Lesson 2: Introducing Map Algebra

Topic: Map algebra

    Concepts
- **The Map Calculator**
- **Expressions**
- **Writing expressions**
- **Help for writing expressions**
- **Map Calculator syntax rules**
- **Avenue requests and the Map Calculator**
- **Request rules**

    Exercise
- **Find help for a request**

Topic: Map algebra operators

    Concepts
- **Measurement scales**
- **Using grids and numbers**
- **Arithmetic operators**
- **Relational operators**
- **Boolean operators**
- **Conditional processing**
- **Requests or menu options?**

    Exercises
- **Use Avenue requests with the Map Calculator**
- **Use arithmetic operators**
- **Use relational and Boolean operators**
- **Use conditional processing**

Lesson summary

Lesson self test

Goals

In this lesson, you will learn:

- what map algebra is
- how to use the Map Calculator to create map algebra expressions
- how to use Avenue requests in the expression builder
- what rules apply to writing expressions and using requests
- how to use map algebra operators

TOPIC1: Map algebra

Map algebra is a high-level computational language for performing spatial analysis using grid (raster) data. This language establishes a set of conventions for data processing control. The conventions describe how operations are specified, the data to operate on, and the order in which the operations should be processed.

The language has been designed for ease of use and understanding. It it a mixture of algebra and normal prose. While the prose influence allows for intelligible statements, the algebra maintains the power of the mathematical base underlying the grid's cell-based structure.

Map algebra operations are similar to spatial overlay operations, but when comparing the layers, there is a mathematical (algebraic) component. The overlay operations tend to use expressions that contain And, Or, and Xor, standard Boolean connectors.

Map algebra uses math-like expressions that normally return numeric values. These values are assigned to an output grid.

In the example below, the two themes [Pop90] and [Pop60] are being overlayed, but the overlay expression is really an algebraic expression, (A - B) = C. A is the [Pop90], B is the [Pop60], and C is the difference in population [Change] between 1990 and 1960.
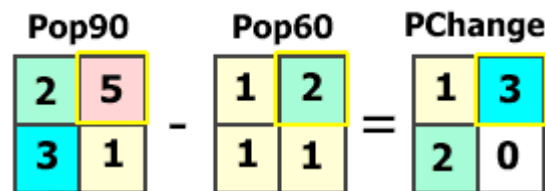
[Pop90] - [Pop60] = [PChange]

5   -   2   =   3

Map algebra uses expressions consisting of grids and mathematical operators.

The example could be continued to show percentage of population growth with the following equation:

(([PopChange] /[Pop60]) * 100)

The highlighted cell would display a 150 percent growth from 1960 to 1970, (3 / 2) = 1.5, (1.5 * 100) = 150 percent growth.

For more information about map algebra, see Tomlin, C. Dana. *Geographic Information Systems and Cartographic Modeling.* Englewood Cliffs: Prentice Hall, 1990.

Concept

The Map Calculator

The Map Calculator helps you create an expression that will result in a new theme. The expression can perform analysis on one or many grid themes in the active view. You create algebra-like expressions either by typing into the expression box or by clicking the layers, operators, and requests.
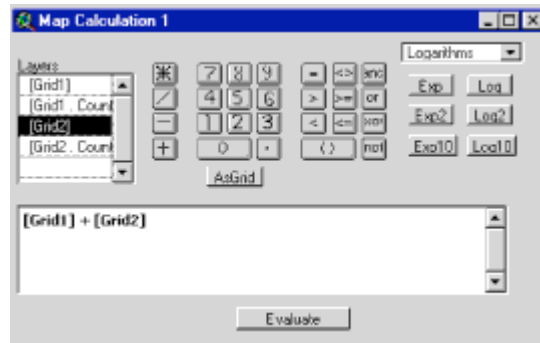
The Map Calculator allows you to create mathematical statements with the grid themes found in the active view. It creates an output grid theme using an expression. Here, two themes are being added together. [Click to enlarge]

Layers are objects that represent grid themes. Grid themes in the active view are listed in the layers list. Integer grid themes have a separate layer for every numeric field in the Value Attribute Table (VAT). These layers are listed as [Grid name.Field Name]. Other objects that can be used in the Map Calculator include filenames, numbers, strings, and any other supported Avenue objects.

Operators are the mathematical operators (i.e., *, /, +, -) or relational operators (i.e., >, <, etc.). These map algebra operators are considered Avenue requests.

Mathematical operations including arithmetic, logarithmic, trigonometric, and powers choices on the Map Calculator are also considered Avenue requests. There are many more Avenue requests that do not appear in the Map Calculator, but you can find them in the online help.

When the Map Calculator opens, it has a set of empty parentheses just like the standard ArcView GIS Query Builder. To start your expression, click between the parentheses and then enter the expression by double-clicking a layer or typing.

The result of a Map Calculator expression is a new temporary theme (grid dataset) that is stored in the project's working directory. If the theme is deleted, the corresponding dataset in the working directory is also deleted. Save the dataset from the Theme menu by choosing Save Data Set or simply by saving the project.

To change the expression used to create the grid theme, choose Edit Theme Expression from the Theme menu. This brings up the Map Calculator and the expression used to create the theme. This works for temporary and permanent grids created with the Map Calculator

Concept

Expressions

Map Calculator expressions are made up of objects, requests, and request parameters.

Objects are ArcView objects like grids, strings, and numbers. For a listing of supported ArcView objects, look in the online help at the ArcView architecture for Class Hierarchy.

Requests are commands that perform some operation on or with the objects. Requests can be mathematical operators (+, -, /, *) or spatial analysis operations (e.g., Slope, Aspect, HillShade, Visibility).

Some requests have parameters which are also objects. Parameters provide specific information on how the request is to be carried out.

> [Grid7].HillShade(315, 45, nil)

In the above example, the HillShade request has three parameters. The first parameter (315) is the compass direction or azimuth of the imaginary light source. The second parameter (45) is the imaginary light source's angle above the horizon. The third parameter (nil) is the zFactor. Nil is a way of indicating no value. In this case, the default value of 1 is used for the zFactor.

You can think of Avenue expressions as saying "Hey you, do something!" The respondent might hit you for your ill manners; however, in this case, the person is the object and your instruction is the request.

If your request is a simple one like "Follow me," no further instruction is required. If you said "Please get me a drink," the person would need to know whether you wanted a hot or cold drink, a nonalcoholic or alcoholic drink, and how much you were willing to pay for it. Fortunately, objects in Arcview are very obedient and always ready to do your bidding.

