

# DISTRIBUTIVITY, EXTENDED

(Tommy Burch, on and off from 08.2024 to 09.2025) (wxMaxima does not export the cleanest of latex files, so the resulting pdf contains a number of quirks and errors)

## 1 At or Down

This all started back in the late '90s at the UofA in a physics lab prep session (with R. McCroskey and R. Haar, among others). The discussion hinged on finding an operation (i termed it 'at' with the symbol @) over which addition was distributive. The exp / log solution below i came up with followed later the same day... Note that, for multiplication and addition, the distributive property can be "revealed" through exponentials and their inverse, the natural logarithms:

(% i1) `add(a,b) := log(exp(a)*exp(b));`

$$\text{add}(a, b) := \log(\exp(a) \exp(b)) \quad (\% \text{ o1})$$

(% i3) `add(a,b);`  
`add(c*a,c*b);`

$$b + a \quad (\% \text{ o2})$$

$$bc + ac \quad (\% \text{ o3})$$

(% i4) `ratsimp(%);`

$$(b + a) c \quad (\% \text{ o4})$$

Placing the multiplication between the exponentials in the logarithm, we arrive at the relation (addition) over which it is distributive. We can try the same with + between the exp's...

### 1.1 At \_ +

Define the @ (or "down") relation such that addition is distributive over it:  $a + (b @ c) = (a+b) @ (a+c)$ ...Or:  $a + \text{down}(b,c) = \text{down}(a+b,a+c)$ . Trying the following ansatz:

(% i5) at\_eq : At(a,b) = log(exp(a) + exp(b));

$$\text{At}(a, b) = \log(e^b + e^a) \quad (\text{at\_eq})$$

(% i6) down(a,b) := radcan( log(exp(a) + exp(b)) );

$$\text{down}(a, b) := \text{radcan}(\log(\exp(a) + \exp(b))) \quad (\% \text{ o6})$$

Check:

(% i7) down(a+b,a+c);

$$\log(e^c + e^b) + a \quad (\% \text{ o7})$$

yes! (It is also an obviously commutative operation.) This should work with any base for the power / log's: e.g.,

(% i8) log10(x) := log(x) / log(10);

$$\log_{10}(x) := \frac{\log(x)}{\log(10)} \quad (\% \text{ o8})$$

(% i9) radcan( log10(10^ (a+b) + 10^ (a+c)) );

$$\frac{\log(2^c 5^c + 2^b 5^b) + (\log(5) + \log(2)) a}{\log(5) + \log(2)} \quad (\% \text{ o9})$$

(% i10) logcontract(%);

$$\frac{\log(2^c 5^c + 2^b 5^b) + \log(10) a}{\log(10)} \quad (\% \text{ o10})$$

yep! The ambiguity probably speak volumes (see, e.g., M. Burgin, arXiv:1010.3287 and M. Czachor, arXiv:1412.8583 : there, the focus appears not to be in maintaining distributivity, unlike here, where it's the main focus; also, a specific choice for the "projective" function is usually made – like our choice of exp here – rather than exploring any implications of not having to choose; we should keep the same in mind by noting that the base can be changed). It may be worth exploring whether functions other than powers/logarithms can retain distributivity... at first glance, seemingly not. Some values with base e:

(% i11) down(0,0);

$$\log(2) \quad (\% \text{ o11})$$

(% i12) down(1,0);

$$\log (\%e + 1) \quad (\% \text{ o12})$$

(% i13) down(x,0);

$$\log (\%e^x + 1) \quad (\% \text{ o13})$$

(% i15) down(x,x);  
down(x-log(2),x-log(2));

$$x + \log (2) \quad (\% \text{ o14})$$

$$x \quad (\% \text{ o15})$$

Associative:

(% i17) down(a,down(b,c));  
down(down(a,b),c);

$$\log \left( \%e^c + \%e^b + \%e^a \right) \quad (\% \text{ o16})$$

$$\log \left( \%e^c + \%e^b + \%e^a \right) \quad (\% \text{ o17})$$

(% i20) diff(down(a,b),a);  
diff(down(a,b),b);  
diff(diff(down(a,b),b),a);

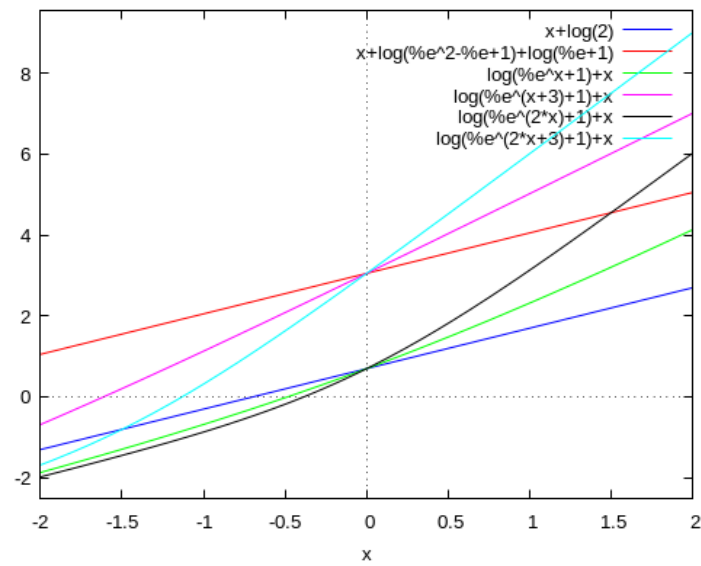
$$\frac{\%e^a}{\%e^b + \%e^a} \quad (\% \text{ o18})$$

$$\frac{\%e^b}{\%e^b + \%e^a} \quad (\% \text{ o19})$$

$$-\frac{\%e^{b+a}}{\left( \%e^b + \%e^a \right)^2} \quad (\% \text{ o20})$$

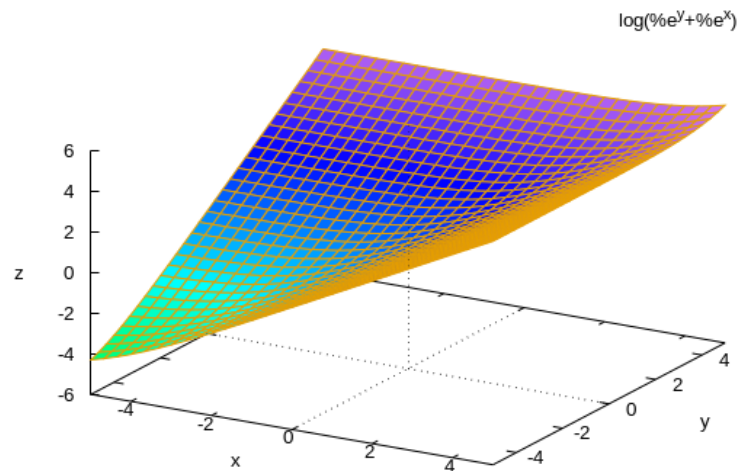
Let's have a look:

```
(% i21) wxplot2d ([down(x,x),down(x,x+3),down(x,2*x),down(x,2*x+3),down(x,3*x),down(x,3*x+3)],  
[x, -2, 2])$
```



(% t21)

