

## Introduction

Spiral is a little program showing the kind of spirals that appear in the centre of composite flowers (they're akin to the spirals on pineapples, pine cones, strawberries etc.), as well as many other spiral patterns. I wrote it originally for a friend and for myself, to illustrate the mathematics involved, but it's so nice to play with that i want to share it with everyone.

You do not need to know any math to **enjoy** the myriads of striking patterns the program generates - just click on the sliders, and watch the pictures. Have a look through the menu options too: you can change the item shapes, switch spirals on and off, or change them. You can change the size of the window (just like with other Windows apps) and the inside will adapt.

But if you get **curious** about the math behind it all, this help file will try and answer some questions. And, as often the case, you may find that understanding adds to the enjoyment.

### How to use Spiral

[The  \$x\$  slider](#)

[The  \$yi\$  slider](#)

[Why a !\[\]\(003082e50e3009141f59bd5df831749f\_img.jpg\) button?](#)

[Additional mouse behavior](#)

[Menu overview](#)

[The hi-lite dialog box](#)

### What's behind it

[Spiral basics](#)

[Metaspirals](#)

### Spiral the program

[Program history](#)

[Distribution policy](#)

[Hardware considerations](#)

## The $x$ slider

The pattern you see is affected by two numbers that Spiral calls  $x$  and  $yi$ . The number  $x$  affects the **spacing in the radial direction**: if  $x$  gets bigger the pattern is stretched, if  $x$  gets smaller it is squashed.

This stretching/squashing is **not** simply scaling up/down (zooming in/out): see [Spiral basics](#).

The same amount of change in  $x$  has a lot more effect when  $x$  is small. To ensure the same amount of visual impact per click at all levels, this slider has been made to work logarithmically, which means that the change in  $x$  is always proportional to the value of  $x$  at the time. Here a click on the up or down arrow button will increase or decrease  $x$  by about 1/30 part of its present value.

Spiral follows a convention among many Windows programs where a click in the area between the movable button and the up or down button is worth ten clicks on the up or down button itself.

You can also drag the movable button directly, this causes the numbers displayed to be updated while you drag, but redrawing does not start until you let go of the mouse button.

There is an arbitrary **maximum**  $x$  of just under 0.4 (by which point the display has become too boring - anyway we have to stop somewhere) and an arbitrary **minimum**  $x$  of just under 0.0004 (beyond which the display would soon fill up with "ink", redraw times become prohibitive, and the program's fast integer math would come apart at the seams).

Compare [the  \$yi\$  slider](#), or back to the [introduction](#)?

## The $y_i$ slider

The pattern you see is affected by two numbers that Spiral calls  $x$  and  $y_i$ . The number  $y_i$  (that's  $y$  times  $\sqrt{-1}$ ) affects the **angles** in the picture: if  $y$  gets bigger things move anticlockwise, if  $y$  gets smaller clockwise.

This movement is **not** simply a matter of rotating the whole picture: see [Spiral basics](#).

The same amount of change in  $y_i$  has a lot more effect when  $x$  (that's right,  $x$ ) is small. To ensure the same amount of visual impact per click at all levels, the  $y_i$  slider resolution has been made dependent on the value of  $x$  at the time. A click on the up or down arrow button will increase or decrease  $y$  by about  $1/30$  part of the present value of  $x$ .

Spiral follows a convention among many Windows programs where a click in the area between the movable button and the up or down button is worth ten clicks on the up or down button itself.

You can also drag the movable button directly, this causes the numbers displayed to be updated while you drag, but redrawing does not start until you let go of the mouse button. Values selected in this way are simply proportional to the position along the slider (independently of  $x$ ) and are whole numbers of minutes of arc. Even just clicking on the movable button will **round** your angle to the nearest minute.

This slider is **circular**, it will wrap around past 0 and  $2\pi$  (haven't yet managed to code a scrollbar that is circle-shaped <g>).

[The Reset  \$y\_i\$  button](#) resets  $y_i$  to its default value of  $2\pi / (\frac{1}{2} + \frac{1}{2}\sqrt{5})$ .

Compare [the  \$x\$  slider](#), or back to the [introduction](#)?

## The Reset $y_i$ button

Pressing this button is a quick way to restore  $y_i$  to its default value of  $2\pi / (\frac{1}{2} + \frac{1}{2}\sqrt{5})$ .

