Going to Infinity: What Happens to Functions When the Independent Variable Gets Bigger and Bigger and Bigger?

Note: You may notice differences between this Maple worksheet and the equivalent Mathematica notebook. These differences were introduced to preserve the content of these modules and were necessary because of major functional differences between Maple and Mathematica.

Introduction

OBJECTIVE: To interpret limits going to infinity graphically and numerically.

In this module, you will explore limits of functions as the independent variable approaches infinity, and you will formulate and test some end-behavior models. We will examine graphs and create tables of function values for large values of the independent variable. The focus here is on comparing the behavior of some important functions that you will encounter throughout your study of calculus.

Technology Guidelines

NOTE: If you have just finished a worksheet, **restart** *Maple* before executing a new worksheet.
TO OPEN SECTIONS,
Click on the **PLUS** sign at the left hand side of the screen *or* select **Expand All Sections** from the **View** drop down menu.

TO STOP AN EXECUTION
Click on **STOP** button from the toolbar.

ORDER OF EXECUTION
Execute commands in the order given. Do not skip any *Maple* Input lines within a given worksheet

Alternatively, you can execute the entire worksheet by selecting the **Execute Worksheet** command from the **Edit** drop down menu.

SAVING WORKSHEETS.

You can save anytime to any directory you choose, and it is wise to save often.
EXPERIENCING MAJOR PROBLEMS
Save if appropriate, and then shut dow*n Mapl*e and start it up again.

Part I: Exploring End Behavior

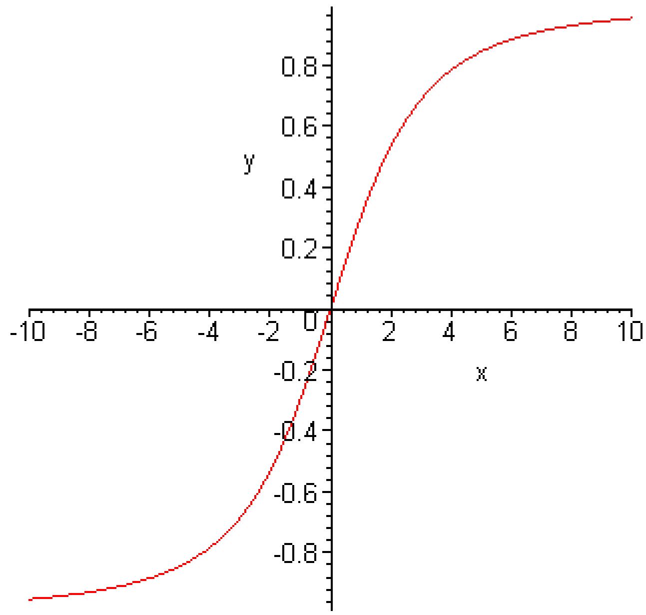
Consider the following four examples. Can you determine why the end behavior is so different in each case?

> **restart;**

- **Example 1:** $\frac{x}{\sqrt{10+x^2}}$

> **g1:=x/sqrt(10+x^2):**

> **plot(g1, x=-10..10, labels=[x,y]);**



It looks as though the function is approaching 1 as *x* gets large. Let's evaluate the function at 1 million.

> **eval(g1, x=1000000.0);**

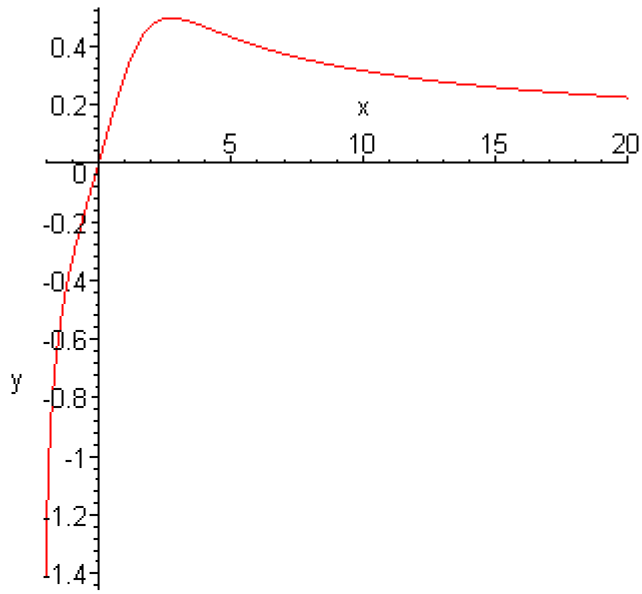
1.000000000

- **Example 2:** $\frac{x}{\sqrt{10+x^{23}}}$

Now, we will increase the power of *x* in the denominator.