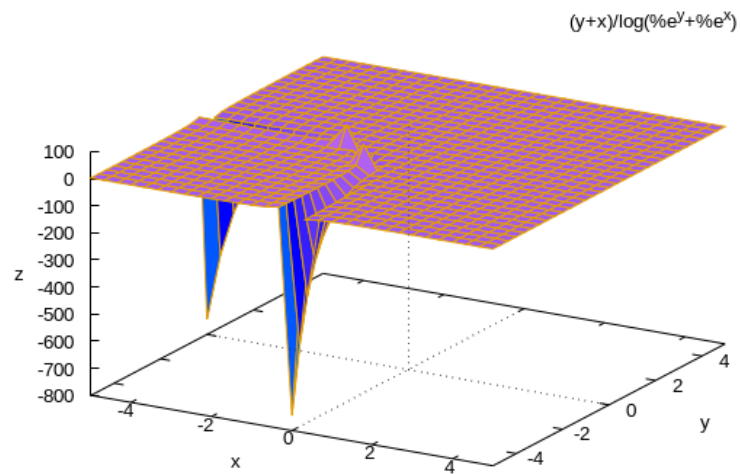
```
(% i22) wxplot3d (down(x,y), [x, -5, 5], [y, -5, 5])$
```



(% t22)

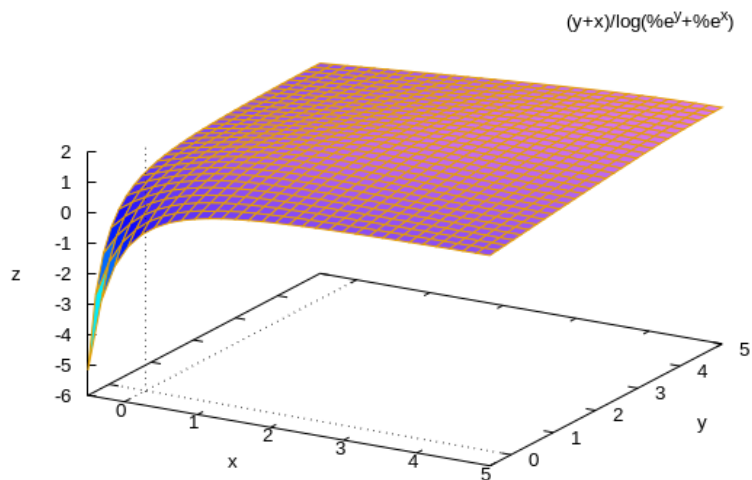
Comparing  $x+y$  with  $\text{down}(x,y)$  :

(% i23) `wxplot3d ((x+y)/down(x,y), [x, -5, 5], [y, -5, 5])$`



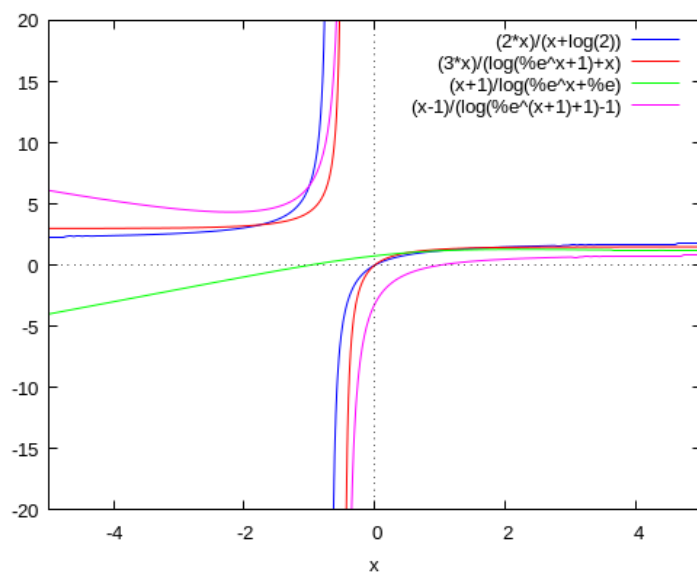
(% t23)

(% i24) wxplot3d ((x+y)/down(x,y), [x, -0.5, 5], [y, -0.5, 5])\$



(% t24)

(% i25) wxplot2d ([2\*x/down(x,x), 3\*x/down(x,2\*x), (x+1)/down(x,1), (x-1)/down(x,-1)], [x, -5, 5], [y, -20, 20])\$



(% t25)

## 1.2 At<sub>-</sub>,\*,/

We can also consider a "down-like" operation with a minus instead a plus:

(% i26) `down_min(a,b) := radcan( log(exp(a) - exp(b)) );`

$$\text{down\_min}(a, b) := \text{radcan}(\log(\exp(a) - \exp(b))) \quad (\% \text{ o26})$$

Associativity:

(% i30) `down(down_min(a,b),c);`  
`down_min(down(a,c),b);`  
`down(down_min(a,b),down_min(c,d));`  
`down(down_min(a,d),down_min(c,b));`

$$\log\left(\%e^c - \%e^b + \%e^a\right) \quad (\% \text{ o27})$$

$$\log\left(\%e^c - \%e^b + \%e^a\right) \quad (\% \text{ o28})$$

$$\log\left(-\%e^d + \%e^c - \%e^b + \%e^a\right) \quad (\% \text{ o29})$$

$$\log\left(-\%e^d + \%e^c - \%e^b + \%e^a\right) \quad (\% \text{ o30})$$

Same distributive property:

(% i31) `down_min(a+b,a+c);`

$$\log\left(\%e^b - \%e^c\right) + a \quad (\% \text{ o31})$$

As we've already seen, with times, we achieve addition. With divide, one achieves subtraction:

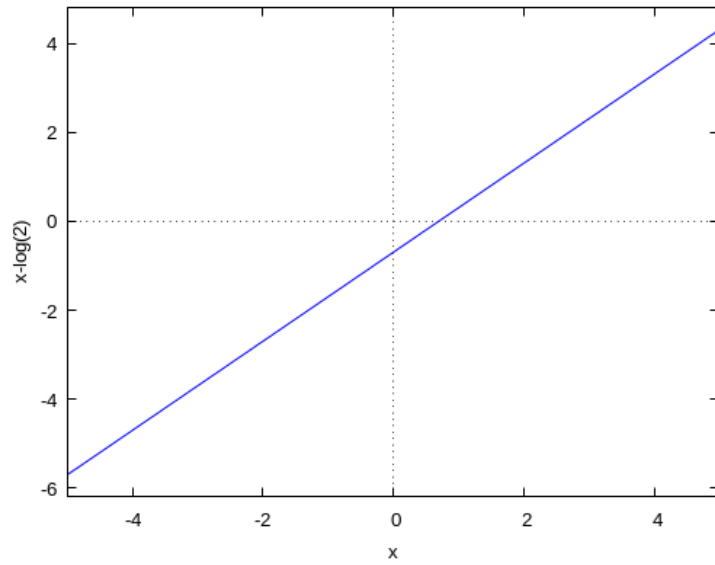
(% i33) `down_div(a,b) := radcan( log(exp(a) / exp(b)) );`  
`down_div(a,b);`

$$\text{down\_div}(a, b) := \text{radcan}\left(\log\left(\frac{\exp(a)}{\exp(b)}\right)\right) \quad (\% \text{ o32})$$

$$a - b \quad (\% \text{ o33})$$

Using @\_/ or down\_div for the "ratio" of + and @...

```
(% i35) down_div(x+x,down(x,x));
wxplot2d (2*x - down(x,x), [x, -5, 5])$
x - log(2)                                     (% o34)
```



(% t35)

