

NUMB3RS Activity: A Pursuit Curve Problem Episode: "Spree"

Topic: Data collection, graphs, distance formula

Grade Level: 9 - 12

Objective: In this activity, students will study the path of four bugs that are in pursuit of one another. Students will identify the path of motion for each bug and use the distance formula to find the distance each bug travels before catching its prey.

Time: 25 - 30 minutes

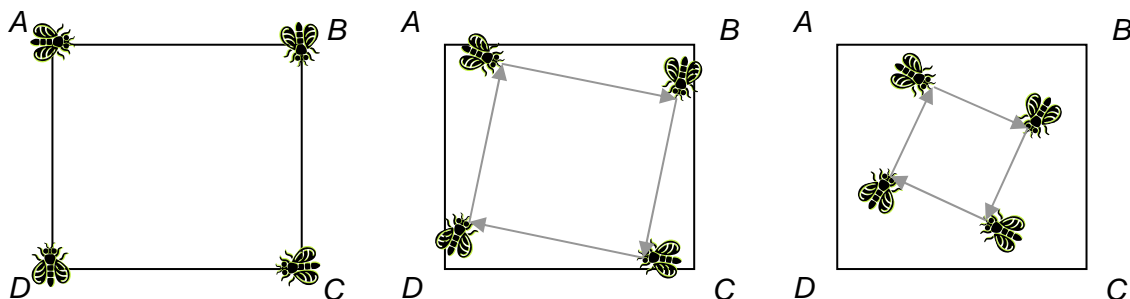
Materials: TI-83 Plus/TI-84 Plus graphing calculator, TI-Navigator™ system, and the following files: *ThreeBugs.act*, *FourBugs.act*, *FiveBugs.act*, and *DISTANCE.8xp*. To download these files, go to <http://education.ti.com/exchange> and search for "7821."

Introduction

In "Spree," two lovers, Buck and Crystal, have committed a series of crimes at various locations ranging from Austin, Texas to San Bernardino, California. Agents Eppes and Edgerton have plotted a precise map of their movements and enlisted Charlie's help in trying to detect a pattern from their analysis. Later in the episode, Charlie thinks of and describes a possible solution: "This map tells us only half of the story of two people, traveling across the country, committing crimes. But that's only half the story. It's like we're watching an airplane, flying across the sky, banking and spinning. From the flight path we try to discern a pattern, perhaps even a destination. What the map doesn't show us, what we're not seeing is that there's a second airplane, chasing after the first one, affecting the plane's movement and weighing on each of the pilot's decisions."

Charlie has concluded that the agents' pursuit of Buck and Crystal has forced them to change their paths: "What we are seeing is a variation of something called a *pursuit curve*, created when one point chases another. You see, if a fighter pilot flies straight at the point where he sees an enemy plane, by the time he gets there, the plane will be gone. Just to maintain his following distance, the chasing pilot has to constantly adjust his direction, matching the movements of his quarry. If he wants to catch the plane, he has to anticipate its movement and aim his nose ahead of the lead plane."

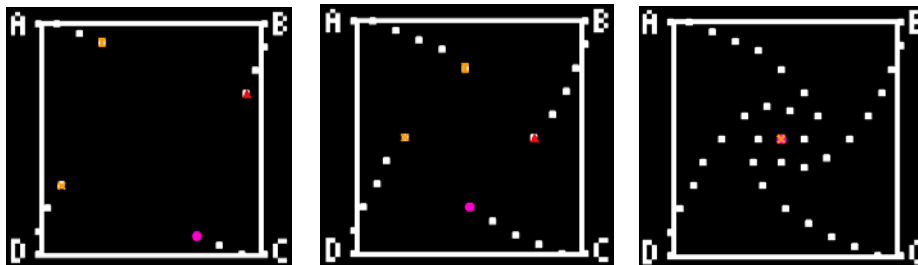
In this activity, students will study a *pursuit curve* problem modeled by four bugs chasing one another within the confines of a square. This is commonly referred to as the Four Bug Problem. As shown in the diagram below, the four bugs start at each of the four corners of the square. They then start moving towards their prey, the bug found by moving in a clockwise direction. That is, Bug A chases Bug B, Bug B chases Bug C, and so on. After some time, the position of the bugs will resemble the second and third diagrams shown. This activity examines the path that each bug follows.



Part I: Chasing Bugs

1.
 - a. Launch TI-Navigator™ on the computer and press **Begin Class** to start the session.
 - b. Have each student log into NavNet on their calculators.