In the growth tips of plants, tiny "primordia" are laid out one after the other that will turn into leaves, scales, petals, florets etc. A very common pattern is the kind of spiral where each successive primordium appears at an angle of  $2\pi / (\frac{1}{2} + \frac{1}{2}\sqrt{5})$  radians relative to the previous one. You'll find this in pine cones, and in many flowers and the resulting fruits, strawberries, pineapples, artichokes, and many more. Nature has a good reason to do this, as it generates the most efficiently packed spiral pattern possible.

At this angle, the metaspirals you'll see are all of Fibonacci number order. If you want to see spiral patterns as they occur in Nature, always keep  $y_i$  at this value, and vary only  $x$ .

For a certain range of values of  $x$  we see metaspirals of order 5 and 8 (as in pine cones), for smaller values of  $x$  the orders are 8 and 13 (as in pineapples), and for even smaller  $x$  higher Fibonacci numbers become noticeable, 21, 34, 55, 89... (as in the centre of various composite flowers such as sunflowers).

## Additional mouse behavior

Moving the mouse around in Spiral, you will see that whenever it moves over an item (circle or dot etc.), the cursor changes to incorporate a **number**. This is the sequential number of the item, the  $k$  in  $exp\ kz$  (see [Spiral basics](#)).

When you **click left** the cursor will jump to the next item, and when you **click right** it moves to the previous item. You can click repeatedly, to give you an idea of the single spiral actually underlying the whole [metaspiral](#) structure.

If you have chosen an  $x$  smaller than 0.000500, there can be more than 10,000 items on screen (!) but there is only really space in the cursor for the last four digits. This shouldn't worry you too much, as these numbers are to a large extent arbitrary anyway --- only the difference between two of them has an absolute meaning. Look on it as a mile or km counter, so 0000 is one more than 9999.

To **specks** and **dots** the cursor is sensitive within a few pixels on either side. This is a design compromise so, on the one hand, it's not too hard to pick an item out with the mouse, and on the other hand you still have some resolution in the crowded area near the centre. With **circles**, you can put the cursor anywhere inside, or on the edge. With **squares** and **hexagons**, the target area bulges slightly beyond the sides (it's actually still circular).

## **Menu overview**

### **Fibonacci menu:**

Hi-lites...

Exit

### **Shapes menu:**

Specks (fast)

Dots

Squares

Hexagons

Circles (nice)

### **Help menu:**

Introduction

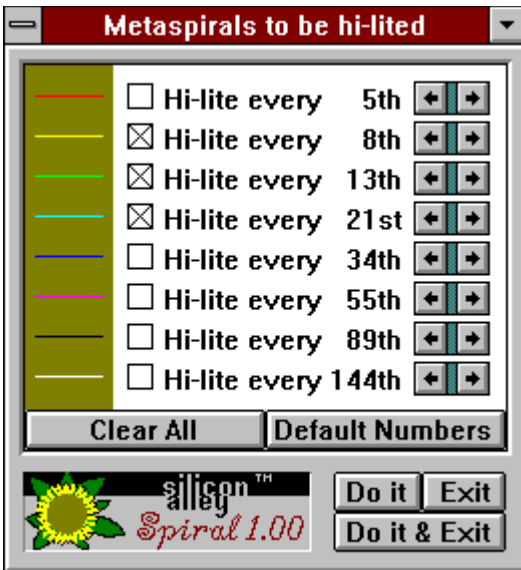
Search for Help on...

How to Use Help

About Spiral...

## The hi-lite dialog box

Choosing **Hi-lites...** from the **Fibonacci** menu opens a dialog box. You can click on the various elements in the picture here now to see what they're for.



To check whether a metaspiral family you see is of order  $n$ , where  $n$  can be any number between 1 and 255 inclusive, you can hi-lite a metaspiral of the family of order  $n$  in a contrasting color.

On startup, the eight Fibonacci numbers between 5 and 144 inclusive are already associated with eight colors. You can select one or more of these. You can also change any of the numbers to any value from 1 to 255.

If you have left  $y$  at its default value of  $2\pi/\varphi$  (or used the button) you can expect metaspirals of a Fibonacci number order. With other values of  $y$  you can get metaspirals of any order.

Hi-liting is also useful to create a feature that serves as an anchor for your eyes when you tweak  $x$  and/or  $y$  in the main window, so you can see what goes where (tip: keep at least two boxes checked for this).