### 1.3 Down differentiation

Given down, down\_min, and down\_div, one is free to define a "down derivative". However, we must take care with the notion of infinitesimals: normally  $h \rightarrow 0$  in the definition of the derivative, but here we need  $h \rightarrow -\infty$  to approach the @\_+ identity ( $x @ -\infty = x$ )...

```
(% i36) Df_Dx_eq : Df_Dx = limit( down_min( f(down(x,h)) , f(x) ) - h , h, -infinity);
```

$$Df\_Dx = \lim_{h \searrow -\infty} \log \left( \frac{e^{f(\log(e^x + e^h))}}{e^{f(x)}} - h \right) \quad (Df\_Dx\_eq)$$

```
(% i40) Df(x,h) := down_min( f(down(x,h)) , f(x) );
Df(x,h);
Df_Dx(x) := ratsimp( limit( Df(x,h) - h , h, minf) );
Df_Dx(x);
```

```
Df(x,h) := down_min( f(down(x,h)) , f(x))                                     (% o37)
```



$$\log \left( \%e^{\text{f}(\log(\%e^x + \%e^h))} - \%e^{\text{f}(x)} \right) \quad (\% \text{ o38})$$

$$\text{Df\_Dx}(x) := \text{ratsimp}(\text{limit}(\text{Df}(x, h) - h, h, -\infty)) \quad (\% \text{ o39})$$

$$\lim_{\rightarrow h \setminus \setminus > -\infty} \log \left( \%e^{\text{f}(\log(\%e^x + \%e^h))} - \%e^{\text{f}(x)} \right) - h \quad (\% \text{ o40})$$

$$\begin{aligned} (\% \text{ i43}) \quad & \text{f(x)} := A * x^b + c; \\ & \text{Df(x,h)}; \\ & \text{Df\_Dx(x)}; \end{aligned}$$

$$\text{f}(x) := Ax^b + c \quad (\% \text{ o41})$$

$$\log \left( \%e^{A \log(\%e^x + \%e^h)^b} - \%e^{Ax^b} \right) + c \quad (\% \text{ o42})$$

$$(b-1) \log(x) + Ax^b - x + c + \log(b) + \log(A) \quad (\% \text{ o43})$$

$$\begin{aligned} (\% \text{ i46}) \quad & \text{f(x)} := \log(A * x^b + c); \\ & \text{Df(x,h)}; \\ & \text{Df\_Dx(x)}; \end{aligned}$$

$$\text{f}(x) := \log(Ax^b + c) \quad (\% \text{ o44})$$

$$\log \left( \log \left( \%e^x + \%e^h \right)^b - x^b \right) + \log(A) \quad (\% \text{ o45})$$

$$(b-1) \log(x) - x + \log(b) + \log(A) \quad (\% \text{ o46})$$

$$\begin{aligned} (\% \text{ i49}) \quad & \text{f(x)} := \exp(A * x^b + c); \\ & \text{Df(x,h)}; \\ & \text{Df\_Dx(x)}; \end{aligned}$$

$$\text{f}(x) := \exp(Ax^b + c) \quad (\% \text{ o47})$$

$$\log \left( e^{e^{A \log(e^x + e^h)^{b+c}}} - e^{e^{Ax^b+c}} \right) \quad (\% \text{ o48})$$

$$(b-1) \log(x) + e^{Ax^b+c} + Ax^b - x + c + \log(b) + \log(A) \quad (\% \text{ o49})$$

(% i52) f(x) := down( g(x) , p(x) );  
           Df(x,h);  
           Df\_ Dx(x) ;  
 f(x) := down( g(x) , p(x)) \quad (\% \text{ o50})

$$\log \left( e^{p(\log(e^x + e^h))} + e^{g(\log(e^x + e^h))} - e^{p(x)} - e^{g(x)} \right) \quad (\% \text{ o51})$$

$$\lim_{h \searrow -\infty} \log \left( e^{p(\log(e^x + e^h))} + e^{g(\log(e^x + e^h))} - e^{p(x)} - e^{g(x)} \right) - h \quad (\% \text{ o52})$$

So the down-derivative maintains not only the extended distributivity, but also the property that the exponential function keeps its special role:  $D\_Dx(e^x) = e^x$ .

(% i55) f(x) := b^ (A\*x+c);  
           Df(x,h);  
           Df\_ Dx(x) ;  
 f(x) := b^{Ax+c} \quad (\% \text{ o53})

$$\log \left( e^{b^{A \log(e^x + e^h)^{b+c}}} - e^{b^{Ax+c}} \right) \quad (\% \text{ o54})$$

$$(A \log(b) - 1) x + \log(b)c + b^{Ax+c} + \log(\log(b)) + \log(A) \quad (\% \text{ o55})$$

(% i58) f(x) := log(x);  
           Df(x,h);  
           Df\_ Dx(x) ;  
 f(x) := log(x) \quad (\% \text{ o56})

$$\log \left( \log \left( e^x + e^h \right) - x \right) \quad (\% \text{ o57})$$

$$-x \quad (\% \text{ o58})$$

(% i59) kill(f);

done (\% o59)

(% i60) Eig\_Df\_eq : Df\_Dx(x) = f(x);

$$\lim_{h \rightarrow -\infty} \log \left( e^{f(\log(e^x + e^h))} - e^{f(x)} \right) - h = f(x) \quad (\text{Eig\_Df\_eq})$$

As we can see from the previous solutions of Df\_Dx, this DE is uniquely solved by f(x)=x.

(% i61) kill(f);

done (\% o61)

Alternative expressions for Df\_Dx(x) (with x=log(z)):

(% i65) x = log(z);  
log(diff( exp(f(log(z))) ,z,1));  
log(diff( exp(f(log(z))) ,z,1));  
f(x) - x + log(diff(f(x),x,1));

$$x = \log(z) \quad (\% \text{ o62})$$

$$\log \left( \frac{d}{dz} e^{f(\log(z))} \right) \quad (\% \text{ o63})$$

$$\log \left( e^{f(\log(z))} \left( \frac{d}{dz} f(\log(z)) \right) \right) \quad (\% \text{ o64})$$

$$\log \left( \frac{d}{dx} f(x) \right) + f(x) - x \quad (\% \text{ o65})$$

## 2 Down2

Heading further "down", define "three dots" (therefore), or down2, such that: a  
 @ down2(b,c) = down2(a@b,a@c) ...Or: down(a,down2(b,c)) = down2(down(a,b),down(a,c))  
 ...

(% i66) td\_eq : Td(a,b) = log( down(exp(a),exp(b)) );

$$\text{Td}(a, b) = \log \left( \log \left( {}_0e^{{}_0e^b} + {}_0e^{{}_0e^a} \right) \right) \quad (\text{td\_eq})$$

(% i67) down2(a,b) := radcan( log( down(exp(a),exp(b)) ) );

$$\text{down2}(a, b) := \text{radcan} \left( \log \left( \text{down} \left( \exp(a), \exp(b) \right) \right) \right) \quad (\% \text{ o67})$$

(% i68) down2(down(a,b),down(a,c));

$$\log \left( \log \left( {}_0e^{{}_0e^c} + {}_0e^{{}_0e^b} \right) + {}_0e^a \right) \quad (\% \text{ o68})$$

yes!down2-identity would have to be at  $\infty + i\pi$ .