Concept

Writing expressions

Expressions used in the Map Calculator follow Avenue syntax rules using objects and requests.

```
Object.Request(Parameters)
```

or

```
Object Request Object
```

The general expression format is written in object request syntax where there is an object, a dot, and a request (e.g., [myGrid].Aspect). In the following example, the Sin request calculates the sine of each cell in Grid7.

```
[Grid7].Sin
```

The format is slightly different for mathematical operations, where there are usually two grid objects with a request (or mathematical operator) in the middle. Here are some examples:

```
[Grid1] + [Grid2]
[Grid3] / [Grid4]
[Grid5] - 360
```

Help for writing expressions

ArcView's online help provides a description of every request and type of object (class). The help topic for each request provides a description of what the request does and what parameters are required for it to work.
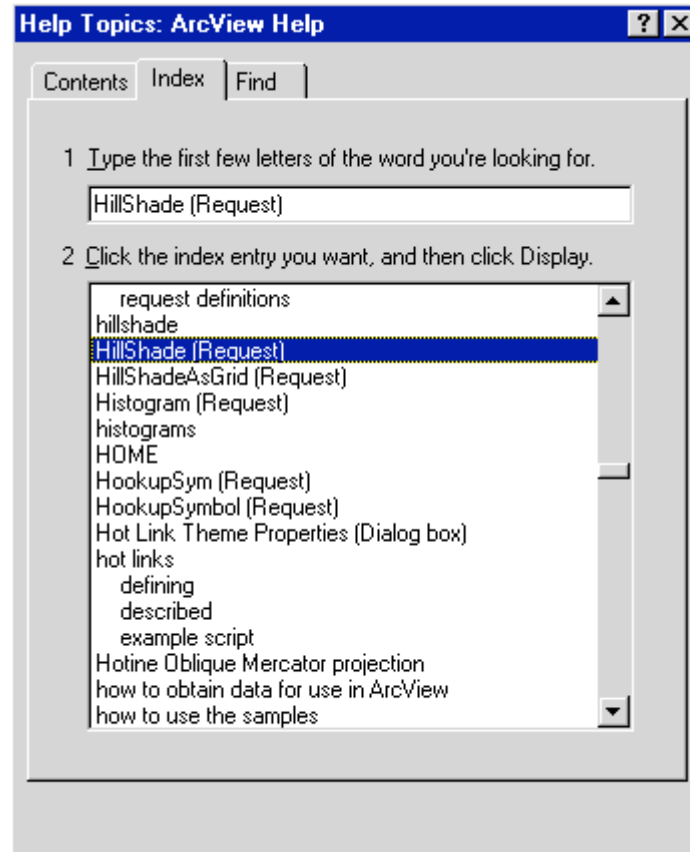
ArcView's online help system consists of numerous topics, including all requests. Here, the Hillshade request is selected.

Each request's help topic includes a generic example of how you might use the request. This syntax example is important because it shows you the type of object that the request is sent to (in the above example, the HillShade request is sent to aGrid), and the type of objects that are required for each parameter (in the following example, the HillShade request has three parameters: anAzimuth, anAltitude, and a zFactor).
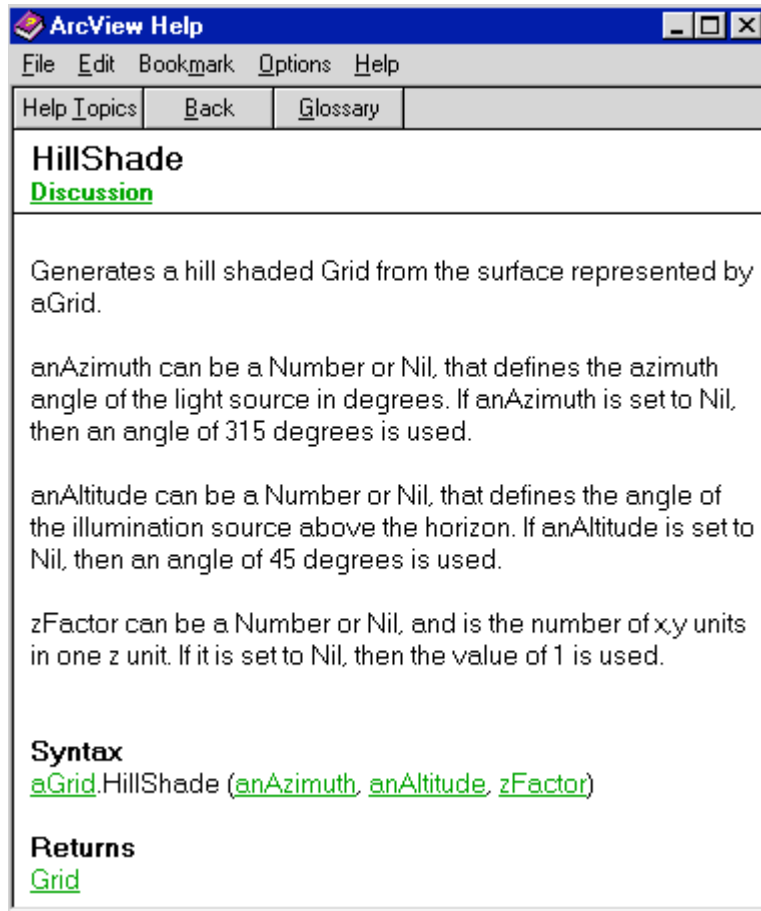
Description of the Hillshade request within ArcView Help.

Each request that you issue in the Map Calculator must have a grid as its returned object. If the returned object is something other than a grid, you will get an error message.

Concept

Map Calculator syntax rules

When you're creating expressions in the Map Calculator, there are some general rules you need to remember.

1. When using the Map Calculator, the view with your grid themes should be the current active document. If you enter an expression in the Map Calculator, then click the project window or some other document and click Evaluate, you will get an error.
2. Grid names in a Map Calculator expression are always in square brackets (e.g., [Slope]). Inside the Map Calculator, there is a list of grids from which you can select. You can select from the Layers list by double-clicking or you can type in the expression yourself.
3. Some requests require a character string to be entered, especially when one of the request parameters is a FileName. You would have to enter the path name as a string followed by the AsFilename request, which then creates a grid dataset in the specified path.

```
"c:\data\grid1".AsFileName
```

4. Optional blank spaces can be used to separate objects, requests, and parameters. The blank spaces are used to help readability in the Map Calculator.

```
[Grid1] * [Grid2]
```

5. Expressions are evaluated from left to right, regardless of the mathematical operator. Be sure to control the order of evaluation by using parentheses.

```
[Grid1]+([Grid2]*[Grid3])
```

6. You may nest expressions as long as the entire expression evaluates to a grid. The inner-most parentheses set is evaluated first.
7. For long and complex equations, it is perfectly OK and helpful to go to the next line. Again, readability is improved.

```
(([Grid1] + [Grid2]) * 0.40)+
((([Grid3] + [Grid4]) * 0.10)*
(([Grid5] + [Grid6])
```

Concept

Avenue requests and the Map Calculator

Avenue is very important to ArcView Spatial Analyst. Only a few of the most popular and important Spatial Analyst functions are available from the user interface (the Analysis menu, the Surface menu, and other menu choices, buttons, and tools). However, every ArcView Spatial Analyst function is available as an Avenue request.