Back to the [introduction](#) or to the [menu overview](#)?

Standard **system menu** allowing you to **Move**, **Minimize** (i.e. reduce to an icon), or **Close** (exit) the dialog box, or to **Switch To** another application via the Task Manager (or its replacement).

Double-clicking this field will close the box immediately. Moving is easier done by dragging the whole box by its Title Bar (to the right of this field), and minimizing can be done by clicking the Minimize Button (top right corner).



You can position the whole dialog box anywhere on the desktop by dragging it by this **Title Bar**.

You can reduce the dialog box to an icon by clicking this **Minimize Box**. Later you can double-click the icon (or choose Hi-lites... from the Fibonacci menu again) to restore the dialog box.

Nothing will happen if you click here in the dialog box. These lines show the eight colors that will be used for hi-lighting if you select the corresponding check boxes.

Clicking on a **check box** (or on the **text** to the right of it) checks (puts an X in) the box, or clears it if already checked. Checking a box causes a corresponding metaspiral to be hi-lited in the main window once you press the Do it or the Do it & Exit button.

The eight boxes here can be checked or cleared independently of each other, hi-liting more than one metaspiral if you so desire, in the eight colors indicated here. Where metaspirals cross, colors further down the box will take precedence.

The eight **mini-scrollbar**s (appearing as pairs of ← and → buttons) will lower and raise the corresponding numbers, to show different metaspiral families. If you have kept  $y_i$  at its default value, you won't need to change these numbers.

The **Clear All** button is just a quick shortcut to clear all eight check boxes simultaneously.

The **Default Numbers** button will reset all eight numbers to their original Fibonacci values. If you have kept  $y_i$  at its default value, you'll only need these numbers.

Merely a decorative logo! As you can see, this snapshot of the dialog box was taken from the previous version. The current one says *Spiral 1.10*.



The **Do it** button uses your changed settings to redraw the main window, but does not close the hi-lite box; it stays on screen ready for use. You can work with the main window while leaving the hi-lite box displayed.

You can make a number of changes in the settings without annoying redraw delays in between, because they do not take effect until you click on Do it or on Do it & Exit.

The **Do it & Exit** button (similar to an OK button) closes the hi-lite box, using your changed settings to redraw the main window.

You can make a number of changes in the settings without annoying redraw delays in between, because they do not take effect until you click on Do it or on Do it & Exit.

The **Exit** button (similar to a Cancel button) closes the dialog box without saving your changes.

## What does Exit do?

Guess! <g>

Back to the [introduction](#) or to the [menu overview](#)?

## Changing item shapes

From the **Shapes** menu, you can choose between

- **Specks** of one pixel (which plot quickest but are hard to see).
- **Dots** of 4 pixels (which plot almost as quick).
- **Squares** that become bigger as you go outward (they look good when the two most conspicuous metaspiral families are at right angles).
- **Hexagons** that become bigger as you go outward (they look good when there are three prominent metaspiral families).
- **Circles** that become bigger as you go outward (a good choice to see any metaspiral pattern there is).

There is always one of these menu items checked.

Back to the [introduction](#) or to the [menu overview](#)?

## Windows Help

From the **Help** menu,

**Introduction** opens the Help file you are reading now, at its first page (just like the **Contents** button here does).

**Search for Help on...** opens a dialog box to search this Help file (just like the **Search** button here does).

**How to Use Help** opens another Help file giving help on the Windows Help system itself (this option is available in the **Help** menu here too).

Back to the [introduction](#) or to the [menu overview](#)?

## About Box

**About Spiral...** from the **Help** menu opens the customary About box with the version number, and our address.

Back to the [introduction](#) or to the [menu overview](#)?

## Hardware considerations

### Processor

If your system runs Windows, it should run Spiral. Please [let me know](#) if this is not the case.

### Memory

Spiral's requirements depend on the size of its window. If it has the size it starts up with in 800x600 16-color mode (its startup size depends on your video mode) it uses 305 K, but Windows may require temporarily some more memory to actually start it up. While Spiral is minimized, it only takes up 119 K (of which 96 K is fixed memory from the global heap). In addition it uses a modest amount of GDI and User heap.

If there is not enough memory to start Spiral up, it will not run, but display a message inviting you to close down some other apps and try again.