(% i69) down2(x,x);

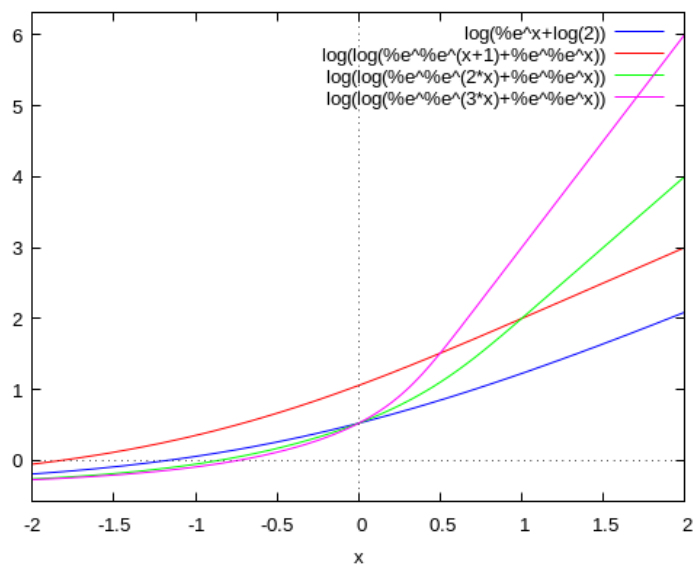
$$\log({}_0e^x + \log(2)) \quad (\% \text{ o69})$$

(% i70) diff(down2(a,b),a);

$$\frac{{}_0e^{{}_0e^a + a}}{\left( {}_0e^{{}_0e^b} + {}_0e^{{}_0e^a} \right) \log \left( {}_0e^{{}_0e^b} + {}_0e^{{}_0e^a} \right)} \quad (\% \text{ o70})$$

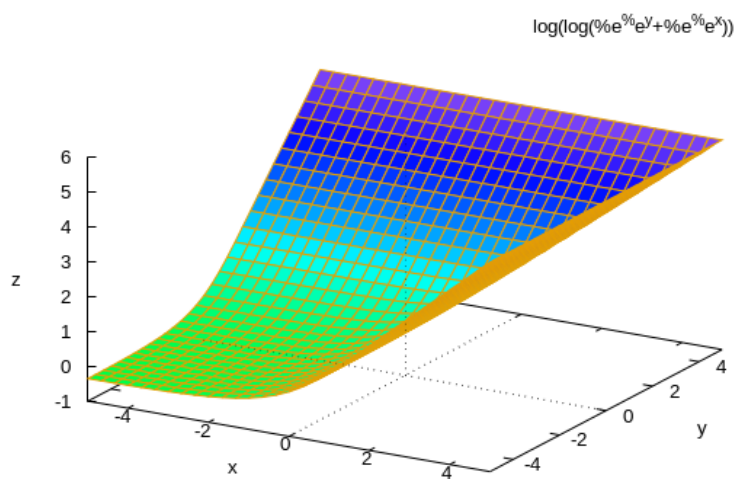
Let's take a look...

(% i71) wxplot2d ([down2(x,x),down2(x,x+1),down2(x,2\*x),down2(x,3\*x)], [x, -2, 2])\$



(% t71)

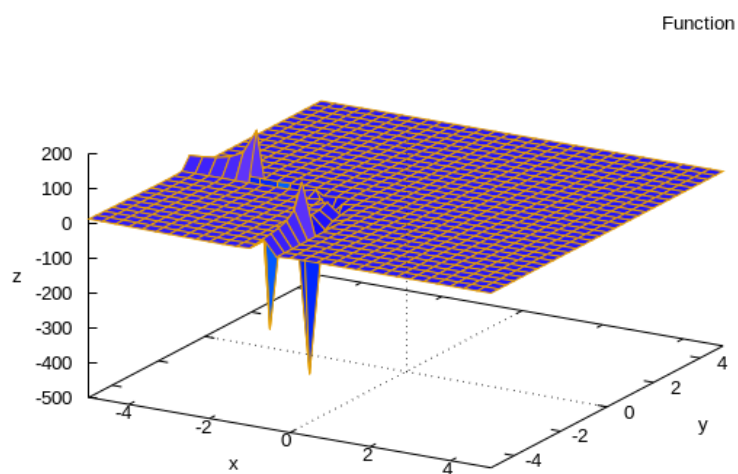
(% i72) wxplot3d (down2(x,y), [x, -5, 5], [y, -5, 5])\$



(% t72)

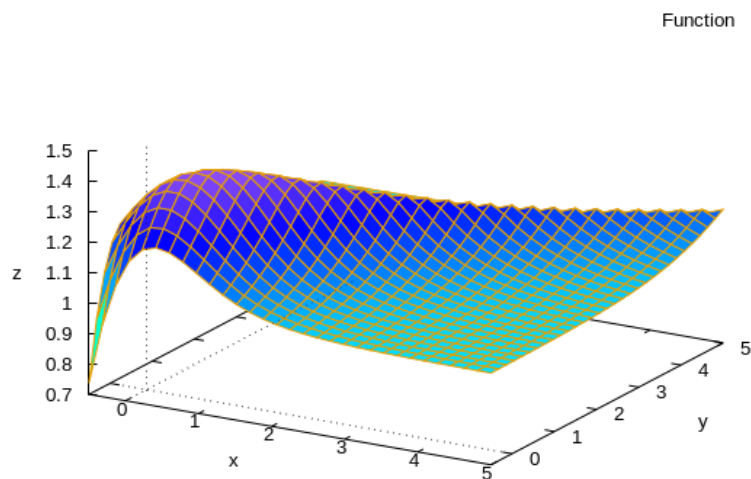
Comparing  $\text{down}(x,y)$  with  $\text{down2}(x,y)$  :

(% i73) wxplot3d (down(x,y)/down2(x,y), [x, -5, 5], [y, -5, 5])\$



(% t73)

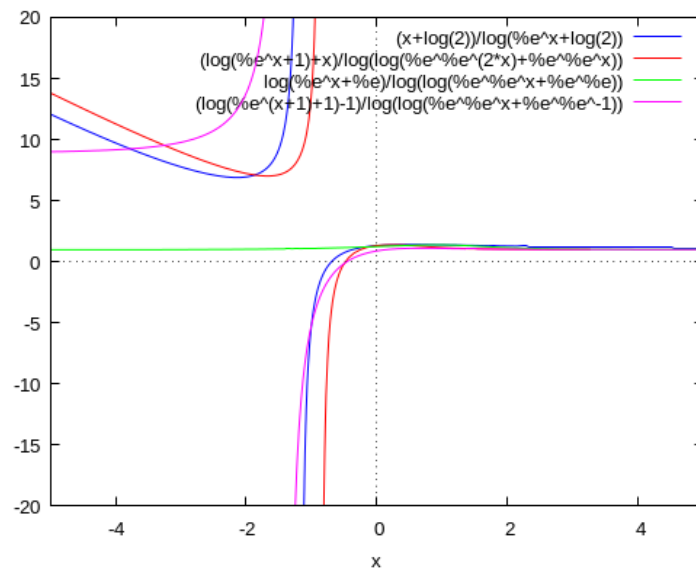
(% i74) wxplot3d (down(x,y)/down2(x,y), [x, -0.5, 5], [y, -0.5, 5])\$



(% t74)