You do not have to know Avenue to be successful in using these requests, but you do have to learn to use the Map Calculator and some Avenue syntax. The Map Calculator can perform analysis on one or many grid themes.

Remember that you can create algebraic expressions either by typing into the expression box or by clicking on objects, operators, and requests. Objects are the layers representing grid themes or numbers on the calculator. Operators are the mathematical operators (i.e., +, -, *, /) or relational operators (i.e., >, <, etc.).

Map Calculator requests are sent to a grid. Map Calculator requests perform an operation and create a new grid. For example, the Aspect request shown below takes an elevation grid theme, performs its calculation operations, creates a new theme of Aspect, and adds it to the view.

```
[Elevation].Aspect
```

There are several kinds of requests, including Arithmetic, Logarithmic, Trigonometric, and Powers. There are many other requests that do not appear in the Map Calculator, but which you can find in the online help.
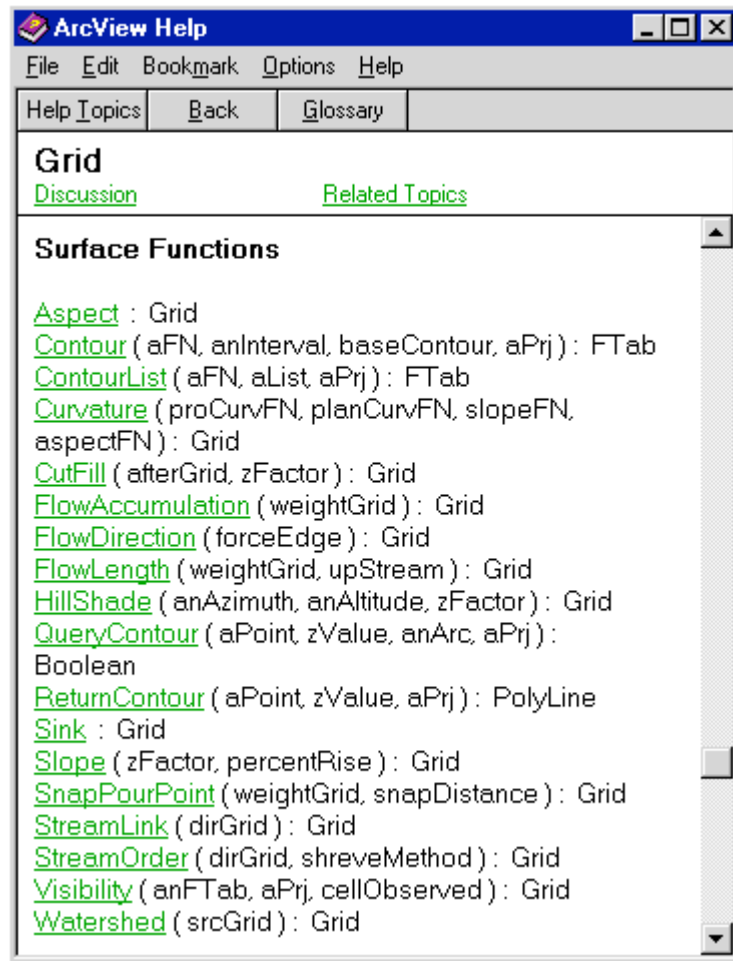
List of surface function requests in ArcView Help

Concept

Request rules

Requests are like commands; they perform an operation on an object. In an expression, the object and request are separated by a dot (period). Only requests that create a grid theme can be used in the Map Calculator.

`[Elevation].Aspect`--returns an aspect grid

Generally, requests perform their operation and return another object. Because requests may be nested, it is important to be sure that the final object returned from an expression is a grid.

`[Elevation].Hillshade ( ((35 + 270) / 2), 45, nil)`

Parameters are placed in parentheses and separated by commas if there are more than one. You will get an error if you leave out a required parameter. However, there will be times when you will use a placeholder when you want to use a default value or forego some output. The Avenue Nil object fulfills this purpose and indicates an absence of value.

Below is an example where nil is used to specify the default value. The third parameter for the Hillshade request is a number for the zFactor, or vertical exaggeration. By entering nil, this value defaults to 1.

```
[Elevation].Hillshade (315, 45, nil)
```

Requests like ZonalGeometryTable, Contour, ReturnCostPath, and AsPolygonFTab do not return grids. They cannot, therefore, be used in the Map Calculator.

Exercise

Find help for a request

The objective of this exercise is to learn how to access help for writing expressions and using Avenue requests. It is always a good idea to open the ArcView online help before you start writing Avenue requests in the Map Calculator.

If you have not downloaded the exercise data for this module, you should download the data now.

Step 1 Start ArcView

Start ArcView and load the Spatial Analyst extension.

Note: If you are running ArcView GIS 3.1, you see a Welcome to ArcView GIS dialog. Click Cancel to close this dialog.

If ArcView is already running, close any open projects.

Step 2 Open ArcView online help

From the Help menu, choose Help Topics.

The Help topics browser appears.

Step 3 Choose a topic

In the Help Topics browser, click the Index tab.

Where ArcView Help says "Type the first few letters of the word you are looking for," type **grid**.

Notice that Grid (Class) has been highlighted.

REVIEW CONCEPT

Step 4 View a topic

Click the Display button or double-click Grid (Class).

The help for the Grid class displays. Read a little about grids in the first few paragraphs, then scroll down to the section labeled Surface Functions. As you are scrolling, you will see many requests that can be used on grids. Any green text is a hypertext link that takes you to another help topic.

In the list of Surface Functions, locate the Aspect, HillShade, and Slope requests.

Look at the help topic for the Aspect request by clicking it. Read the help for Aspect and answer the following questions:

[Aspect questions](#)

Next, read more about Aspect by clicking on its Discussion. Read the discussion for Aspect and be sure you can answer the questions below.

[More Aspect questions](#)

In the Help window, click the Back button to display the syntax for Aspect. You will use the Aspect request in the next exercise.

Step 5 **Close help**

From the File menu, choose Exit to close the online help. Exit ArcView.

You have completed this exercise.

TOPIC2: Map algebra operators

Map algebra operators usually perform an action on two input grids or numbers. Most of the operators can be typed as algebraic symbols. Four types of operators may be used: arithmetic, Boolean, relational and bitwise.

- The arithmetic operators are basic math operators. When using arithmetic operators, if any input grid theme is a floating point grid, the output grid theme will also be floating point.
- The Boolean operators logically examine the input grids. The integer output grid will contain values of 0 (false) and 1 (true).
- Relational operators compare two numbers. If the result is true, 1 is returned; otherwise, 0 is returned. For example, a comparison of 10 < 20 would return 1, because 10 is less than 20.
- Bitwise operators perform operations on the binary representations of the input integer values (floating point values are truncated). The output is always integer. These operators are useful for manipulating multiple logical grids stored as bit positions in one physical grid and for evaluating other binary-encoded data.

