Computer Algebra Systems Activity: Solving Diophantine Equations

© R. Meisel, April 14, 2009
rollym@vaxxine.com

Topic: Solving Diophantine Equations

Notes to the Teacher:

a) This activity is designed to use the CAS on the TI-Nspire CAS calculator to enhance understanding and instruction. All screen shots are from the TI-Nspire CAS.

b) The instructions for the activity assume that the user has some elementary experience with a CAS. Novice users should complete the activity Computer Algebra Systems: An Introduction before attempting this activity.

c) The activity is presented in a **Teacher Version**, with all screen shots and solutions present, as well as a **Student Version**, which can be duplicated and handed out to students.

d) This material may be used freely by teachers in their classrooms.

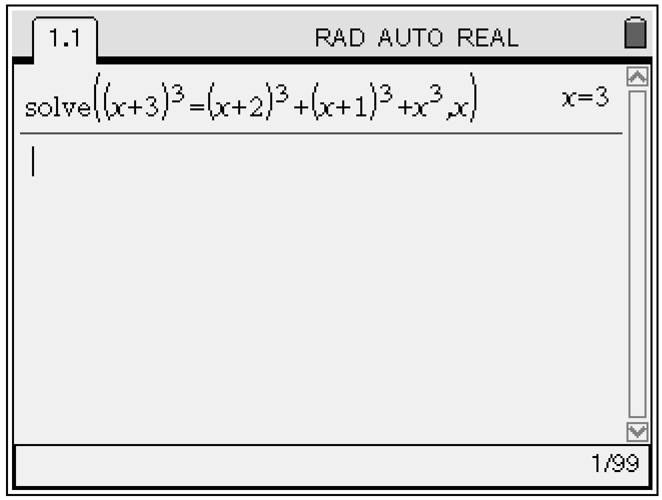
Computer Algebra Systems Activity: Solving Diophantine Equations

© R. Meisel, April 14, 2009
rollym@vaxxine.com

Teacher Version:

Introduction: A Diophantine Equation is one which has integer coefficients, and for which only integer solutions are desired. They are named after Diophantus of Alexandria, a Greek mathematician who lived in the third century C.E., and is sometimes known as the "father of algebra".

1. Find four consecutive integers such that the cube of the largest equals the sum of the cubes of the other three. This problem leads to the Diophantine equation: $(x+3)^3 = (x+2)^3 + (x+1)^3 + x^3$. You can select **1:Solve** under the **Algebra** menu on your TI-Nspire CAS to try to find a solution for this equation.



Is there a solution which consists of integers?

[Answer: yes. $6^3 = 5^3 + 4^3 + 3^3$.]

2. Some Math History: A famous Diophantine equation occurs in Fermat's Last Theorem. Fermat was considering the equation

$x^n + y^n = z^n$

If $n = 2$, the equation becomes the Pythagorean theorem, which has many integral solutions, such as 3, 4, and 5.

Find two other "Pythagorean triples", which are not multiples of 3, 4, and 5, or of each other.

[Answer (may vary): 5, 12, 13; 7, 24, 25.]

Fermat conjectured that there are no integral solutions when $n > 2$. As an example, there are no integers x, y, and z such that $x^3 + y^3 = z^3$. Fermat noted that he had found a proof of this conjecture, but the proof was not discovered among his papers when Fermat died in 1665. His theorem remained a conjecture until 1994, when Andrew Wiles published a proof.

3. Another Diophantine equation is Pell's equation

$x^2 - ny^2 = 1$

Consider the special case, when $n = 92$.

$x^2 - 92y^2 = 1$

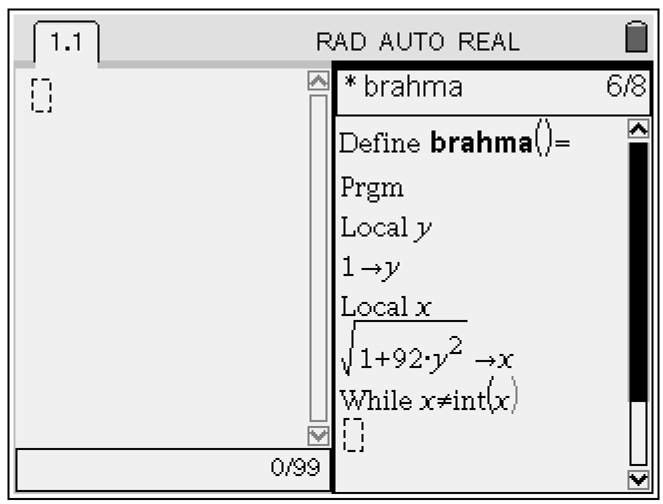
This is known as Brahmagupta's Problem. While many Diophantine equations have no solutions, this one does.