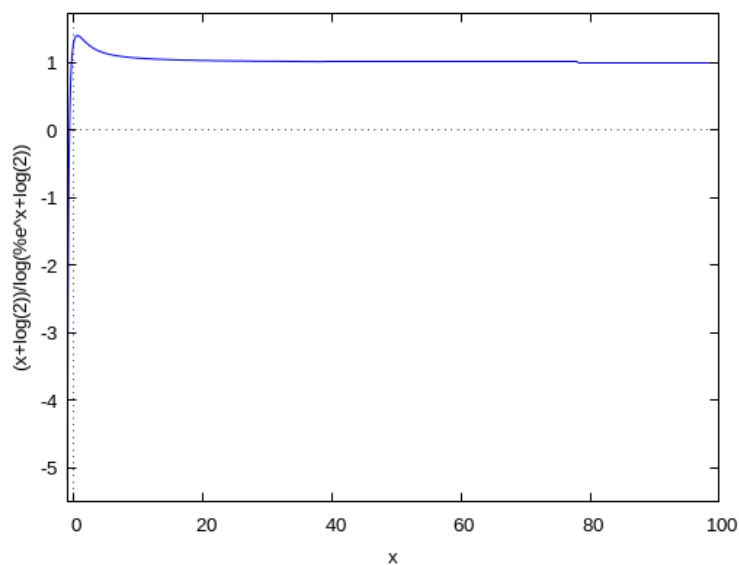
(% i75) wxplot2d ([down(x,x)/down2(x,x), down(x,2\*x)/down2(x,2\*x),down(x,1)/down2(x,1),down(x,-1)/down2(x,-1)], [x, -5, 5], [y, -20, 20])\$

*plot2d : expression evaluate on non-numeric values somewhere in plotting range.*



(% t75)

```
(% i76) wxplot2d (down(x,x)/down2(x,x), [x, -1, 100])$
```



(% t76)

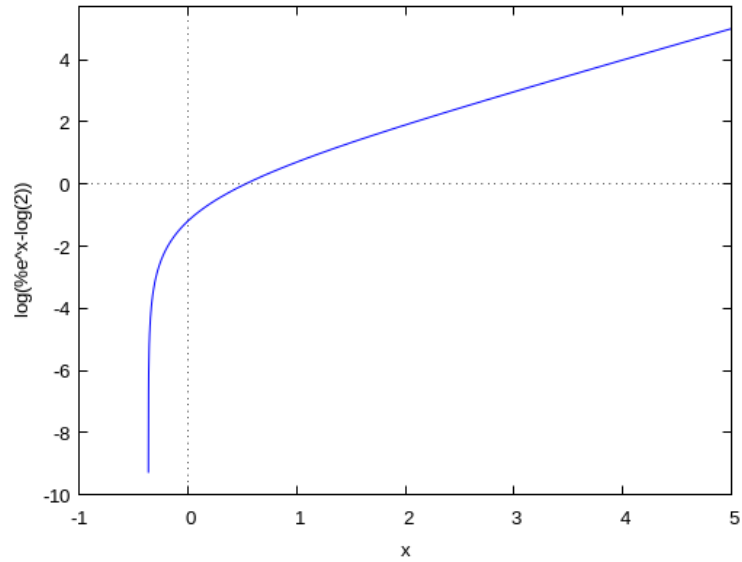
Using down2\_div (or down\_min) for the "ratio"...

```
(% i78) down_min(down(x,x),down2(x,x));
wxplot2d (down_min(down(x,x),down2(x,x)), [x, -1, 5])$
```

$\log(e^x - \log(2))$  plot2d : expression evaluatest on non-numeric values somewhere in plotting range.

(% o77)





(% t78)

### 3 Up

Let's take a look in the "up" direction for an operation which is distributive over multiplication. Defining "star" (or "up"):  $a * (b \cdot c) = (a * b) \cdot (a * c)$  ...Or:  $\text{up}(a, b \cdot c) = \text{up}(a, b) \cdot \text{up}(a, c)$  ...

(% i79) `str_eq : Str(a,b) = exp(log(a)*log(b));`

$$\text{Str}(a, b) = e^{\log(a) \log(b)} \quad (\text{str\_eq})$$

(% i80) `up(a,b) := radcan( exp(log(a)*log(b)) );`

$$\text{up}(a, b) := \text{radcan}(\exp(\log(a) \log(b))) \quad (\% \text{ o80})$$

(% i81) `up(a,b);`

$$a^{\log(b)} \quad (\% \text{ o81})$$

(% i82) `up(a,b*c);`

$$a^{\log(c) + \log(b)} \quad (\% \text{ o82})$$

```
(% i83) up(a,b) * up(a,c);
```

$$a^{\log(c) + \log(b)} \quad (\% \text{ o83})$$

which is the same as the former. So, yes! Check with "down"-like operation:

```
(% i84) log( up(exp(a),exp(b)) );
```

$$ab \quad (\% \text{ o84})$$

yes! up(a,b) could also be expressed with b as the base:

```
(% i88) is( up(a,b) = b ^ log(a) );
         is( up(a,b) = a ^ log(b) );
         is( up(b,a) = a ^ log(b) );
         is( up(a,b) = up(b,a) );
```

$$\text{false} \quad (\% \text{ o85})$$

$$\text{true} \quad (\% \text{ o86})$$

$$\text{true} \quad (\% \text{ o87})$$

$$\text{true} \quad (\% \text{ o88})$$

... although it seems Maxima has trouble (it enforces some type of alphabetization?) with that realization in the present form...? So one advantage of up(a,b), over "naive" exponentiation (a<sup>b</sup> or b<sup>a</sup>), can be seen immediately: namely commutativity... up(a,b)=up(b,a). Identity for up (or up\_times):

```
(% i89) up(e,a);
```

$$a \quad (\% \text{ o89})$$

... the base e. Some special values:

```
(% i91) up(x,x);
         x ^ log(x);
```

$$e^{\log(x)^2} \quad (\% \text{ o90})$$

$$x^{\log(x)} \quad (\% \text{ o91})$$

```
(% i94) diff(up(a,b),a);
        diff(up(a,b),b);
        diff(diff(up(a,b),a),b);
```

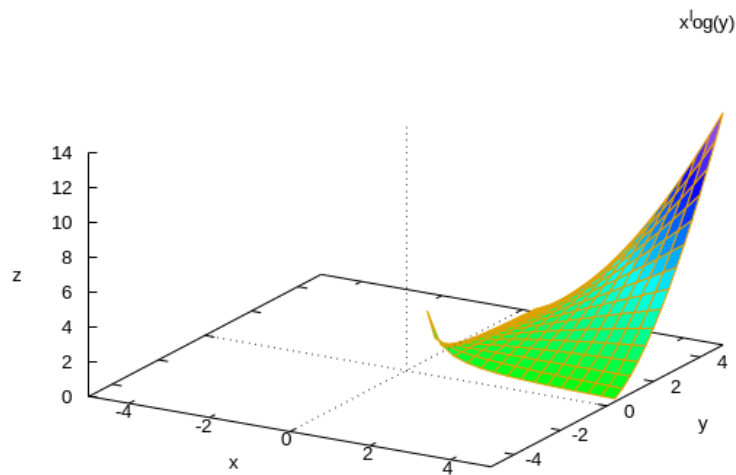
$$a^{\log(b)-1} \log(b) \quad (\% \text{ o92})$$

$$\frac{a^{\log(b)} \log(a)}{b} \quad (\% \text{ o93})$$

$$\frac{a^{\log(b)-1} \log(a) \log(b)}{b} + \frac{a^{\log(b)-1}}{b} \quad (\% \text{ o94})$$

Let's take a look:

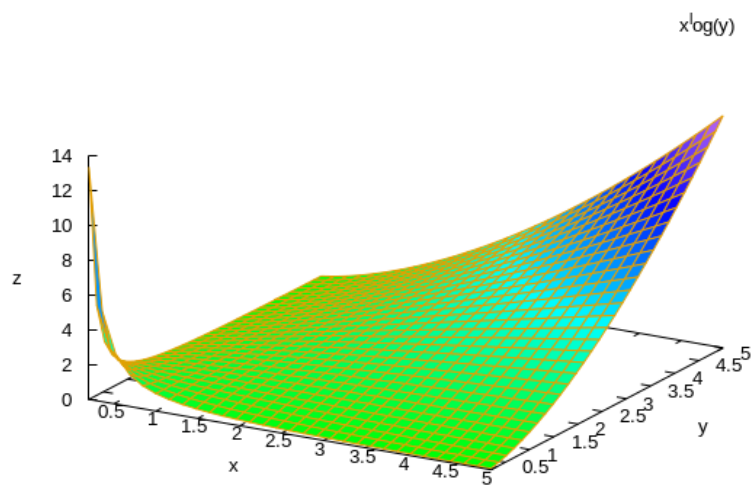
```
(% i95) wxplot3d (up(x,y), [x, -5, 5], [y, -5, 5])$
```



(% t95)

The singularity at the origin comes as no surprise. And for finite, real values,  $x$  and  $y$  should remain  $> 0$ .

(% i96) wxplot3d (up(x,y), [x, 0.2, 5], [y, 0.2, 5])\$



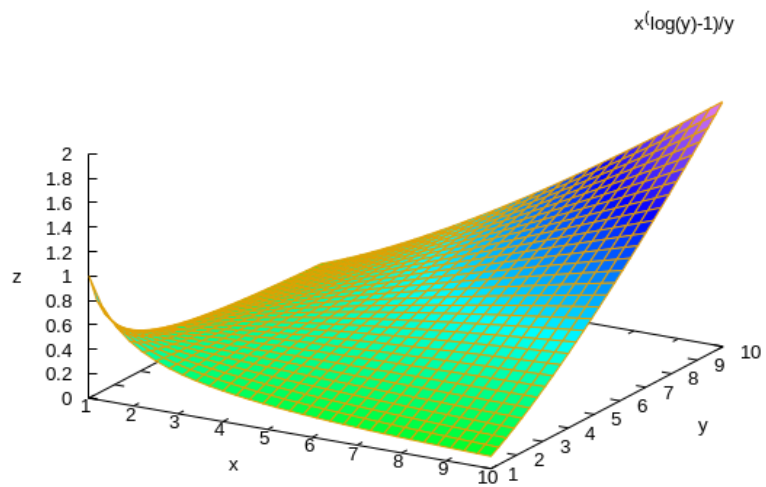
(% t96)

Compare with x.y :

(% i98) up(x,y)/(x\*y);  
wxplot3d (up(x,y)/(x\*y), [x, 1, 10], [y, 1, 10])\$

$$\frac{x^{\log(y)-1}}{y}$$

(% o97)



(% t98)

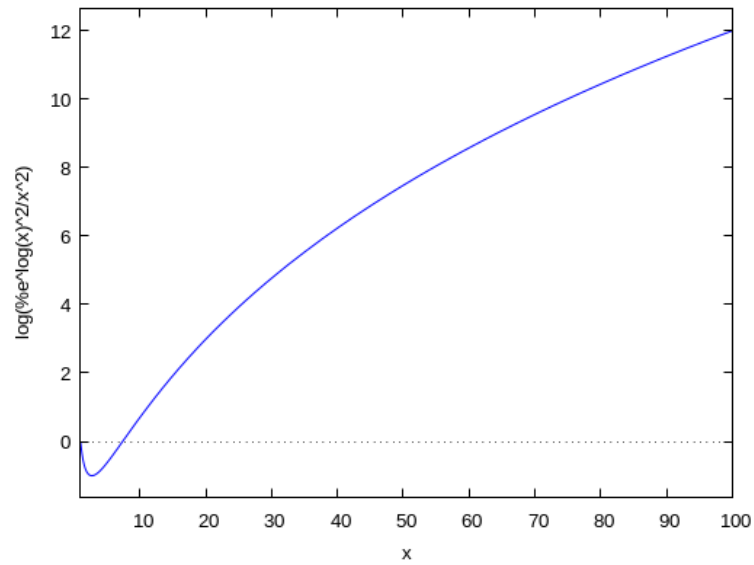
```
(%      up(x,x)/(x*x);
i101)  x^(log(x)-2);
        wxplot2d (log(up(x,x)/(x*x)), [x, 1, 100])$
```

$$\frac{e^{\log(x)^2}}{x^2}$$

(% o99)

$$x^{\log(x)-2}$$

(% o100)



(% t101)

Derivative of this ratio:

```
(%      factor(diff(x^ (log(x)-2),x,1));
i102)
```

$$2x^{\log(x)-3}(\log(x) - 1)$$

(% o102)

Minimum obviously at  $x=e$ :

```
(%      up(e,e);
i104)  find_root(diff(x^ (log(x)-2),x,1), x, 1, 10);
```

$$e$$

(% o103)

$$2.718281828459045$$

(% o104)

Also,  $up(x,x)$  does eventually win over  $x^2 \dots$  at  $x=e^2$ , where  $up(x,x)$  is ...

```
(%      up(exp(2),exp(2));
i106)  float(%), numer;
```

$$e^4$$

(% o105)

54.59815003314423 (% o106)

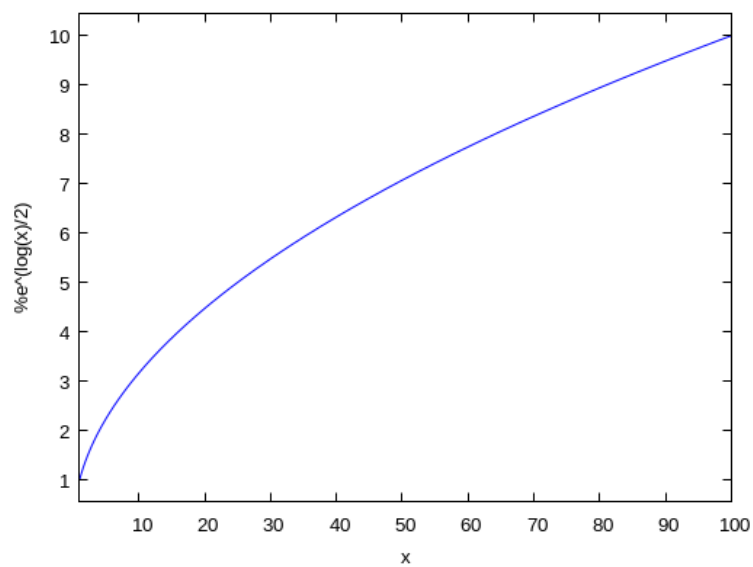
Using up\_div for the "ratio"...

```
(%
i108)  exp(log(up(x,x))/log(x*x));
        radcan(%);
```

$e^{\frac{\log(x)}{2}}$  (% o107)