  For more information on the bitwise operators, see the online help for the bitwise AND (&), the bitwise OR (|), the bitwise Complement LC (~), and the bitwise XOR (!) requests

Measurement scales

Before using any map algebra operators, you must understand measurement scales. The type of measurement system used may have a dramatic effect on the interpretation of the resulting values.

A distance of 20 kilometers is twice as far as 10 kilometers and something that weighs 100 pounds weighs one-third as much as something that is 300 pounds. But someone who came in first place may not have done three times as well as someone in third place, and soil with a pH of 3 is not half as acidic as soil that has a pH of 6.

To carry this even further, someone who is 60 years old is twice as old as someone who is 30 years old. But the older of the two individuals can only be twice as old as the younger individual just once in a lifetime. Also, if their birth dates are examined, and if the older individual was born in 1930 and the younger was born in 1960, the number 1930 is not twice the value of 1960.

The point to this discussion on numbers is that all numbers cannot be treated the same. The types of mathematical operations you can apply to your data depend upon how it was measured. There are four types of measurement scales: nominal, ordinal, interval, and ratio. The table below shows the operations that may be performed with each.

Scale	Example	Allowed Operations
Nominal	ID number, ZIP Code, phone number	=
Ordinal	Order or place	<, =, >
Interval	Time of day, years, temperature or pH level	<, =, >, +, -
Ratio	Age, distance, weight and volume	<, =, >, +, -, *, /

You cannot mathematically combine data that measures different things. For example, suppose you know that grading costs are somehow based on both the soil type (rock outcrops are harder to grade than sandy soils) and the slope of the land (tractors tip over on steep slopes). Even so, you cannot simply add a soil code to a slope measurement in degrees; the result would be meaningless.

Different measurement techniques result in different kinds of data. The technique used to make a measurement places the value on a nominal, ordinal, interval, or ratio scale. Each type of data is suitable for different kinds of analysis, and there are restrictions on the type of mathematical operations that may be applied against the data.

For example, it is meaningless to divide interval data like temperature in units of Fahrenheit or Celsius. Rather, temperature data would first have to be converted to Kelvin (ratio) in order to perform meaningful division.

The illustration below may help you better understand the four measurement scales.
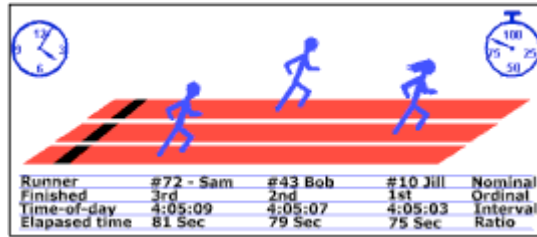
A foot race illustrates the measurement scales. [Click to enlarge]

Nominal measurements are not really measurements at all. They are simply a unique name assigned to some entity to allow identification, like the numbers the runners in a race wear on their shirts. No math is legal with nominal data (does it make sense to add two soil codes together and divide by two to get the average soil?). You can only test for equality.

Ordinal measures rank entities relative to one another, like the finish order of the runners in the example. Ordinal measures indicate only that one thing is more important than another, not how much more important it is. Again, no math is allowed, but you can test for relative value (e.g., value GT 3).

Interval measures are the magnitude of the difference between one value and another, but not between a value and zero, like the difference in time-of-day between two of the runners: Sam finished at 4:05:09 and Jill finished at 4:05:03, so Jill is six seconds faster. Without knowing when the race started, you cannot tell if the difference is big or small (was the race a sprint or a marathon?). Addition and subtraction are valid between interval values, but multiplication and division are not. However, interval values may be scaled by multiplying or dividing them by a constant.

Ratio measures are the magnitude of the difference between a value and a zero point, like the elapsed time for each runner measured with a stopwatch. All math is valid between ratio values, so you can not only tell the time difference between Jill and Sam (six seconds), but also how much faster Jill was ([81 sec - 75 sec] / 75 sec = .08, or 8% faster).

Concept

Using grids and numbers

Avenue requires that objects on either side of an operator (such as + or -) be of the same type. When entering numbers into the Map Calculator, you can then convert them into grids by clicking the AsGrid button, or by typing .AsGrid.

```
6.AsGrid + [theGrid]
```

Above, if you do not enter AsGrid, Avenue will interpret the "6" as a number object, instead of adding a grid object where all cells have the value of 6 to another grid object (theGrid).

There is an exception to this rule when a number follows the operator and a grid precedes the operator. For example: `[theGrid] + 6.`

Avenue will anticipate that the object after the plus sign is a grid and will convert the number to a grid object. To be safe, you should always click the AsGrid button or type the AsGrid request when you want a number to be evaluated as a grid.

When numbers are entered as parameters for requests, you do not need to use AsGrid. The following example illustrates this:

```
[Elevation].Slope (0.3048, true)
```

In this case, the number 0.3048 is a zFactor and indicates that for every unit of x,y there are 0.3048 units of z. The use of a zFactor is essential for correct slope calculations when the surface z units are expressed in units that are different from the ground units. DEMs often have x,y units in meters and z units in feet. In this case, 0.3048 is the conversion factor that specifies that there are 0.3048 meters in a foot.

The conversion problem explained above could be resolved by creating a grid where z is in the same units as x,y. To do this, use the following expression `[myGrid] * 0.3048`. Remember, in Avenue a period separates an object and a request. You will get a syntax error if you do not place a 0 before the decimal place.

Concept

Arithmetic operators

Arithmetic operators provide basic math functions such as multiplication, division, addition, subtraction, and raising to a power.
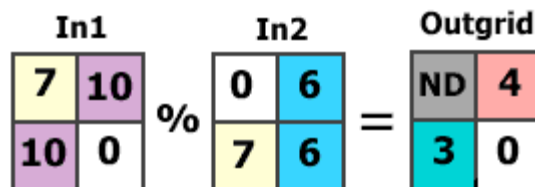
[In1] % [In2] = [Outgrid]

In this diagram, the modulus (%) request is being used to return the remainder of dividing the cell values in [In1] with those in [In2]. For example, 10 divided by 6 produces a remainder of 4.

The Modulus (%) request returns an integer representing the remainder when one integer is divided by another. This function can be used in an expression to determine whether the input grid is an even multiple. For example, the following expression creates an output grid where the cells are 0 if the input grid is evenly divisible by 10:

```
[inGrid] % 10.AsGrid
```

If floating point grids are inputs, use the FMod request:

```
[inGrid].FMod(10)
```

Multiplication is performed using the asterisk (*). Floating division is performed using the forward slash symbol (/). If all inputs are integer, the output is a truncated integer. However, if any of the inputs are a floating point, the output is a floating point.

Addition is performed using the plus sign (+). Subtraction is performed using unary minus. Unary minus flips the sign of all values for one grid theme (-aGrid) or subtracts one grid from another (aGrid - bGrid).