To find positive solutions, solve the equation for x, keeping the positive root.

[Answer: $x = \sqrt{1+92y^2}$]

By inspection, it is obvious that any real number substituted for y will result in a solution for the equation. However, the trick is to find an integral solution. You will use the programming power of your TI-Nspire CAS to do this.

4. Press **menu**. Select **8: Functions&Programs**, then **1:Program Editor** and then **1: New**… . Type a name for your program, such as brahma. Tab to **OK**, and press **enter**. A program editing window will open.



Tab to the blank the line after **Prgm**. Press **menu**. Select **3:Define Variables**, then **1:Local**. Type ***y*** then **enter**. Store the value 1 in ***y***. Define another local value ***x***, and store the expression derived above in ***x***. You have now initialized the values of the variables that you want to use.

You are looking for an integral value of ***y*** that makes the value of ***x*** an integer as well. You will use the "guess and check" method, starting from ***y*** = 1, and increasing the value of ***y*** by 1 until you find one that works.

Press **menu** and select **4:Control**. Select **6:While…EndWhile**. Type *x* **ctrl** = int(*x*) to control the loop. The **int()** function returns the largest integral value of a number which is less than or equal to the number.

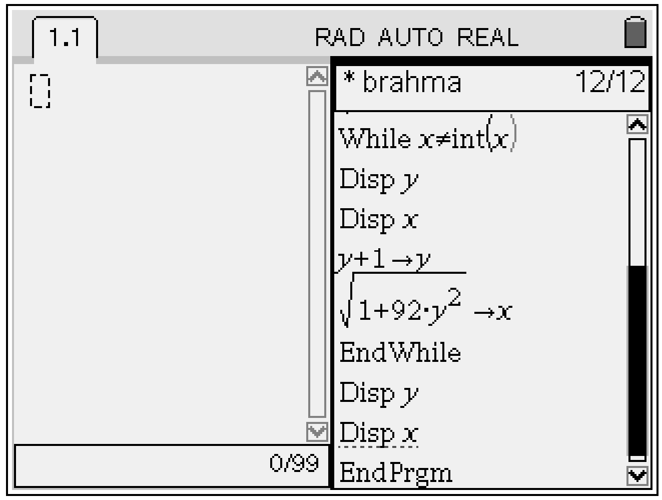
Note: the ≠ is entered by pressing **ctrl**, then =.

Press **menu** and select **6:I/O**. Select **1:Disp**, followed by *y*. In a similar manner, display the value of *x*.

Increase the value of *y* by 1, and store in *y*.

Calculate the new value of *x*, and store in *x*.

Tab to the end of the line **EndWhile**, and press **enter** to insert a line.



Display the values of *y* and *x* once again.

Check your program carefully to ensure that you have entered all lines using the correct syntax. Ensure that you understand the flow of the program, and what each line does.

Press **menu**, and select **2:Check Syntax & Store**, and then **1:Check Syntax & Store**. This will check that you have entered commands correctly, and store your program in memory.

If ***x*** = **int(*x*)**, then ***x*** must be an integer. When this condition is met, the loop ends.

Press **ctrl** then **tab** to return to the calculator page. Enter the name of your program, such as **brahma()** in the command line. Don't forget the brackets. The program will run slowly enough that you can follow its progress on the calculator screen. Be patient. It may take a bit of time to find the solutions.

What are the smallest integral solutions for ***x*** and ***y***?

[Answer: *x* = 1151 and *y* = 120.]

Note: If your program didn't work properly, and continued to run, you can stop it by pressing the **on** key.