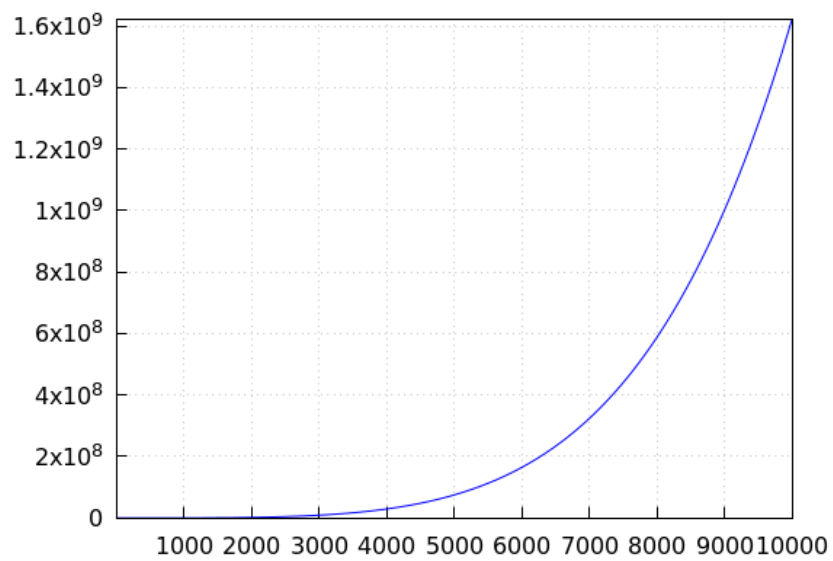
$\sqrt{x}$  (% o108)

```
(%
i109)  wxplot2d (exp(log(up(x,x))/log(x*x)), [x, 1, 100])$
```



(% t109)

```
(% wxdraw2d( grid = true, nticks = 1000,
i110) parametric( t^ 2 , up(t,t) ,
t,1,100 ) )$
```

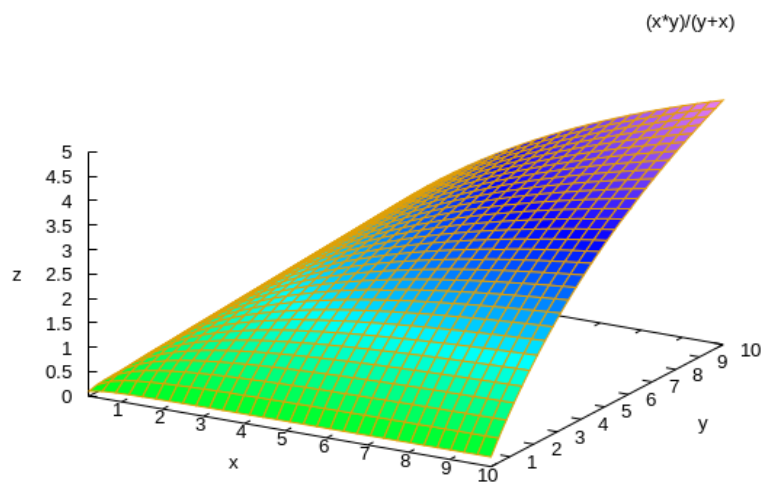


(% t110)

And x,y with x+y :



```
(% wxplot3d (x*y/(x+y), [x, 0.2, 10], [y, 0.2, 10])$
i111)
```



```
(% t111)
```

Possible relations to polylogarithms?:

```
(% integrate(log(t)/(t-1), t, 1, x);
i115) integrate(log(t)/(t+1), t, 1, x);
integrate(log(t)/t, t, 1, x);
integrate(log(y)/t, t, 1, x);
```

"Is "

$$x - 1$$

" positive, negative or zero?"

*positive;*

$$\log(1-x)\log(x) + \operatorname{li}_2(x) - \frac{\pi^2}{6} \quad (\% \text{ o112})$$

"Is "

$$x - 1$$

" positive, negative or zero?"

*positive;*

$$\log (x) \log (x+1)+\operatorname{li}_2(-x)+\frac{\pi^2}{12} \quad (\% \text{ o113})$$

"Is "

$$x-1$$

" positive, negative or zero?"

*positive;*

$$\frac{\log (x)^2}{2} \quad (\% \text{ o114})$$

"Is "

$$x-1$$

" positive, negative or zero?"

*positive;*

$$\log (x) \log (y) \quad (\% \text{ o115})$$

### 3.1 Up with generalized base

$$\mathbf{u\_b(x,y)} = \mathbf{b^{(\log\_b(x)*\log\_b(y))}} \dots$$

(% logb(b,x) := log(x)/log(b);  
i116)

$$\log b(b, x):=\frac{\log (x)}{\log (b)} \quad (\% \text{ o116})$$

(% bup(b,x,y) := radcan( b^ (logb(b,x)\*logb(b,y)) );  
i117)

$$\mathbf{bup(b,x,y)}:=\mathbf{radcan}\left(b^{\log b(b, x) \log b(b, y)}\right) \quad (\% \text{ o117})$$

```
(%      bup(b,x,y);
i119)   exp(log(x)*log(y)/log(b));
```

$$x^{\frac{\log(y)}{\log(b)}} \quad (\% \text{ o118})$$

$$\%e^{\frac{\log(x)\log(y)}{\log(b)}} \quad (\% \text{ o119})$$

```
(%      dbup(b,x,y) := diff(bup(b,x,y),b);
i121)   dbup(b,x,y);
```

$$\text{dbup}(b, x, y) := \frac{d}{db} \text{bup}(b, x, y) \quad (\% \text{ o120})$$

$$-\frac{x^{\frac{\log(y)}{\log(b)}} \log(x) \log(y)}{b \log(b)^2} \quad (\% \text{ o121})$$

```
(%      dbupdx(b,x,y) := diff(bup(b,x,y),x);
i124)   dbupdx(b,x,y);
        dbup(b,x,y) / dbupdx(b,x,y);
```

$$\text{dbupdx}(b, x, y) := \frac{d}{dx} \text{bup}(b, x, y) \quad (\% \text{ o122})$$

$$\frac{x^{\frac{\log(y)}{\log(b)}-1} \log(y)}{\log(b)} \quad (\% \text{ o123})$$

$$-\frac{x \log(x)}{b \log(b)} \quad (\% \text{ o124})$$

```
(%      dbup(b,x,x);
i127)   dbupdx(b,x,x);
        dbup(b,x,x) / dbupdx(b,x,x);
```

$$-\frac{\%e^{\frac{\log(x)^2}{\log(b)}} \log(x)^2}{b \log(b)^2} \quad (\% \text{ o125})$$

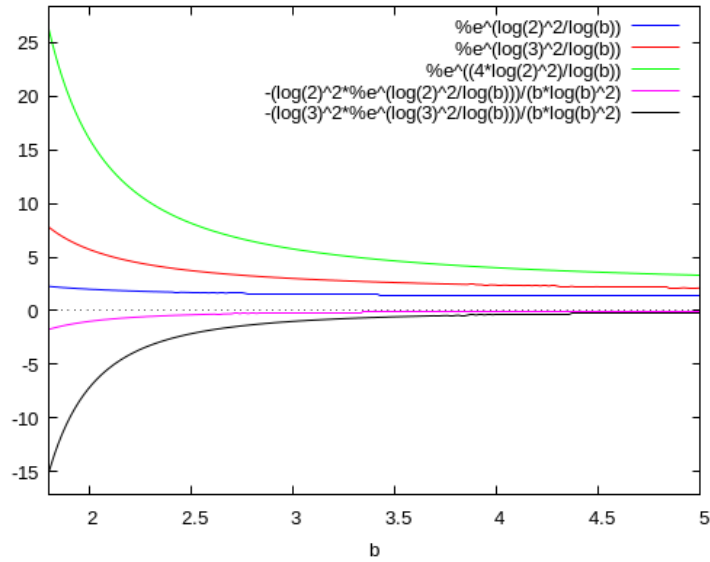
$$\frac{2e^{\frac{\log(x)^2}{\log(b)}} \log(x)}{\log(b)x} \quad (\% \text{ o126})$$

$$-\frac{x \log(x)}{2b \log(b)} \quad (\% \text{ o127})$$

```
(%      bup(exp(beta)*b,exp(alpha)*x,exp(alpha)*y);
i128)
```

$$b^{\frac{\alpha^2 + (\log(y) + \log(x))\alpha + \log(x) \log(y)}{\beta^2 + 2 \log(b)\beta + \log(b)^2}} x^{\frac{(\alpha + \log(y))\beta}{\beta^2 + 2 \log(b)\beta + \log(b)^2}} y^{\frac{\alpha\beta}{\beta^2 + 2 \log(b)\beta + \log(b)^2}} e^{\frac{\alpha^2\beta}{\beta^2 + 2 \log(b)\beta + \log(b)^2}} \quad (\% \text{ o128})$$

```
(%      wxplot2d ([bup(b,2,2),bup(b,3,3),bup(b,4,4),dbup(b,2,2),dbup(b,3,3)], [b, 1.8,
i129) 5])$
```