If other apps have exhausted memory after Spiral started up (for instance while Spiral was minimized) it may present you with a blank window when you restore it, like many Windows apps do under those circumstances. The solution is to minimize it again, and choose to close either Spiral or some other apps.

### Video hardware

A video mode is called **isotropic** if the number of pixels per inch is the same horizontally as vertically. With an ordinary monitor, with a height that's 3/4 of its width, any of the usual 640x480, 800x600, 1024x768, 1280x960 etc. modes are isotropic. On a portrait monitor, 768x1024 would be isotropic. Modes that are **anisotropic** include 640x350 (EGA) and 1280x1024.

Some effort has gone into making sure that the main window display is still circular, and the items have the right shape, in anisotropic modes. But Spiral takes advantage of isotropic modes. Because it does not have to treat horizontal and vertical distances separately, it can calculate the dimensions of its circles, squares and hexagons more quickly.

If you have a choice of modes, you may find Spiral slightly faster in isotropic ones. On the other hand, the difference may be swamped by some modes being simply faster than others on your video card.

Back to the [introduction](#) or to the [menu overview](#)?



## Spiral basics

The program paints its items (circles, dots, etc.) one by one, from the inside out. The first item (item number 0) is placed at an arbitrary position (actually 2 pixels to the right of the centre). Each next item is placed a little bit further from the centre than the previous one, the distance grows by the same factor each time. This "radial growth factor" is displayed in the status bar at the bottom. At the same time, each item is also placed in a different direction from the centre, each time we turn by an angle which is also displayed in the status bar.

**The number  $y$  that you change with the right slider** is this angle expressed in radians; a full circle ( $360^\circ$ ) is equivalent to  $2\pi$  radians (which is 6.283185).

Changing  $y$  does not just rotate the whole picture. The effect is cumulative, which you can see clearly if you choose a very small angle. If the angle between two successive items is  $y$ , the angle of the  $k$ th item with the (fixed) 0th item is  $ky$ . Now if you change  $y$ , item 0 stays put, item 1 gets nudged by 1 times the difference between old and new  $y$ , item 2 is nudged by two times that amount, item  $k$  by  $k$  times that amount, and so on. The picture does not change like the spokes of a rolling wheel, but the outer parts are swept along much further than the inner parts.

**The number  $x$  that you change with the left slider** is the logarithm of the radial growth factor. The underlying math is simpler in terms of this  $x$  than in terms of the radial growth factor itself. In fact, this way (and with  $y$  in radians)  $x$  and  $y$  really measure displacements in the same units. Try setting both to the same value (about 0.2 is a nice one) and you'll see that each next item moves just as far outward as it moves anticlockwise (you can see the jumps are all at a  $45^\circ$  angle with a line to the centre).

By the way, the relation between slider displacement and  $x$  is itself logarithmic once again, for a different reason: that way the amount of change in the picture per mouse click turns out to be the same at all  $x$  scales.

Changing  $x$  does not simply zoom the picture in or out. I've been asked a few times why Spiral does not give an option to zoom in and out. The answer is that there's not much else to see at different scales of magnification. If you would blow up the tantalizingly crowded centre portion, you would see pretty much the same sort of pattern you see now. Rather, what changing  $x$  does is cumulative (just like with  $y$ ): the items further out are affected much more than the inner ones.

There is a curious analogy between what happens in the  $x$  (outward) direction and what happens in the  $yi$  direction (going round):

With zooming, the proportions of the pattern would stay the same. For example, if one dot or circle is twice as far from the centre as another one, it would move twice as much when zoomed, so afterwards it's still twice as far from the centre as the other one. Changing  $x$ , the item twice as far from the centre moves more than twice as far, the very shape of the pattern is distorted, and by the magic of complex number arithmetic different metaspiral families crowd each other in and out of view.

With a rotation, the shape of the pattern would stay the same. For example, if one dot or circle is twice as far from the centre as another one, it would move twice as far sideways when rotated, so it rotates by the same angle. Changing  $y$ , the item twice as far from the centre moves more than twice as far, the shape of the pattern is twisted, and again by the magic of complex number arithmetic different metaspiral families swirl in and out of view.

This analogy flows naturally from **complex number** arithmetic. If you're new to the wonders of complex numbers and this program manages to whet your appetite, it will have served its purpose well. I don't recommend any book in particular, tastes differ. Almost any library will carry some titles on the subject.