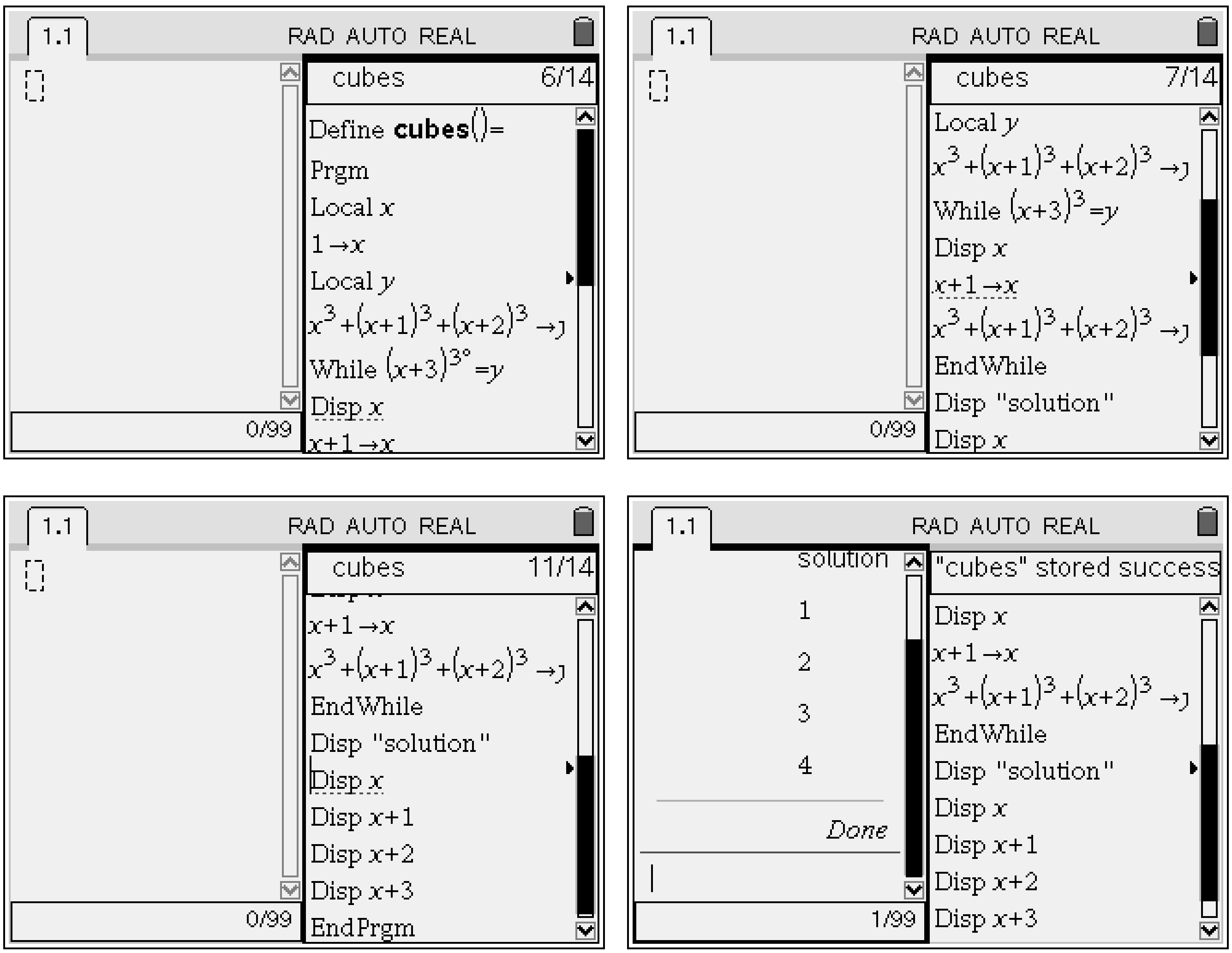
To close the program window, press **menu**, select **1:Actions**, and then **D:Close**.

5. Adjust your program to find solutions of Pell's equation using other values, such as 8, 50 and 99.

[Answer:

n	*x*	*y*
8	3	1
50	99	14
99	10	1

6. Write a program to solve the problem posed in part (1).



Computer Algebra Systems Activity: Solving Diophantine Equations

Student Version:

Introduction: A Diophantine Equation is one which has integer coefficients, and for which only integer solutions are desired. They are named after Diophantus of Alexandria, a Greek mathematician who lived in the third century C.E., and is sometimes known as the "father of algebra".

1. Find four consecutive integers such that the cube of the largest equals the sum of the cubes of the other three. This problem leads to the Diophantine equation: $(x+3)^3 = (x+2)^3 + (x+1)^3 + x^3$. You can select **1:Solve** under the **Algebra** menu on your TI-Nspire CAS to try to find a solution for this equation.

Is there a solution which consists of integers?

2. Some Math History: A famous Diophantine equation occurs in Fermat's Last Theorem. Fermat was considering the equation

$x^n + y^n = z^n$

If $n = 2$, the equation becomes the Pythagorean theorem, which has many integral solutions, such as 3, 4, and 5.

Find two other "Pythagorean triples", which are not multiples of 3, 4, and 5, or of each other.

Fermat conjectured that there are no integral solutions when $n > 2$. As an example, there are no integers x, y, and z such that $x^3 + y^3 = z^3$. Fermat noted that he had found a proof of this conjecture, but the proof was not discovered among his papers when Fermat died in 1665. His theorem remained a conjecture until 1994, when Andrew Wiles published a proof.

3. Another Diophantine equation is Pell's equation

$x^2 - ny^2 = 1$

Consider the special case, when $n = 92$.