Power raises a grid to the power of another grid or number. Use either the Pow or (^) requests

Relational operators

The relational operators compare cell values in the input grids and return 0 or 1 to the output grid, depending on whether the comparison was false or true.
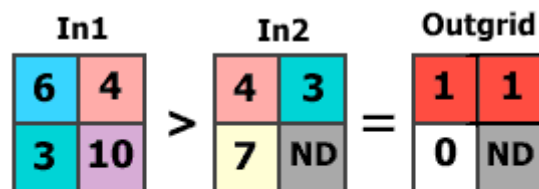
[In1] > [In2] = [Outgrid]

Relational operators compare values and return 1 (true) or 0 (false). In this diagram, cell values in [In1] that are greater than corresponding cell values in [In2] will return a value of 1 (true) in [Outgrid].

If you wanted to find out if a forest had gotten larger over time, you might write the following expression:

```
[Forest1980] < [Forest1990]
```

Increase in the forest size would be recorded in the output grid with values of 1 (if the forest got larger in 1990) and decreases in the forest would appear as value 0 (if the forest was larger in 1980).

The relational operator requests include:

Operator	Meaning
<	less than
< =	less than or equal to
< >	not equal to
=	equal to
>	greater than
> =	greater than or equal to

Boolean operators

Boolean operators check to see if cells have a non-zero value. A zero cell value returns a false (0), any other values return true (1). Any Boolean operation involving a No Data value will always return No Data.
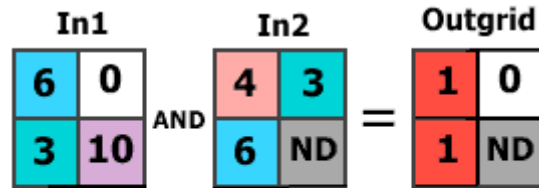
[In1] AND [In2] = [Outgrid]

Boolean operators check for non-zero values. In this example, corresponding cells in both [In1] and [In2] must be non-zero to produce a value of 1 (true) in [Outgrid].

Boolean operators include:

| Operator	Meaning
AND	Both sides must have a non-zero value to return true (1).
OR	One side must have a non-zero value to return true (1).
XOR	Either side, but not both, must have a non-zero value to return true (1).
NOT	If Boolean result is true, false (0) is returned. If Boolean result is false, true (1) is returned.

Concept

Conditional processing

Conditional processing is an important component of modeling. Conditional processing gives you the power to control the flow of operations. It permits you to specify the conditions that must be evaluated as true before an action should be taken and to specify the appropriate actions for false conditions.

The Con request is used to control the flow of operations based on the evaluation of one or more conditions. It is similar to a programmer's "If Then Else" statement in that you can perform multiple operations in a single Con request, although the final output will be only one grid. Dozens of temporary grids may be created and evaluated along the way. (Pick is another conditional processing request.)

Avenue syntax:

```
[aGrid].Con(yesGrid, noGrid)
```

For each cell, it returns the value found in yesGrid if aGrid is non-zero (true); otherwise, it returns the value found in noGrid.

In1

3	3	1
ND	0	0
3	2	1

Outgrid

1	1	2
ND	2	2
1	1	2
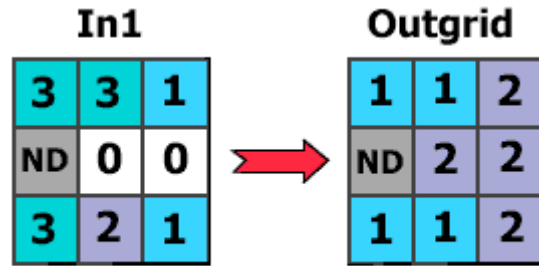
([In1] > 1.AsGrid).Con(1.AsGrid, 2.AsGrid)

Conditional processing is performed with the Con request. It is used to check a condition for true or false. In this example, values greater than 1 in [In1] are assigned a value of 1 in [Outgrid]. Else, a value of 2 is assigned.

The example below is a statement for example purposes only. If Then Else statements do not work in the Map Calculator and cannot evaluate at the individual cell level.

```
If (values in in1 are greater than 1) Then
    Calculate the output value to 1
Else
    Calculate the output value to 2
End
```

In some situations, you can nest conditions within other conditions. Instead of a grid to handle false conditions, you can add another entire Con request.

```
([in1] = 2.AsGrid). Con (1.AsGrid, ([in1] = 3.AsGrid).Con (2.AsGrid,
3.As Grid) )
```

```
If (value = 2) Then
    Calculate the output value to 1
Else If (value = 3) Then
    Calculate the output value to 2
Else
    Calculate the output value to 3
End
```

Concept

Requests or menu options?

How do you decide whether to use the Spatial Analyst menu options or Avenue requests to perform analysis?

In the beginning of this lesson, you learned how important Avenue requests are to ArcView Spatial Analyst. Each of the Analysis and Surface menu options has a built-in Avenue program called a script. Each time you click one of those menu options, its script runs. Inside the script are the corresponding Avenue requests.

For example, the Derive Aspect option runs a script (Spatial.Aspect) that contains Avenue statements including the Aspect request. The script does much more than just the Aspect request: it renames the theme and creates a legend with special colors, classification, and labels.

It is quite simple to use some of these Avenue requests to perform analysis from the Map Calculator. Using requests in the Map Calculator gives you more flexibility, as you can use parameters not available for a menu option. For example, the Hillshade request allows you to specify a zFactor to control vertical exaggeration, a parameter not available using the Compute Hillshade menu option.

It is also important to mention again that a majority of ArcView Spatial Analyst functionality is not available from the menus. Rather, the additional functionality is available through Avenue requests.
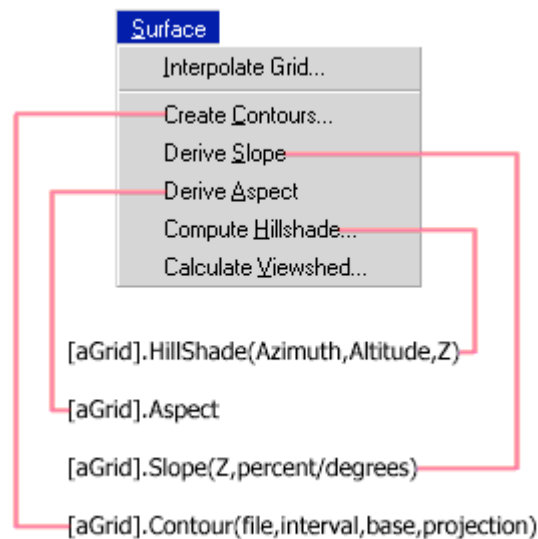
Menu options carry out requests. Using requests outside the menu offers more functionality and flexibility. Menu options are quick to access, however.

Exercise

Use Avenue requests with the Map Calculator

The objective of this exercise is to use the Map Calculator to enter Avenue requests and map algebra expressions.

If you have not downloaded the exercise data for this module, you should download the data now.

Step 1 Start ArcView

Start ArcView and load the Spatial Analyst extension.