2.
 - a. Load the **FourBugs.act** activity settings file into Activity Center and click the 'List-Graph' tab. This will display a square superimposed on a grid.
 - b. Click 'View' and select 'Individualize Student Cursors.'
 - c. Select four students to be the "bugs" for this activity and assign each of them to the letters *A*, *B*, *C*, and *D*.
 - d. Press **Start Activity** and tell each "bug" to move to their assigned corner of the square and mark a point. Carefully track their coordinates and make sure that they mark their points at $A(-5, 5)$, $B(5, 5)$, $C(5, -5)$, and $D(-5, -5)$.
 - e. Each bug should move in the direction of the bug it is chasing. It is important that the bug moves in the direction of the last marked point. For each turn, students are allowed to press the left/right arrow keys (← and →) a maximum of 5 times and the up/down arrow keys (↑ and ↓) a maximum of 5 times. Have each bug mark its position at the end of each turn. This activity is designed so that the bugs should converge on one another at the tenth and final point. Tell students that they should try to actually "catch" their prey on this last point. If they have done this activity correctly, this last point should be very close to the center of the square. The sequence of screens below shows the expected result from this activity. If students have some difficulty with these procedures, stop the activity, clear the activity data, and redo the activity. You may also want to give different students an opportunity to try out the activity.



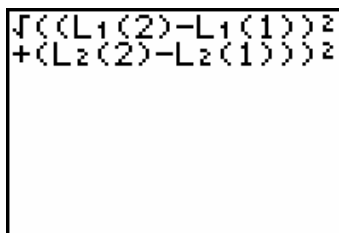
- f. After the bugs have converged somewhere near the center of the square, press **Stop Activity**. Click the column heading above the names of the students to sort the points alphabetically and highlight an individual student's points as shown on the left below. This will reveal this particular student's path in blue, as shown on the right below.

Rachel	-5	5
Rachel	-4.2	5
Rachel	-3.2	4.6
Rachel	-2.2	4.2
Rachel	-1.2	3.8
Rachel	-.2	3
Rachel	.8	2
Rachel	1.4	1
Rachel	.8	0
Rachel	-.2	0

- g. Discuss the patterns seen in the graph, with particular emphasis being placed on the fact that the path creates a spiral, and that the bugs eventually converge at the center of the square.

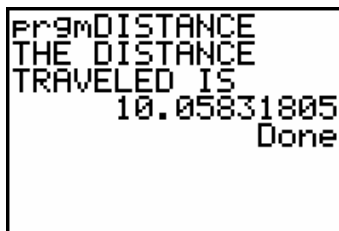
Part II: Calculate Distances

3. a. In this part of the activity, students will calculate the distance each bug traveled by finding the sum of the distances between successive points.
- b. Instruct students to exit NavNet on their calculators. With the Activity Center window in view, have each student input the data for their assigned bug into lists **L1** and **L2**. (Note: You will need to scroll up and down to reveal the data.)
- c. If you want students to practice working with the distance formula, have them calculate the individual distances between pairs of points by hand. You can also show them how to use the calculator and the $\boxed{2\text{nd}}$ [ENTRY] command to quickly find each distance. The command below will give the distance between the points contained in the first two rows of **L1** and **L2**.



```
√((L1(2)-L1(1))²  
+(L2(2)-L2(1))²
```

As another option, you can send students the program **DISTANCE** which, when executed, will automatically calculate distances, find their sum, and display the result on the home screen. This is the length of the path traveled by the selected bug.



```
PrgmDISTANCE  
THE DISTANCE  
TRAVELED IS  
10.05831805  
Done
```

- d. Discuss the results with students and point out that their distances should be close to the length of the side of the square (10 units).

The goal of this activity is to give your students a short and simple snapshot into a very extensive math topic. TI and NCTM encourage you and your students to learn more about this topic using the extensions provided below and through your own independent research.

Extensions

The Four Bug Problem can also be extended to any number of bugs greater than or equal to 3. That is, you can use three bugs and an equilateral triangle, five bugs and a regular pentagon, and so on. Repeat this activity using the activity settings files ***ThreeBugs.act*** and ***FiveBugs.act***. In each case, have students examine the relationship between the length of one bug's path and the side length of the shape. Ask students to make a conjecture about what happens as the number of sides (and bugs) increases.

Additional Resources

- These two Web sites have graphics and other information regarding the Four Bug Problem.
<http://mathworld.wolfram.com/MiceProblem.html>
<http://www.mathpages.com/home/kmath492/kmath492.htm>
- The *NUMB3RS* activity "The Four Bug Problem: Step on No Pets" includes a classroom activity where students will walk along a pursuit curve. To download this activity, go to <http://education.ti.com/exchange>, and search for "7423."
- The *NUMB3RS* activity "The Chase" further explores pursuit curves, in the context of an FBI agent chasing a pair of criminals. To download this activity, go to <http://education.ti.com/exchange>, and search for "7422."
- An animated description of pursuit curves can be found at <http://curvebank.calstatela.edu/pursuit/pursuit.htm>.
- A more complex description of pursuit curves can be found at <http://mathworld.wolfram.com/PursuitCurve.html>.
- If you would like to learn more about TI-Navigator™, visit <http://education.ti.com/navigator>.