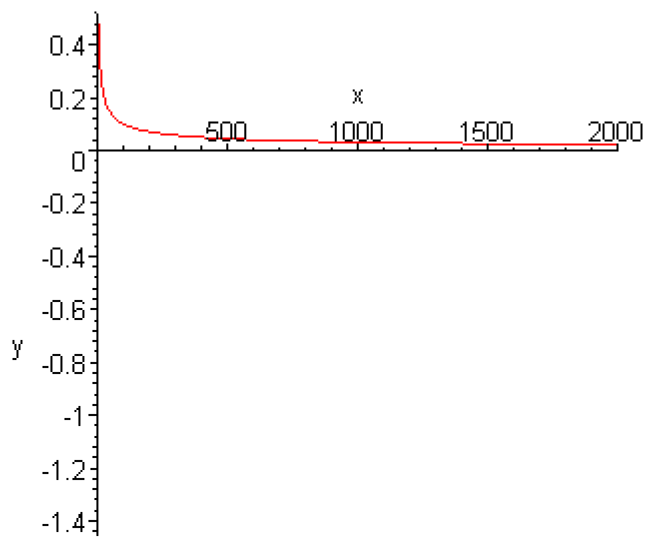
> **g2:=x/sqrt(10+x^3):**

```
> plot(g2, x=-2..20, labels=[x,y]);
```



What do you think happens as *x* gets bigger and bigger? Let's plot the function out further and also evaluate it at a large value of *x*, say 1 billion.

```
> plot(g2, x=-2..2000, labels=[x,y]);
```




```
> eval(g2, x=10.0^9);
```

0.00003162277660

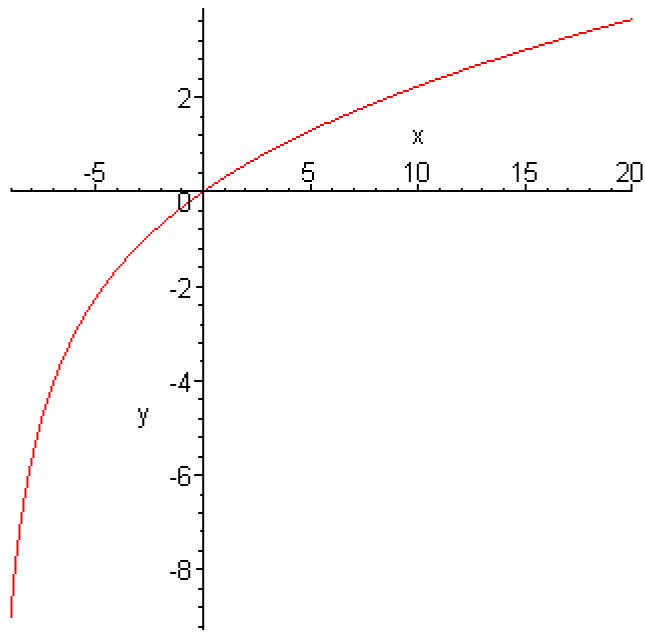
- **Example 3:** $\dfrac{x}{\sqrt{10+x}}$

Now we decrease the power of *x* in the denominator.