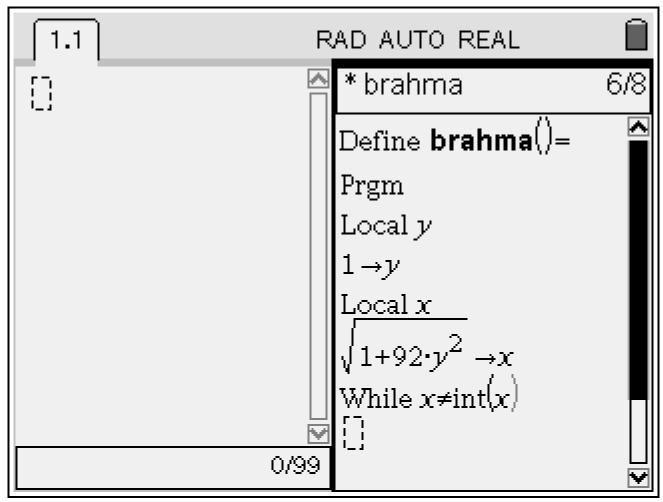
$x^2 - 92y^2 = 1$

This is known as Brahmagupta's Problem. While many Diophantine equations have no solutions, this one does.

To find positive solutions, solve the equation for x, keeping the positive root.

By inspection, it is obvious that any real number substituted for y will result in a solution for the equation. However, the trick is to find an integral solution. You will use the programming power of your TI-Nspire CAS to do this.

4. Press **menu**. Select **8: Functions&Programs**, then **1:Program Editor** and then **1: New**… . Type a name for your program, such as brahma. Tab to **OK**, and press **enter**. A program editing window will open.



Tab to the blank the line after **Prgm**. Press **menu**. Select **3:Define Variables**, then **1:Local**. Type ***y*** then **enter**. Store the value 1 in ***y***. Define another local value ***x***, and store the expression derived above in ***x***. You have now initialized the values of the variables that you want to use.

You are looking for an integral value of ***y*** that makes the value of ***x*** an integer as well. You will use the "guess and check" method, starting from ***y*** = 1, and increasing the value of ***y*** by 1 until you find one that works.

Press **menu** and select **4:Control**. Select **6:While…EndWhile**. Type *x* **ctrl** = int(*x*) to control the loop. The **int()** function returns the largest integral value of a number which is less than or equal to the number.

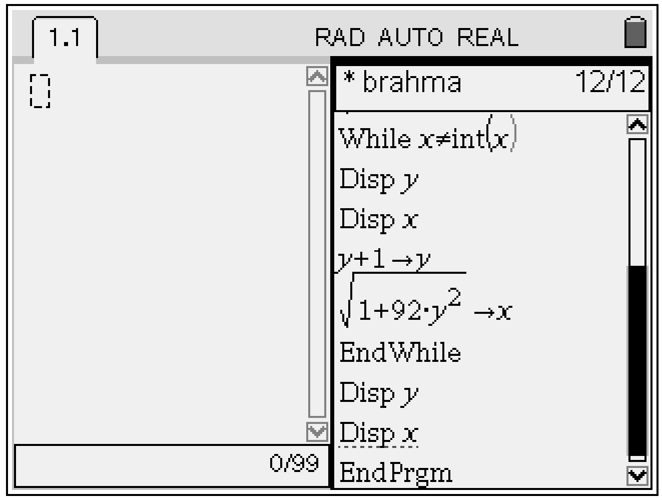
Note: the ≠ is entered by pressing **ctrl**, then =.

Press **menu** and select **6:I/O**. Select **1:Disp**, followed by *y*. In a similar manner, display the value of *x*.

Increase the value of *y* by 1, and store in *y*.

Calculate the new value of *x*, and store in *x*.

Tab to the end of the line **EndWhile**, and press **enter** to insert a line.



Display the values of *y* and *x* once again.

Check your program carefully to ensure that you have entered all lines using the correct syntax. Ensure that you understand the flow of the program, and what each line does.

Press **menu**, and select **2:Check Syntax & Store**, and then **1:Check Syntax & Store**. This will check that you have entered commands correctly, and store your program in memory.

If ***x*** = **int(*x*)**, then ***x*** must be an integer. When this condition is met, the loop ends.

Press **ctrl** then **tab** to return to the calculator page. Enter the name of your program, such as **brahma()** in the command line. Don't forget the brackets. The program will run slowly enough that you can follow its progress on the calculator screen. Be patient. It may take a bit of time to find the solutions.

What are the smallest integral solutions for ***x*** and ***y***?

[Answer: *x* = 1151 and *y* = 120.]

Note: If your program didn't work properly, and continued to run, you can stop it by pressing the **on** key.

To close the program window, press **menu**, select **1:Actions**, and then **D:Close**.

5. Adjust your program to find solutions of Pell's equation using other values, such as 8, 50 and 99.

n	***x***	***y***
8		
50		
99		

6. Write a program to solve the problem posed in part (1).