Note: If you are running ArcView GIS 3.1, you see a Welcome to ArcView GIS dialog.
Click Cancel to close this dialog.

If ArcView is already running, close any open projects.

Step 2 **Open a project**

From the File menu, choose Open Project. Navigate to the **mapalsa\lesson2**
directory and open the project **l2_ex02.apr**.

Note: If you are running ArcView GIS 3.1, you see an Update l2_ex02.apr message
box. Click No to dismiss this box.

When the project opens, you see the Elevation Analysis view containing an elevation
theme.

VIEW RESULT

Step 3 **Prepare to use Avenue requests**

The Map Calculator can be used to issue Avenue requests. There are six steps you
should perform each time you use the Map Calculator.

1. Determine which request to use.
2. Look at the online help for that request to understand how it works.
3. Open the Map Calculator and create the expression.
4. Check the expression for errors.
5. Evaluate the expression.
6. Rename the new theme.

First, you'll determine which request to use. You have the Elevation Analysis view
open and you will perform analysis operations on the aspect, slope, and hill locations
of the Elevation grid theme.

Start by looking in online help for grid themes. Locate the Aspect, HillShade, and
Slope requests.

Step 4 **Open the Map Calculator**

Make the view active, then from the Analysis menu, choose Map Calculator.

Step 5 **Enter a simple request**

When the Map Calculator opens, build an expression by double-clicking Elevation in
the Layers list, then type a dot (**.**) and the word **Aspect**. Place the expression inside
parentheses. Your expression should look like this:

([Elevation].Aspect)

When you have finished creating your expression, check it for correctness by
comparing it to the example in the online help window. It is a good practice to double-
check your expression with the online help syntax. The online help also provides an
explanation of the request and its parameters. Even a veteran Spatial Analyst user
should refer to the online help when typing an expression in the Map Calculator.

Before evaluating the expression, make sure that you can answer yes to the questions in the following checklist:

1. Is the grid theme name surrounded by brackets ([ ])?
2. Is there a dot between the grid and the request?
3. Does the request have all its parameters?
4. Are the parentheses all correct and matching?

If you answered yes to all the questions, click Evaluate.

Did you get a Syntax Error message?

If you got an error, that's OK. Everyone makes mistakes.

Click OK to close the error message box and check your expression again for correctness.

Make any necessary corrections and click Evaluate.

If no errors appear (good job!), close the Map Calculator.

In the view, you see a new theme called Map Calculation 1. As you have probably noticed by now, the Map Calculator names its themes and numbers them sequentially (i.e., Map Calculation 1 through Map Calculation n).

The new theme contains floating point values that represent Aspect locations. The Aspect request creates the same aspect theme as the Derive Aspect option from the Analysis menu, but it does not name the theme or classify the legend.

Change the name of this theme to **Aspect** and turn it on. Turn off the Elevation theme.

VIEW RESULT

In the next step, you will use another request. This time, you'll need to specify some parameters.

Step 6 Enter a request with parameters

Next, you will use the Slope request, which has some parameters. Begin by searching online help for the Slope request.

In the help topics Index, type in the key word **Slope**.

In the list of Index entries, double-click Slope (request) to jump to the help topic for Slope.

Read the help topic for Slope.

Notice that the Slope request has two parameters, zFactor and PercentRise. The Slope request is used on a Grid object and it returns a Grid.

Next, examine the Discussion for Slope, then minimize the Help window.

Now that you know something about the Slope request, you will try it out.

From the Analysis menu, choose Map Calculator.

When the Map Calculator opens, build an expression: double-click on the Elevation theme in the Layers list, then type a dot (**.**) and **Slope**.

Now add the Slope parameters, NIL and TRUE, inside a pair of parentheses and separated by a comma. Your expression should look like the following:

([Elevation].Slope(NIL, TRUE))

When you have finished typing the expression, check it for correctness by comparing it to the syntax in the Slope help topic and the following checklist.

1. Is the grid surrounded by brackets ([ ])?
2. Is there a dot between the grid and the request?
3. Are the parentheses all correct and matching?
4. Are the parameters separated by commas?
5. Do the parameters have their correct value?

If a parameter requires a Boolean (true or false) you cannot enter a number such as 0, you must type in "true" or "false."

If you answered yes to all the questions, click Evaluate.

If you get a syntax error, check your expression, correct it, and evaluate it again.

If no errors occurred, close the Map Calculator.

In the view, you see the new theme called Map Calculation 1. Change the name of this theme to **Slope** and turn it on. Turn off the Aspect theme.

VIEW RESULT

Step 7 **Close the project**

If you want to do the Challenge, go there now. Otherwise, close the project without saving any changes. You have completed this exercise

Challenge:
Use online help to enter parameters for the Hillshade request

In this exercise, you learned how to enter requests into the Map Calculator. You will now use the HillShade request. This time, however, you are on your own. You will only receive a brief set of instructions.

Refer to the online help if you need help building the correct expression.

1. Open the Map Calculator and create an expression that will hillshade the Elevation theme. The imaginary light source should have an azimuth of 315, an

altitude of 45, and a Z factor of 2. (You cannot enter a Z factor when using the HillShade option from the Analysis menu.)

2. Evaluate the expression.
3. Change the name of the new theme to HillShade and turn it on.
4. Change the HillShade theme colors and legend classification.
   - Double-click the HillShade theme to open the Legend Editor.
   - Change the classification method to EqualArea with nine classes.
   - Change the colors by first double-clicking on the symbol for the first class and setting its color to Black.
   - Next, click the last class (not the No Data class) and set its color to White.
   - Click the Ramp button ⊞↓.
   - Click Apply to see the changes to the Hillshade theme.

Notice that the imaginary light source is casting a shadow to the lower right or bottom of the hills.

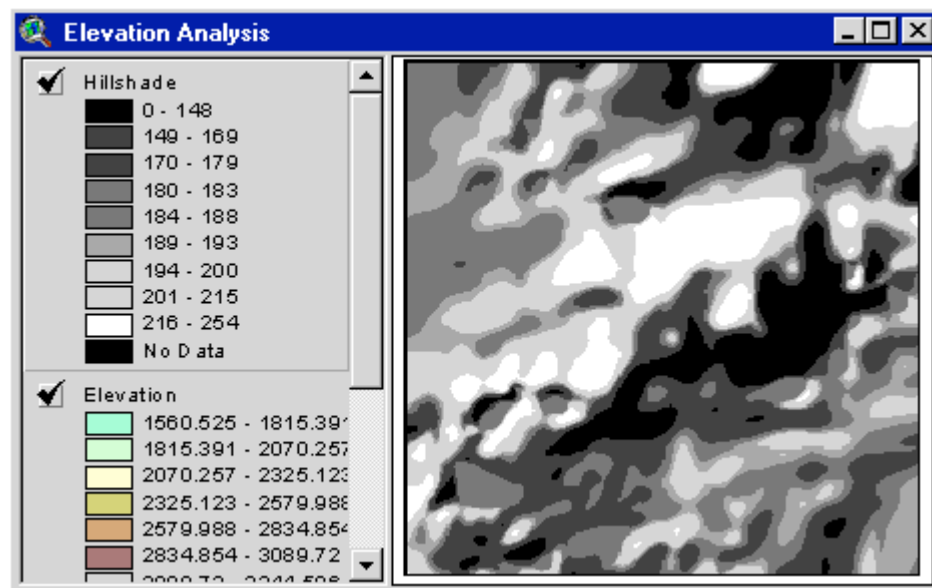Increasing the Z factor exaggerates the height of the hills.