> **g3:=x/sqrt(10+x);**

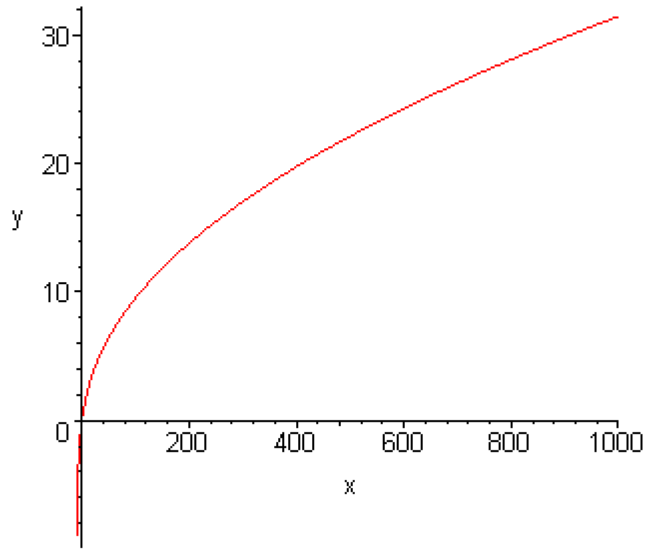
$$g3 := \frac{x}{\sqrt{10+x}}$$

> **plot(g3, x=-9..20, labels=[x,y]);**



What do you suppose is happening as *x* gets large beyond bound? Let's extend the graph.

> **plot(g3, x=-9..1000, labels=[x,y]);**



It looks as though it is getting bigger, but not terriby fast. Let's evaluate the function at 1 trillion.

```
> eval(g3,x=10.0^12);
```

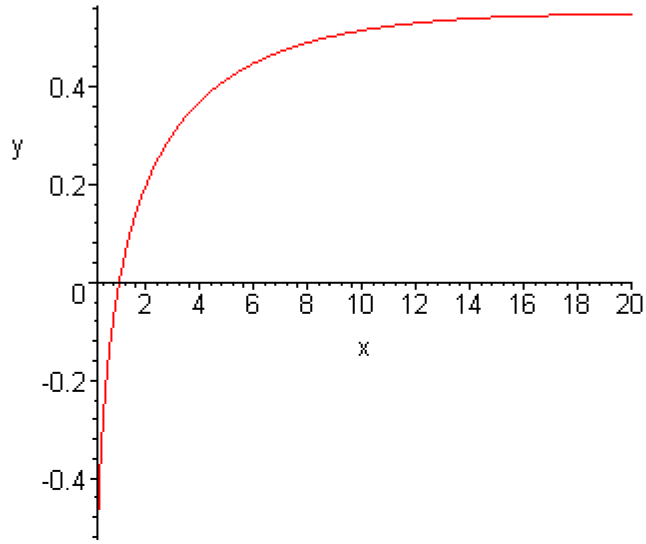
$$0.1000000000\ 10^{7}$$

- **Example 4:** $\dfrac{\mathbf{Log}(x)}{\sqrt{10+x^2}}$

Here is a function involving the natural log of *x*.

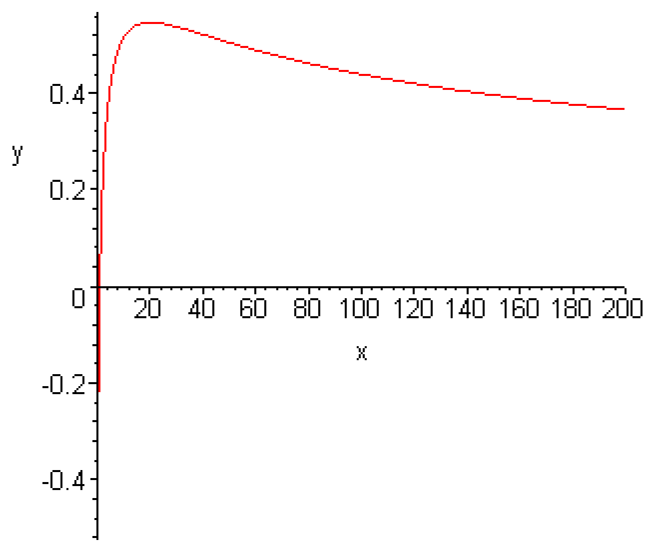
```
> h2:=log(x)/sqrt(10+x);
  plot(h2, x=0.2..20,labels=[x,y]);
```

$$h2 := \frac{\ln(x)}{\sqrt{10+x}}$$



What do you think happens to the value of $h(x)$ as x gets bigger and bigger? Do we need to extend the plot?

> **plot(h2, x=0.2..200, labels=[x,y]);**



Now it has turned downward and seems to be approaching 0, but very slowly. Is that what you guessed before? Let's evaluate the function at $x = 1$ trillion to see if the downward trend continues.