If you're familiar with complex numbers, read on. The program simply plots  $\exp kz$  for successive integer  $k$ . These are the successive powers of a complex number  $\exp z$ , so the jump from each item to the next represents a multiplication by  $\exp z$ . Splitting the logarithm  $z$  of this number into its real and imaginary parts we get our  $x + yi$ . Now  $\exp z = \exp(x+yi) = (\exp x)(\exp yi)$ . This means the jump corresponds to increasing the radial distance by a factor  $\exp x$ , and at the same time turning anticlockwise by  $y$  radians.

Zooming or rotating would multiply each item by the same fixed  $\exp a$  or  $\exp bi$ , leaving the proportions between items' positions intact. Changing  $z$  (so  $x$  or  $yi$ ) on the other hand affects those  $\exp kz$  with larger  $k$  more than those with smaller  $k$ .

Read on about [metaspirals](#), or go back to the [introduction](#) or to the [menu overview](#)?

## Metaspirals

Read [Spiral basics](#) first?

In one sense, the program always only plots one spiral. Exponential sequences in complex numbers always follow a spiral pattern. But you will almost certainly not see this one base spiral. You can follow it by repeatedly clicking a [mouse](#) button. What's special about the spirals in Nature that this program emulates is two things.

One feature is that (for most of the range of settings in this program) the growth in the radial direction, at each step, is quite small, whereas the angle is large. You will not associate successive steps with each other, because they're all over the place. But you will associate an item with another item say  $n$  steps further. Sooner or later the cumulative effect of big angles will add up to something close to a multiple of  $360^\circ$  (or  $2\pi$ ), say after  $n$  steps. And precisely because the radial steps are so small, the  $n$ th item will still be close to the item you started off with. The  $n$ th item appears to be only a small "metastep" away from the 0th item. And of course the  $2n$ th item is another metastep away, then the  $3n$ th one... forming a metaspiral of order  $n$ .

You can point at metaspirals for any  $n$ , just by counting off every  $n$ th item (or let Spiral's [highlighting](#) mechanism do it for you). But only for some numbers will the metasteps be small, and appear to be the "next one" you come across going outward. There are no absolutes here, metaspirals of any order are present in every pattern, and you can smoothly go from a pattern where spirals of some orders (say)  $a$ ,  $b$ , and  $c$  are most visible, to one where orders (say)  $p$  and  $q$  are more prominent.

The other feature of Nature's spirals is that the angle of the base step is exactly  $2\pi/\varphi$  where  $\varphi$  is the number  $\frac{1}{2} + \frac{1}{2}\sqrt{5}$ , of [Fibonacci](#) and Golden Mean fame. To see why this is so is one of the reasons why i wrote this program. And it is easier seen than explained. With almost any other angle, the circles touch or even overlap, with one spiral periodicity dominating.  $2\pi/\varphi$  seems to be the only angle where items don't get in each other's way, no matter how small you make  $x$  (and with it the radial growth factor)!

Admittedly, the circles are made smaller for smaller  $x$  (Spiral makes them proportional to  $\sqrt{x}$ , and to the distance from the centre of course), but this is only natural, and related to the available surface area. The point is that only with  $2\pi/\varphi$  this area is distributed efficiently; with other angles there are bare patches here and overlaps there.

Try it. With any other angle  $y$ , although there may not be overlaps at a large value of  $x$ , the pattern will mess up when you squash it by decreasing the  $x$ . But as long as the angle is  $2\pi/\varphi$ , you'll see that on decreasing  $x$  an 8's and 13's pattern effortlessly changes to a 13's and 21's pattern, to a 21's and 34's, and so on. All the items are in the right place, the angle between the 8's and 13's widens out and the 21's fit in between. This is truly beautiful mathematics at work.

I sort of understand why this only happens at this one angle, i think, when i got it worked out exactly (or have read up on it) i'll include the explanation here (in a future version).

You can also keep  $x$  fixed, and change  $y$ . For very small  $x$ , near the bottom of the scale of this program, you can see how even a one or two minutes of arc deviation from  $2\pi/\varphi$  in  $y$  destroys the pattern. Yet Nature, in her 13, 21, and 34 repeating spiral manifestations, works at these kind of  $x$  scales. Amazingly, a pineapple or sunflower has grown within a tolerance of only minutes of arc, otherwise we would have seen a large distortion in its shape!

Back to the [introduction](#) or to the [menu overview](#)?

## Program history

Spiral was written by Marijke van Gans, publishing as silicon alley.

