover command to set the unit distance to one inch (1 inch ≈ 0.0254 meters). The **RV.GRID. M/UNIT** command is found in the **Rover (RV)… > RV Setup…** menu.

   Send("SET RV.GRID.M/UNIT .0254")

2. Include four **Input** statements for the coordinates of the two points. Keep in mind that variables on the TI-84 Plus CE are single letters only.

   Note that we use the variables X, Y, Z, and W for the coordinates.

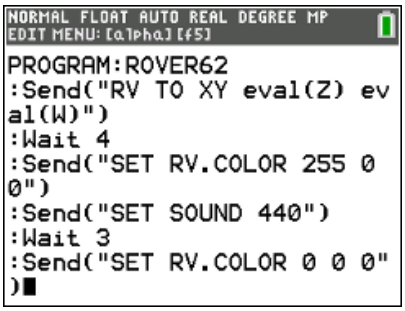
> **Teacher Tip**: The default unit distance is 10 cm. Setting the unit distance to 1 inch ensures that there's no scaling needed to convert from the measured distance (in inches) to the calculated distance. "M/UNIT" stands for Meters per Unit, so 0.0254 M/UNIT represents 1 inch per unit of movement.

3. Next, we get the Rover to move to the first point, **Wait** while the Rover moves, and then make a **SOUND** and LED **color** to indicate that the Rover has reached its point. We will also use a **Pause** statement here and turn the LED off after pressing **enter**. Recall that the sound will play for only one second by default.

```
NORMAL FLOAT AUTO REAL DEGREE MP
EDIT MENU: [alpha] [f5]
PROGRAM:ROVER62
:Send("RV TO XY eval(X) ev
al(Y)")
:Wait 4
:Send("SET RV.COLOR 0 200
200")
:Send("SET SOUND 256")
:Pause "PRESS ENTER"
:Send("SET RV.COLOR 0 0 0"
)
```

> **Teacher Tip:** When the Rover stops, the marker tends to leave a spot of ink. Adjust the **Wait** statement to control the size of the ink blot or just note that there will be a corner at that spot. The program will draw two segments.

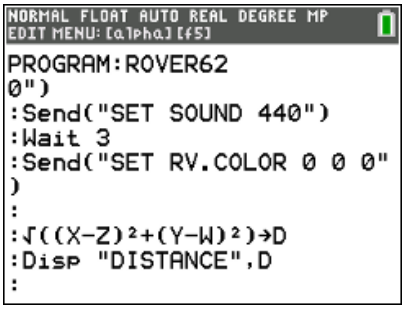
4. Next, we get the Rover to move to the second point. A different sound and different LED color is used to indicate that the Rover has reached the point. A **Wait** statement is used here to keep the LED on for a few seconds before turning it off.



5. Finally, program the calculator to compute and display the distance between the two points using the Distance Formula:

$$\sqrt{((X-Z)^2+(Y-W)^2)}\rightarrow D$$

Be careful with the parentheses.



6. Measure the second segment drawn (between the two points) and compare that measurement with the computed value. How does the measurement compare with the computed value? What is the percent error of the measurement?

   *(measurement-**D**istance)/**D**istance*100*

Teacher Tip: The **Wait** command values are approximate and can be adjusted.

The completed program:

```
ClrHome
Disp "ROVER UNIT6 SB2"
Send("CONNECT RV")
Send("SET RV.GRID.M/UNIT .0254")
Input "X1?",X
Input "Y1?",Y
Input "X2?",Z
Input "Y2?",W
Pause "PRESS ENTER TO START"
Send("RV TO XY eval(X) eval(Y)")
Wait 4
Send("SET RV.COLOR 0 200 200")
Send("SET SOUND 256")
Pause "PRESS ENTER"
Send("SET RV.COLOR 0 0 0")
Send("RV TO XY eval(Z) eval(W)")
Wait 4
Send("SET RV.COLOR 255 0 0")
Send("SET SOUND 440")
Wait 3
Send("SET RV.COLOR 0 0 0")
√((X-Z)²+(Y-W)²)→D
Disp "DISTANCE",D
```