> **eval(h2, x=10.0^12);**

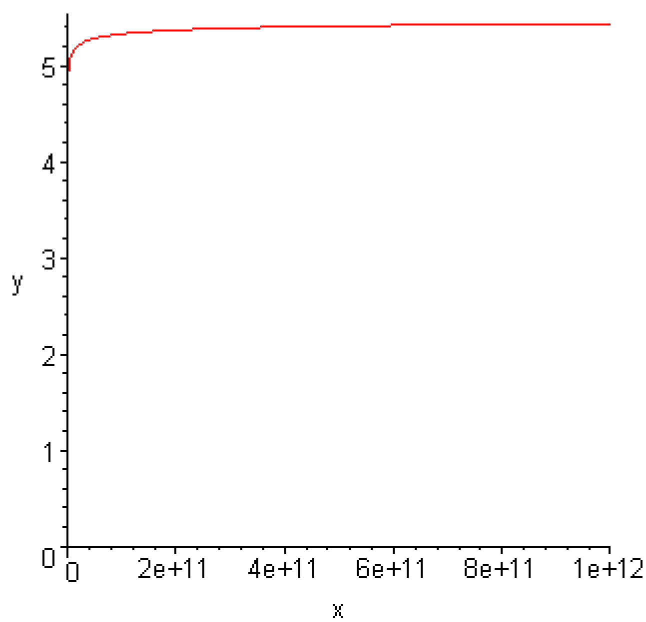
0.00002763102112

You Try It: Part I

- **Limits Involving** $\frac{\text{Log}(x)}{\sqrt{10+x^n}}$

In the fourth example in Part I, we used 2 for the power of *x* in the denominator of the function $\frac{\text{Log}(x)}{\sqrt{10+x^n}}$. In this example, we want you to consider other powers of *x*, therefore, we leave the power variable and call it *n*. Explore what happens as you change the value of *n*, and see if you can find a value for *n* that will give a finite nonzero value for the limit as *x* approaches infinity. We help you get started by looking at $n = 0.1$.

```
> h1:=log(x)/sqrt(10+x^0.1):
  bigx:=10^12:
  plot(h1, x=1..bigx, labels=[x,y]);
  eval(h1, x=bigx):
  evalf(%);
```



5.434700723

Do you think this function approaches a finite limit other than 0 as *x* approaches infinity? Check it out for a very large value of *x* (say, 10^{50} , for example) by changing the value of **bigx** in the cell above and re-executing it. If it doesn't work, try some other values for *n*, and see if you can find one that does work.

Part II: End Behavior Models

An end behavior model for a function $f(x)$ is a function $g(x)$ such that $\lim_{x \to \infty} \frac{f(x)}{g(x)} = 1$. Can we find an end behavior model for $f(x) = \frac{x}{\sqrt{10+x^2}}$, the first example function in Part I? To do this you should consider what happens to the denominator when x is a very large number. In this case, $10+x^2 = x^2$, since 10 is almost insignificant in comparison to x^2 when x is large, and $\sqrt{10+x^2} = \sqrt{x^2} = x$. Therefore, we guess that $g(x) = \frac{x}{x} = 1$ is an end behavior model for

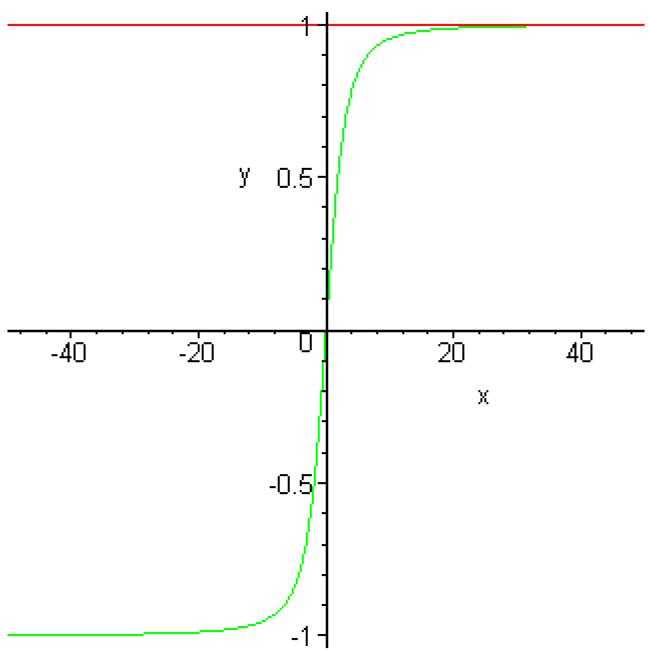
$f(x)$. Let's test it out, first by taking the limit of $\frac{f(x)}{g(x)}$ as x approaches infinity, and then by plotting $f(x)$ and $g(x)$ on the same graph.

```
> f:=x->x/sqrt(10+x^2):
```

```
> g:=x->1:
  limit(f(x)/g(x), x=infinity);
```