This program was inspired by a question from Robert Sacks on the CompuServe Science/Math Forum arising from his fascination with Fibonacci spirals. Discussions with him on programming have also helped its development.

Pre-release versions:

- 0.00, the original, had a fixed limited size due to the use of a clipping region. All the calculations were redone each time it reappeared from underneath another window or its own menus...
- 0.01 adds resizableability and the use of an off-screen bitmap to store the picture, with an option to watch the spirals being drawn or not. On startup, it calculates a cos and sin table (flower-shaped calculation progress display).
- 0.03 (glitch in numbering) uses fast integer math in the calculation and drawing loop. Also some display routine cleaning up.
- 0.04 offers a single dialog box in stead of separate menu options for the various hi-lites. Also, they now come in 8 different colors.
- 0.99 (getting ready for release!) ditches the Show Redraw option, allows the hi-lite period numbers to be changed, and gives the option to leave the hi-lite box on screen (it used to disappear when you clicked on OK). It makes the scroll bar resolution twice as fine, and brings the minimum  $x$  down to 0.0006. Finally it adds Specks and Squares to the Dots and Circles.
- $\pi/\sqrt{10}$  adds a "Warning - mathematics ahead" screen at the beginning, also new home for table calculation progress display. Hexagon shapes. A new (less misleading) main window layout, with status bar at the bottom. Standard Help menu items. Also a visually revamped help file!

At last! Release.

- 1.00 (27 Sep. '94) dumps the "Warning" screen again. The cosine/sine table is now contained within the program (8K). More elegant internals <g>: it now sends itself messages. Added code to deal with anisotropic video modes (in parallel, so it doesn't slow down isotropic ones). Changed the way slider resolution varies along its range (same visual impact per click at all scales now) to make the program much more usable. Moved the display to one side when maximized, so you got somewhere to put the hi-lite dialog box. Fixed bug that allowed multiple hi-lite boxes. And finally, a total overhaul of the text of this help file.
- 1.10 (9 Dec. '94) adds sophisticated mouse features. Also edited this help file extensively. The original 1.10 will be very rare, because 1.10(a) (same date) fixed a minor blemish in the cursor display.

Plans for future releases already include

- User selection of colors.
- Better keyboard support (for mouseless systems).

- Speeding up the display using WinG.

Suggestions and comments are always welcome!

Back to the [introduction](#) or to the [menu overview](#)?

## Distribution policy

### To anybody using this program:

Spiral is being published as **freeware**. If you like the program, there are a number of things you could do:

Give a **copy** to anybody you think might like it too.

Tell me what you like or don't like, whether the help file is clear enough, and any **suggestions** you may have. If you find a **bug**, definitely tell me!

A small donation is always most welcome, but only do this if you can easily afford it. Better still, get me some **work**!

Marijke van Gans

### To distributors of shareware and freeware:

Please note this software is an original creation of the mind, and as such copyright material under British law, and hence (because of various treaties) under international law. The author reserves all rights to her work, © '94 Marijke van Gans, t/a silicon alley.

You are allowed to charge your customers a reasonable handling fee (such at the absolute discretion of the copyright holder) for copying and distributing this software, as long as you send us a copy of your catalog, ad, or price list with an address to contact you. That way we get a chance to send you upgrades and new software!

Back to the [introduction](#) or to the [menu overview](#)?

## Fibonacci numbers

These numbers, called after Leonardo Fibonacci of Pisa (an Italian mathematician of the 13th century), consist of the sequence

..., -3, 2, -1, 1, 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ...

and so on. Each number is the sum of the previous two in the sequence. They pop up in the most diverse places.

The ratio  $F(k+1) / F(k)$  tends to  $\varphi = \frac{1}{2} + \frac{1}{2}\sqrt{5}$  for large  $k$ , and to  $-1/\varphi = \frac{1}{2} - \frac{1}{2}\sqrt{5}$  for large negative  $k$ .  $\varphi$  and  $-1/\varphi$  are the two solutions of the equation

$$x^2 = x + 1.$$

The two or three sets of spirals visible in many living things like pine cones, pineapples, sunflowers etc. usually have consecutive Fibonacci numbers between 5 and 89 as their number of spirals going round.





P O Box 2799, London E9 7AW, England, Europe.

CompuServe: 100345,2675

Internet: 100345.2675@compuserve.com