5. Close the Elevation Analysis view and the Legend Editor

Solution to challenge

The expression you entered into the Map Calculator should look like this:

[Elevation].Hillshade(315,45,2)

The HillShade grid theme should look like the one below.

Exercise

Use arithmetic operators

The goal of this exercise is to how learn to use arithmetic operators. Arithmetic operators are usually self-explanatory. Addition, subtraction, division, and multiplication behave exactly as you would expect. The modulus operation (%) may not be so familiar. Modulus returns the remainder for integer division. For example, 3 goes into 20 six times, with 2 left over. The modulus comparing 20 and 3 would return a value of 2.

The modulus is useful with cyclical data, like compass directions which start at 0, increase to 360, then start over again at 0. For example, if you add 50 degrees to 330 degrees, the resulting direction should be 20 degrees, not 380 degrees.

In this exercise, you will examine a theme of wind direction and create a map algebra expression to simulate a 90-degree change in the wind direction.

If you have not downloaded the exercise data for this module, you should download the data now.

Step 1 **Start ArcView**

If necessary, start ArcView and load the Spatial Analyst extension.

Note: If you are running ArcView GIS 3.1, you see a Welcome to ArcView GIS dialog. Click Cancel to close this dialog.

If ArcView is already running, close any open projects.

Step 2 **Open the project**

From the File menu, choose Open Project. Navigate to the **mapalsa\lesson2** directory and open the project **l2_ex03.apr**.

Note: If you are running ArcView GIS 3.1, you see an Update l2_ex03.apr message box. Click No to dismiss this box.

When the project opens, you see a Wind Study view, which contains a theme of wind directions.

VIEW RESULT

Step 3 **Load the CellTools sample extension**

This exercise requires the Celltools sample extension.

To load the CellTools sample extension in ArcView, make the Project window active, then choose Extensions from the File menu. The dialog that appears lists all the available extensions. Check the box next to Cell Tools (sample).

If Cell Tools (sample) is not listed in the extensions dialog, you will need to copy the celltool.avx file from ArcView's samples\ext directory to ArcView's ext32 directory. These directories are located in ArcView's installation directory. A typical location (the default) for the installation directory is c:\esri\av_gis30\arcview. If you cannot find this path, search for the file arcview.exe to find the correct path.

Once you've done this, the Cell Tools sample extension will be listed in the Extensions dialog the next time you open it. In that dialog, check the box next to the Cell Tools sample extension and click OK.

If you don't want this sample extension to appear in the Extensions dialog permanently, you can load it temporarily by running a simple script. It will only be available for your current ArcView session.

To run this sample script in ArcView perform the following steps:

1. In ArcView's Project window, click the Scripts icon, then press New to create a new script.
2. Type the following line into the script window:

   ```
   System.SetEnvVar( "USEREXT",
   "$AVHOME/samples/ext".AsFilename.GetFullname)
   ```

3. Compile and run the script.

Once you've done this, the sample extension will be listed in the Extensions dialog the next time you open it. In the Extensions dialog, check the box next to the Cell Tools sample extension and click OK.

Step 4 **Use the CellArrow tool**

Click the CellArrow tool and select a few cells to see which direction the wind is blowing.

Click No when prompted to save arrow graphics.

VIEW RESULT

Step 5 **Use the Map Calculator**

From the Analysis menu, choose Map Calculator. Write an expression to add (+) 90 degrees to the directions in the Windgrid theme and adjust the result with the % (modulus) request shown as follows.

([Windgrid] + 90) % 360

Click Evaluate, then close the Map Calculation 1 dialog.

Change the name of the new theme to **Wind90**, turn it on, and make it active. Turn off the Windgrid theme.

VIEW RESULT

REVIEW CONCEPT

Step 6 **Use the CellArrow tool again**

Use the CellArrow tool again and select some of the same cells as you did in Step 4.

Compare the wind (arrow) directions. Notice how the wind has shifted 90 degrees.

VIEW RESULT

The cell values in Windgrid ranged from 293–357. Adding 90 to those values would result in new values ranging from 383–447. The compass direction values should not exceed 360 degrees.

By using the modulus of 360, the values in Wind90 are calculated and range from 23–87, the result of a 90-degree change in wind direction.

Step 7 **Close the project**

Close the project without saving any changes.

You have completed this exercise

Exercise

Use relational and Boolean operators

The relational and Boolean operators are used to test or compare different sets of values. They always return a grid theme that contains 1 (true) or 0 (false) values. In this exercise, you will use the relational and Boolean operators to write a simple model to find agricultural areas with elevations over 1500 feet.

If you have not downloaded the exercise data for this module, you should download the data now.

Step 1 **Start ArcView**

If necessary, start ArcView and load the Spatial Analyst extension.

Note: If you are running ArcView GIS 3.1, you see a Welcome to ArcView GIS dialog. Click Cancel to close this dialog.

If ArcView is already running, close any open projects.

Step 2 **Open the project**

From the File menu, choose Open Project. Navigate to the **mapalsa\lesson2** directory and open the project **l2_ex04.apr**.

Note: If you are running ArcView GIS 3.1, you see an Update l2_ex04.apr message box. Click No to dismiss this box.

When the project opens, you see a Farm Study view containing an Elevation theme and a General Plan theme.

VIEW RESULT

Step 3 **Use the Map Calculator with the = operator**

First, you will create a theme of only agricultural areas.

Turn off the Elevation theme. You can now see General Plan.

From the Analysis menu, choose Map Calculator and enter an expression to test for General Plan values that are equal to 100 (agricultural areas). The output grid theme will contain values of 0 and 1. Your expression should look like this:

[General Plan] = 100

Click Evaluate and close the Map Calculator. Change the name of the new theme to **Ag Grid** and turn it on. Turn off General Plan.

VIEW RESULT

The agricultural area cells have a value of 1 and all other cells have a value of 0.

REVIEW CONCEPT

Step 4 **Use the Map Calculator with the > operator**

Next, from the Analysis menu, choose Map Calculator and enter an expression to find elevation areas that are greater than 1500 feet. Your expression should look like this:

[Elevation] > 1500

Click Evaluate and close the Map Calculator.

Change the name of the new theme to **High Grid** and turn it on. Turn off Ag Grid.