$$1$$

```
> plot({f(x), g(x)}, x=-50..50, labels=[`x`,`y`]);
```



Because the value of the limit is 1, $g(x)$ is an end-behavior model for $f(x)$. The graph shows that as x gets large, $f(x)$ and $g(x)$ get closer together, as we would expect.

You Try It: Part II

- **How Do Functions End?**

1. Use the approach outlined in Part II to find end behavior models for the two functions in Examples 2 and 3 in Part I. Test your model by evaluating $\lim_{x \to \infty} \frac{f(x)}{g(x)}$ and provide support for the result by plotting $f(x)$ and $g(x)$ on the same graph.

2. Based upon the results of your investigation of the function $f(x) = \frac{\ln(x)}{\sqrt{10 + x^2}}$ in Part I, Example 3, do you think there is an end behavior model $f(x)$ that is of the form $g(x) = x^n$, where n is a positive number less than 1? We said earlier that the log functions grow very slowly as x gets larger. Well, radical functions of the form just described also grow slowly. Which functions do you think grow more slowly log functions or those of the form x^n , where n is a rational number less than 1?

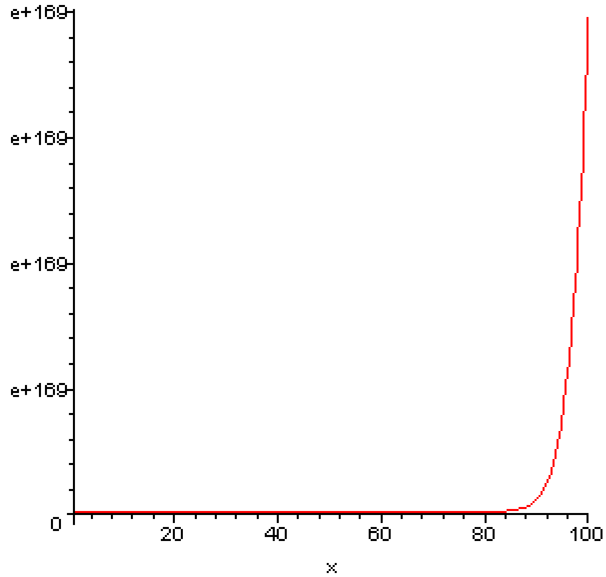
Part III: Rates of Growth of Polynomial Functions Versus Exponential Functions

Which type of function will grow faster as x gets large without bound, a polynomial function like x^{100} or an exponential function like 2^x ? Let's see what we can find out about this issue by graphing the ratio of the two functions.

```
> restart;
  plotsetup(gdi):
```

```
> f:=x^100/2^x;
  plot(f(x) , x=.5..100, labels=["x","y"]);
  print(`This function evaluated at x = 100 is`  , eval(f, x=100.0));
```

$$f := \frac{x^{100}}{2^x}$$

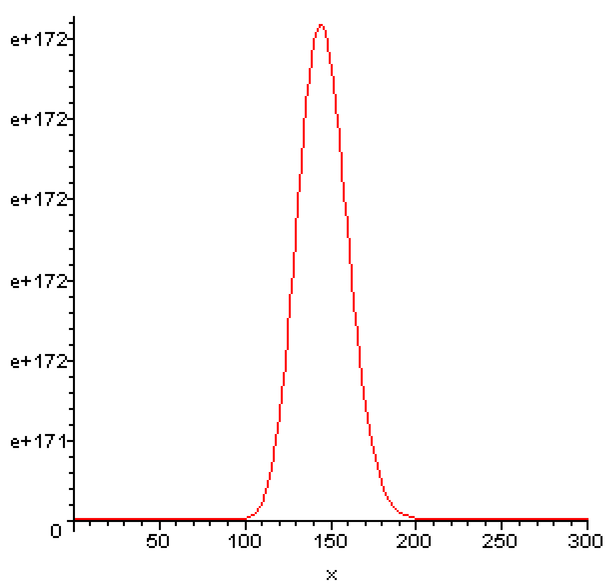


This function evaluated at x = 100 is, 0.7888609054 10^{170}

(Note: In case the numbers on the vertical axis are showing only the powers of 10, we have evaluated the function at 100 so you can see how large the function becomes.)

