HP Forum Archive 14

[Return to Index | Top of Index]

geometric progression

*Message #1 Posted by **Jim Kimes** on 15 June 2004, 3:09 p.m.*

I'm trying to establish a geometric progression formula for the following series but I'm not sure it's a geometric progression. I need to be able to solve the nth term of the following series:

(364/365) * (363/365) * (362/365)...

I want to use 'a' for the changing numerator and 'b' for unchanging denominator and r for the ratio. I know the formula for a geometric progression but it's not that simple. What is the answer here? I need a formula that will fit all similar conditions that I can program on an HP 41. Right now I'm getting the answer by looping but I think there's a better way. Thanks.

Re: geometric progression

*Message #2 Posted by **hugh steers** on 15 June 2004, 6:49 p.m.,*
in response to message #1 by Jim Kimes

how about (aPr)/b^r for r terms. where aPr means permutations of a taking r. ?

Edited: 15 June 2004, 6:51 p.m.

Simplified form

*Message #3 Posted by **Tizedes Csaba [Hungary]** on 16 June 2004, 3:48 a.m.,*
in response to message #1 by Jim Kimes

Hello,

try this:

```
FACTORIAL(n-1)/n^(n-1) = GAMMA(n)*n^(1-n)
```

```
in your case n=365
```

Csaba

Simplified form - I think I was crazy...

*Message #4 Posted by **Tizedes Csaba [Hungary]** on 16 June 2004, 3:56 a.m.,*
in response to message #3 by Tizedes Csaba [Hungary]

```
FACTORIAL(n-1)/n^(n-1) = GAMMA(n)*n^(1-n)
```

I'm so sorry, I think my mind is not workin today... And I can't to delete my message... Sorry again!

Cs.

Re: Simplified form - I think I was crazy...

*Message #5 Posted by **Jim Kimes** on 16 June 2004, 2:33 p.m.,*
in response to message #4 by Tizedes Csaba [Hungary]

The problem with factorials is that calculators and computers can't handle super large factoials. They overflow. Besides I'm looking for a partial or a truncated factorial. Logs of factorials may be able to handle that. With your proposed solution, when a = 364, b = 365 and n = 23, do you get .493? If not the real problem isn't solved. Try the problem manually, doesn't take that long, and verify the .493 answer and then formulate it for any value of a & b & n, and then you will feel comfortable with working on a solution.

Re: geometric progression

*Message #6 Posted by **bill platt** on 16 June 2004, 3:25 p.m.,*
in response to message #1 by Jim Kimes

> Quote:
>
> ---
>
> (364/365) * (363/365) * (362/365)...
>
> ---

Won't this series go on for 364 cycles, decreasing all the time, and at an increasingly rapid rate, until you get to (0/365) at whcih point it goes disjointly to zero?

You could very easily write a loop, using i as a counter variable, and loop 364 times.....what am I missing here?

Regards,Bill

Re: geometric progression

*Message #7 Posted by **Jim Kimes** on 16 June 2004, 8:16 p.m.,*
in response to message #6 by bill platt

"You could very easily write a loop, using i as a counter variable, and loop 364 times.....what am I missing here?"

You're under the impression that I haven't solved my problem. Oh, I've solved it. I have a loop routine that works very well on a 41 but I can't loop with a 19BII and so I want to use some kind of progression formula that will find partial factorials. At n = 23 the ratio becomes .493. I'm beginning to think there is no formula. I haven't heard back from Dr. Math yet.

A solution to Jim's problem [LONG]

*Message #8 Posted by **Valentin Albillo** on 17 June 2004, 4:58 a.m.,*
in response to message #7 by Jim Kimes

Hi, Jim:

Jim posted:

"I can't loop with a 19BII and so I want to use some kind of progression formula that will find partial factorials. At n = 23 the ratio becomes .493. I'm beginning to think there is no formula."

Yes, there is. A "partial" factorial is technically called a ***Pochhammer Symbol***, which itself is nothing more than a Gamma function divided over another gamma function, like this:

$$(x)_n = x*(x+1)*...*(x+n-1) = Gamma(x+n)/Gamma(x)$$

This is the clue for solving your problem non-iteratively. In HP-71B BASIC (i.e.: nearly plain-vanilla English), your problem would be solved like this:

```
10 B=365 @ N=23
20 DISP FACT(B)/FACT(B-N)/B^N
```

where FACT is the factorial function. However, upon running this results in an *overflow*, as FACT(365) exceeds the range for real numbers in the HP-71B and in most any other calculator or system.

What can we do, then ? Easy. Just use the *asymptotic series* for the factorial function, aka **Stirling's series**, to compute the natural logarithm of x! instead of x!. This computation won't overflow for your arguments, and once we have the logarithm, we just use the exponential function EXP (aka *e^x*) to deliver the final result. Stirling series is:

```
x! = (x/e)^x*Sqrt(2*Pi*x)*(1+1/(12*x)+1/(288*x^2)-139/(51840*x^3)+ ...)
```

so its logarithm will be:

```
z = ln(x!) = x*ln(x)-x+ln(2*Pi*x)/2+ln(1+1/(12*x)+1/(288*x^2)-139/(51840*x^3))+ ...
```

and of course your final result is simply *e^z*. The resulting, non-iterative HP-71B BASIC program is:

```
10 B=365 @ N=23
20 DISP EXP(FNF(B)-FNF(B-N)-N*LN(B))
30 END
90 DEF FNF(X)=X*LN(X)-X+LN(2*PI*X)/2+LN(1+1/12/X+1/288/X^2-139/51840/X^3)
```

where DISP is the statement used to show a result on the display (aka PRINT, VIEW, etc), EXP is the exponential function *e^x*, LN is the natural logarithm function, PI is 3.14159265359, and FNF is a user-defined function, which can be substituted for a subroutine in other languages, passing it the appropriate parameter. Upon running, it promptly returns:

```
>RUN
      .492702772644
```

which agrees well with your .493 and is accurate to nearly 8 decimal places (actually, the exact result is 0.49270276567601459277458277+). If still more accuracy is desired, you can use some additional terms to the Stirling's series, have a look at this link:

Stirling's series numerators

and its twin for the denominators. You can get easily 12+ decimals by using more terms. On the other hand, if you don't need that much accuracy, you can save RAM and execution time by deleting terms from the Stirling's series above, like this:

```
90 DEF FNF(X)=X*LN(X)-X+LN(2*PI*X)/2+LN(1+1/12/X)
```

though you should test if the accuracy you then get is adequate for your intended purposes, particularly for large N. For your specific numeric example, you still get 8 decimals !

Hope that helps. Best regards from V.

Edited: 17 June 2004, 10:26 a.m.