VIEW RESULT

Step 5 **Use the Map Calculator with the AND operator**

Finally, you will put the two grids together using the AND operator. AND compares the two Boolean values (true and false). If both are true, a value of 1 (true) is returned. Otherwise, a value of 0 (false) is returned.

Your model statement will be: If Ag Grid is true and High Grid is true, then return a true.

From the Analysis menu, choose Map Calculator and enter an expression to find Ag Grid areas and High Grid areas.

[Ag Grid] AND [High Grid]

Click Evaluate and close the Map Calculator.

Change the name of the new theme to **Potential Farm Areas** and turn it on. Turn the other themes off.

VIEW RESULT

The final farm theme shows true (value of 1) for agricultural areas that are above 1500 feet of elevation.

Your model used three statements.

[General Plan] = 100
[Elevation] > 1500
[Ag Grid] AND [High Grid]

You could have used one larger statement.

([General Plan] = 100) AND ([Elevation] > 1500)

Sometimes models consider dozens of issues, instead of just two. It is much easier to diagram the process and break the model up into its smallest components or goals. One advantage to performing the three separate steps is that you can see what is happening when each criterion is applied. During these steps you may discover something that was not apparent before.

REVIEW CONCEPT

Step 6 **Close the project**

Close the project without saving any changes.

You have completed this exercise

Exercise

Use conditional processing

In this exercise, a new subdivision has been added in a city. The new subdivision has a new land use code. You will use the Con request to replace old General Plan land use values with the new land use values.

Step 1 Start ArcView

If necessary, start ArcView and load the Spatial Analyst extension.

Note: If you are running ArcView GIS 3.1, you see a Welcome to ArcView GIS dialog. Click Cancel to close this dialog.

If ArcView is already running, close any open projects.

Step 2 Open the project

From the File menu, choose Open Project. Navigate to the **mapalsa\lesson2** directory and open the project **l2_ex05.apr**.

Note: If you are running ArcView GIS 3.1, you see an Update l2_ex05.apr message box. Click No to dismiss this box.

When the project opens, you see a Land Study view. The view contains a theme of general land use (General Plan) and a second theme containing a tract of land that will become a new subdivision (New Tract).

VIEW RESULT

Step 3 Use the Identify tool

Turn on the New Tract theme.

The land use code for the new tract is 630 (high-density residential). You will identify some of the current land use in the General Plan theme.

With the General Plan theme active, click the Identify tool, then click on some cells inside the New Tract area (red).

You should find the land use codes are 100 (Agriculture), 300 (Industrial), 700 (Recreational), and 610 (Res-Low Density).

Step 4 Use the Con request

Next, you will use the Con request to update the General Plan with the new land use value. The New Tract theme has values of 0 wherever the General Plan values should be retained. Elsewhere, New Tract has the new land use value of 630 (high-density residential) for the new subdivision.

Open the Map Calculator and use the Con request to check conditions in the New

Tract theme. Wherever the New Tract theme has values, keep them; otherwise (if the value is 0) assign output values from the General Plan theme.

[New Tract].Con ([New Tract], [General Plan])

Change the name of the new theme to **New General Plan** and turn it on. Turn the other themes off.

VIEW RESULT

The New General Plan was created by using the New Tract as a patch over the old General Plan. The new theme contains all the old general plan values, except the new tract has replaced old general plan land use values with 630 (high-density residential).

REVIEW CONCEPT

Step 5 **Compare differences in themes**

Next, you will compare the two general plans to determine where they are different. You have two datasets that reflect the same location over time. You may want to see what changes have taken place between the old general plan and the new general plan.

Use the Map Calculator to compare the General Plan and the New General Plan themes. For cells with differences, use the Con request to output the old General Plan value. If there are no differences, output a 0 (zero) value.

([General Plan] - [New General Plan]) .Con ([General Plan], 0.AsGrid)

Turn on the new theme and turn off the New General Plan.

VIEW RESULT

Notice that the new subdivision has replaced four land use values with 630, the high-density residential.

Step 6 **Close the project**

Close the project without saving any changes.

You have completed this exercise

Summary

In this lesson, you learned that map algebra uses math-like expressions that perform spatial analysis on grid themes. These expressions are entered into the Map Calculator and result in new grid themes. The Map Calculator includes operators that perform numerous arithmetic, logarithmic, trigonometric, and power functions. These operators are also considered Avenue requests.

There are rules that must be followed for writing expressions in the Map Calculator. For example: Grid names are in brackets. Strings use the double quote as a delimiter. Strings are converted to filenames with the AsFileName request. Expressions are evaluated from left to right. Parentheses determine the order of evaluation.

Many of the more popular ArcView Spatial Analyst functions are available as menu options, but a vast majority of ArcView Spatial Analyst functions are accessed using Avenue requests. ArcView's online help contains a description of each request, including its syntax and examples.

There are also rules for requests. A dot separates objects and requests. Parameters are separated by commas and are in parentheses. Request parameters may be nested. Only requests that create a grid theme can be used in the Map Calculator.

Relational operators compare values in grids and return values of true (1) or false (0). Boolean operators check for non-zero values and return values of true (1) or false (0). Conditional processing is performed with the Con request and checks a condition for true or false

This is the **Using Map Algebra in ArcView Spatial Analyst - Lesson 2 Self test**.
⏲ **Please watch your time—you have 2 hours to complete this test.**

Use the knowledge you have gained in *Using Map Algebra in ArcView Spatial Analyst* to answer the following questions. You will need to correctly answer 7 of the following questions to pass.

Netscape Users: Do not resize this browser window. This can cause the page to reload and generate new questions.

GOOD LUCK!

---

1. When numbers are entered as parameters for requests, you must use the AsGrid request.

   ○ True

   ○ False

2. Mathematical operations including arithmetic, logarithmic, trigonometric, and powers are all considered to be Avenue requests.

   ○ True

   ○ False

3. Objects that can be used in the Map Calculator include grids, strings, and numbers.

   ○ True

   ○ False

4. Boolean operators return 1 if a cell has a non-zero value and return No Data if a cell has a zero value.

   ○ True

   ○ False

5. Which measurement scale is temperature in units of Centigrade an example of?

- ○ Nominal
- ○ Ratio
- ○ Ordinal
- ○ Interval

6. Which of the following requests returns an integer representing the remainder when one integer is divided by another?

- ○ %
- ○ +
- ○ /
- ○ *

7. You can open the Map Calculator by selecting Map Calculator from the Surface menu.

- ○ True
- ○ False

8. The magnitude of the difference between a value and a zero point is a:

- ○ Nominal measure
- ○ Ordinal measure
- ○ Ratio measure
- ○ Interval measure

9. When you want to use a default value for a request parameter, use:

- ○ The pound sign (#)
- ○ A blank space separated by commas
- ○ An asterisk (*)
- ○ The Avenue Nil object

10. The AsGrid request does not have to be used when a number follows the operator and a grid precedes the operator.

- ○ True
- ○ False

Calculate My Grade