This graph seems to suggest that the polynomial function grows more rapidly and outstrips the exponential function. To be sure that this pattern continues, let's plot the ratio of the two functions over a larger domain.

```
> plot(f, x=0..300, labels=['x','     y']);
  print(`This function evaluated at x = 150 is`  , eval(f, x=150.0));
  print(`This function evaluated at x = 300 is`  , eval(f, x=300.0));
```



This function evaluated at x = 150 is, $0.2848567768\ 10^{173}$

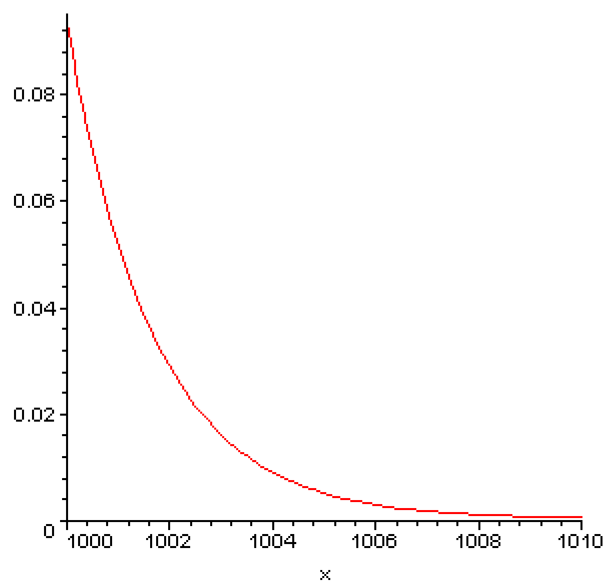
This function evaluated at x = 300 is, $0.2530036419\ 10^{158}$

Look at this! For a while, the polynomial function out-runs the exponential function, but eventually it is overtaken by the exponential function.

(Note: In case the numbers on the vertical axis are showing only the powers of 10, we have evaluated the function at two points so you can see how large the function becomes.)

The value of the ratio at $x = 300$ is still large, however, so let's see what happens for values of x in the neighborhood of 1000.

```
> plot(f(x), x=1000..1010, labels=[x,y]);
  print(`This function evaluated at x = 1010 is` , eval(f, x=1010.0));
```



This function evaluated at x = 1010 is, 0.0002465140958

It appears that 2^x is now growing faster than x^{100} therefore, the ratio of the latter over the former is going to 0, as x gets larger.

Next, we create a table of very large values of x to see if the pattern persists.

```
> for i from 10.0^6 to 2.0*10^6 by 10.0^5 do
    [i,(eval(f(x),x=i))];
    od;
```

$$[0.1000000000\ 10^{7}, 0.1010034059\ 10^{-300429}]$$

$$[0.1100000000\ 10^{7}, 0.1393279145\ 10^{-330528}]$$

$$[0.1200000000\ 10^{7}, 0.8381617301\ 10^{-360628}]$$

$$[0.1300000000\ 10^{7}, 0.2511724808\ 10^{-390727}]$$

$$[0.1400000000\ 10^{7}, 0.4157899561\ 10^{-420827}]$$

$$[0.1500000000\ 10^{7}, 0.4126956904\ 10^{-450927}]$$

$$[0.1600000000\ 10^{7}, 0.2623831222\ 10^{-481027}]$$

$$[0.1700000000\ 10^{7}, 0.1127881232\ 10^{-511127}]$$

$$[0.1800000000\ 10^{7}, 0.3428114698\ 10^{-541228}]$$

$$[0.1900000000\ 10^{7}, 0.7648892038\ 10^{-571329}]$$

$$[0.2000000000\ 10^{7}, 0.1293217592\ 10^{-601429}]$$

If the term **Float[*a*,*b*]** appears here, it refers to the number *a* times 10 raised to the power *b*. You can see that these numbers are very close to 0.

In the long run, exponential functions of the form a^x where *a* is a positive constant greater than 1, will always outgrow a polynomial function of the form x^n, no matter how big *n* is and no matter how close *a* is to 1. The power function may win out at first, but in the end the exponential function always comes out on top.

You Try It: Part III

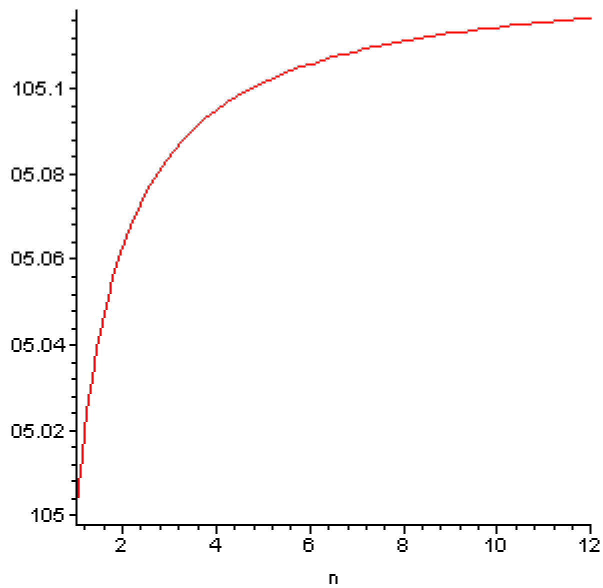
- **What does this have to do with NP-Complete problems?**

A very important concept in computer science is related to polynomial growth rate versus exponential growth rate. Look up the term NP-Complete, and write two paragraphs explaining what this is about and why it is important.

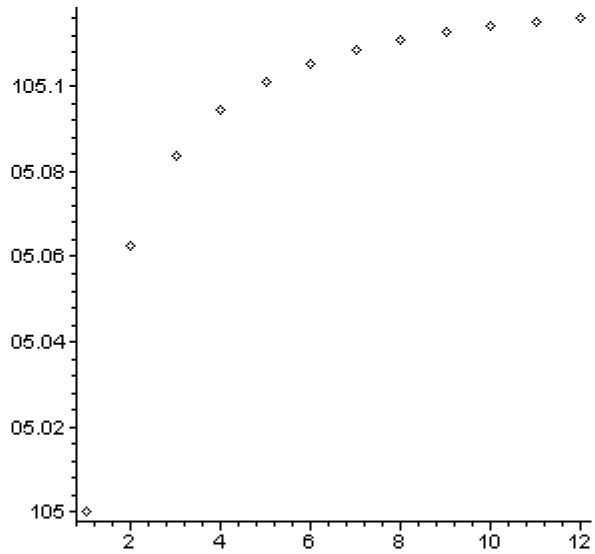
Part IV: Euler's Very Special Number

Here is a function that you may have seen before if you studied interest on an investment that compounds more and more frequently during a year. Let's suppose that you invest $100 in an account that pays interest at an annual percentage rate of 5% and you leave the earned interest in the account to compound. If the interest is compounded *n* times per year, the amount of principal in the account at the end of one year will be $100\left(1+\frac{.5e-1}{n}\right)^n$. We graph the principal in the account after one year for n = 1, 2, . . . , 12, and generate a table of values.

```
> plot(100*(1+0.05/n)^n, n=1..12, labels=["n", "accumulation"]);
```




```
> fn:=100*(1+0.05/n)^n:
  accumulation:= seq([i,subs(n=i, fn)],i=1..12):
  plots[pointplot]([accumulation]);
  print(`Here are some examples of the amount in the account (in dollars) at the end of one y
  the interest is compounded once, twice, ..., up to 12 times a year (monthly)`);
  matrix([[`interest pd / yr`,`account balance`],accumulation]);
```



Here are some examples of the amount in the account (in dollars) at the end of one year if the inte mpounded once, twice, ..., up to 12 times a year (monthly)

interest pd / yr	*account balance*
1	105.00
2	105.0625000
3	105.0837964
4	105.0945337
5	105.1010050
6	105.1053311
7	105.1084274
8	105.1107529
9	105.1125639
10	105.1140132
11	105.1152006
12	105.1161902

What do you think happens to the function below, as n gets larger and larger, in other words, as you approach a situation where the interest compounds continuously at every instant in time during the year? To find out, let's evaluate the limit of the principal function as n goes to infinity.

```
> limit(fn, n=infinity);
```

$$105.1271096$$

Because we wrote the principal function with the interest rate in decimal form, *Maple* evaluates the function using floating-point arithmetic, giving a decimal approximation for the value of the function and the limit. If we rewrite the principal function with the interest rate in fraction form, *Maple* will give us the exact value of the limit, provided it can evaluate the limit. Let's see what happens.

```
> limit(100*(1+5/(100*n))^n, n=infinity);
```

$$100\, e^{\left(\frac{1}{20}\right)}$$

Are you surprised? Why is this happening? The base is Euler's number *e*. Why is this happening?

You Try It: Part IV

- **Limits Involving** $\left(1+\frac{r}{n}\right)^{nt}$

1. Evaluate the last limit in Part IV, but change the initial investment amount to P0, and change the annual percentage rate to *r*. Be sure that you **restart** before you evaluate the limit, otherwise, *Maple* will use the last value that was assigned to *r* when it evaluates the limit.

Now we will consider what would happen if we were to invest \$100 for periods longer than one year. Our investment earns interest for *t* years, and we consider what happens over the years if interest on our investment is compounded quarterly, contrasting this to what happens when the interest is compounded continuously. We start with an annual percentage rate of 15% and see how the investment grows if left untouched over the years. The amount of principal in the account after *t* years is given by $100\left(1+\frac{r}{n}\right)^{nt}$, where *n* is the number of times the interest is compounded each year, and t is the number of years the money is invested.

```
> r:=0.15:
  f1:=100*(1+r/n)^n;
  f2:=100*(1+r/n)^(2*n);
  f3:=100*(1+r/n)^(3*n);
  f4:=100*(1+r/n)^(4*n);
  f5:=100*(1+r/n)^(5*n);
  f10:=100*(1+r/n)^(10*n);
  f20:=100*(1+r/n)^(20*n);
```

$$f1 := 100\left(1+\frac{0.15}{n}\right)^{n}$$

$$f2 := 100\left(1+\frac{0.15}{n}\right)^{(2\,n)}$$

$$f3 := 100\left(1+\frac{0.15}{n}\right)^{(3\,n)}$$

$$f4 := 100\left(1+\frac{0.15}{n}\right)^{(4\,n)}$$

$$f5 := 100\left(1+\frac{0.15}{n}\right)^{(5\,n)}$$

$$f10 := 100\left(1+\frac{0.15}{n}\right)^{(10\,n)}$$

$$f20 := 100\left(1+\frac{0.15}{n}\right)^{(20\,n)}$$

```
> matrix([[`years`, `quarterly compound`, `continuous compounding`],
  [1, subs(n=4, f1), limit(f1, n=infinity)],
  [2, subs(n=4, f2), limit(f2, n=infinity)],
  [3, subs(n=4, f3), limit(f3, n=infinity)],
  [4, subs(n=4, f4), limit(f4, n=infinity)],
  [5, subs(n=4, f5), limit(f5, n=infinity)],
  [10, subs(n=4, f10), limit(f10, n=infinity)],
  [20, subs(n=4, f20), limit(f20, n=infinity)]]);
```

years	quarterly compound	continuous compounding
1	115.8650415	116.1834243
2	134.2470784	134.9858808
3	155.5454331	156.8312185
4	180.2227807	182.2118800
5	208.8151996	211.7000017
10	436.0378759	448.1689070
20	1901.290292	2008.553692

```
> ?

>
```

To see the effect of the interest rate on the investment, build a table that shows the amount invested after 5 years for annual percentage rates of 5%, 10%, 15%, 20%, 25%, and 30%. In each row of the table, show the annual percentage rate, the principal in the investment account after 5 years when the interest is compounded quarterly, when it is compounded continuously, and the difference in the amount of principal for the two compounding methods. Which factor has more effect on the earnings, the annual percentage rate paid on the investment or the frequency at which the interest is compounded?