(% t129)

### 3.2 Up differentiation

To avoid problems with alphabetical preferences...

```
(%      up(x,h);
i131)   up(a,h);

hlog(x)                                     (% o130)
```

```
alog(h)                                     (% o131)
```

we'll use 'a' as the independent variable here...Working with up\_/, up\_+ as times, up\_- as divide, and the new "up\_+"-identity, h->1...

```
(%      Uf(a,h) := f(a*h) / f(a) ;
i135)   Uf(a,h);
        Uf_Ua(a) := radcan( limit( exp( log(Uf(a,h)) / log(h) ) , h, 1, plus) );
        Uf_Ua(a);
```

```
Uf(a,h) := f(a*h) / f(a)                     (% o132)
```

```
f(a*h) / f(a)                               (% o133)
```

```
Uf_Ua(a) := radcan( limit( exp( log(Uf(a,h)) / log(h) ) , h, 1, plus) ) (% o134)
```

```
lim_{h->1+} f(a*h)^{1/log(h)} / f(a)^{1/log(h)} (% o135)
```

```
(%      Uf_Ua(a) := ratsimp( limit( Uf(a,h)^ (1 / log(h)) , h, 1, plus));
i137)   Uf_Ua(a);
```

```
Uf_Ua(a) := ratsimp( limit( Uf(a,h)^{1/log(h)} , h, 1, plus) ) (% o136)
```

```
lim_{h->1+} (f(a*h) / f(a))^{1/log(h)}         (% o137)
```

```
(%      f(a) := n*a + c;
i140)   Uf(a,h);
        Uf_Ua(a);

f(a) := n*a + c                             (% o138)
```

$$\frac{ahn + c}{an + c} \quad (\% \text{ o139})$$

$$\%_0 e^{\frac{an}{an+c}} \quad (\% \text{ o140})$$

```
(%      assume(y>1);
i144)  f(a) := a^ y + c;
        Uf(a,h);
        Uf_ Ua(a);

[y > 1] \quad (\% \text{ o141})
```

$$f(a) := a^y + c \quad (\% \text{ o142})$$

$$\frac{(ah)^y + c}{c + a^y} \quad (\% \text{ o143})$$

$$\%_0 e^{\frac{a^y y}{c+a^y}} \quad (\% \text{ o144})$$

```
(%      f(a) := exp(b*a^ y + c);
i147)  Uf(a,h);
        Uf_ Ua(a);

f(a) := exp(ba^ y + c) \quad (\% \text{ o145})
```

$$\%_0 e^{b(ah)^y - a^y b} \quad (\% \text{ o146})$$

"Is "

$y$

" an "

integer

"?"

*no*;

$$\%e^{a^yby}$$

(% o147)

(%  
i148)       $\text{Uf\_Ua(a)}$ ;

"Is "

*y*

" and "

integer

"?"

*yes*;

$$\%e^{a^yby}$$

(% o148)

(%  
i151)       $f(a) := b^{\wedge} (a^{\wedge} y + c)$ ;  
             $\text{Uf(a,h)}$ ;  
             $\text{Uf\_Ua(a)}$ ;

$$f(a) := b^{a^y+c}$$

(% o149)

$$b^{(ah)^y-a^y}$$

(% o150)

"Is "

*y*

" and "

integer

"?"

*no*;

"Is "

$$y - 1$$

" positive, negative or zero?"

*positive;*

$$e^{a^y \log(b)y}$$

(% o151)

(%  
**i152)**       $\text{Uf\_Ua(a);}$

"Is "

$$y$$

" an "

integer

"?"

*no;*

"Is "

$$y - 1$$

" positive, negative or zero?"

*negative;*

$$e^{a^y \log(b)y}$$

(% o152)

(%  
**i155)**       $f(a) := \exp(b^{\wedge} (a^{\wedge} y + c));$   
             $\text{Uf(a,h);}$   
             $\text{Uf\_Ua(a);}$

$$f(a) := \exp\left(b^{a^y+c}\right)$$

(% o153)



$$\%e^{b^{(ah)^y+c}-b^{c+ay}} \quad (\% \text{ o154})$$

"Is "

$c$

" an "

integer

"?"

*no*;

$$\%e^{a^yb^{c+ay}\log(b)y} \quad (\% \text{ o155})$$

(%      Uf\_ Ua(a);  
**i156)**

"Is "

$c$

" an "

integer

"?"

*yes*;

$$\%e^{a^yb^{c+ay}\log(b)y} \quad (\% \text{ o156})$$

(%      f(a) := g(a) \* p(a);  
**i159)**      Uf(a,h);  
            Uf\_ Ua(a);

$$f(a) := g(a) p(a) \quad (\% \text{ o157})$$

$$\frac{g(ah)p(ah)}{g(a)p(a)} \quad (\% \text{ o158})$$

$$\lim_{h \rightarrow 1} \left( \frac{g(ah)p(ah)}{g(a)p(a)} \right)^{\frac{1}{\log(h)}} \quad (\% \text{ o159})$$

So the up-derivative maintains not only the extended distributivity, but also the property that the exponential function keeps its special role:  $U\_Ua(\mathfrak{e}\hat{a}) = \mathfrak{e}\hat{a}$ .

```
(%      f(a) := b^ exp(a^ y + c);
i162)   Uf(a,h);
        Uf_ Ua(a);
```

$$f(a) := b^{\exp(a^y + c)} \quad (\% \text{ o160})$$

$$b^{\%e^{(ah)^y + c} - \%e^{c + a^y}} \quad (\% \text{ o161})$$

```
"Is "
      c
" an "
      integer
"? "
      no;
```

$$\%e^{a^y \log(b)y} \%e^{c + a^y} \quad (\% \text{ o162})$$

```
(%      Uf_ Ua(a);
i163)
"Is "
      c
" an "
      integer
"? "
      yes;
```

$$\%e^{a^y \log(b)y} \%e^{c+a^y} \quad (\% \text{ o163})$$

```
(%      f(a) := log(a);
i166)   Uf(a,h);
        Uf_ Ua(a);

f(a) := log(a) \quad (\% \text{ o164})
```

$$\frac{\log(ah)}{\log(a)} \quad (\% \text{ o165})$$

$$\%e^{\frac{1}{\log(a)}} \quad (\% \text{ o166})$$

```
(%      kill(f);
i167)

done \quad (\% \text{ o167})
```

Alternative expressions for Uf\_ Ua(a) (with a=exp(z)):

```
(%      a = exp(z);
i172)   exp('diff( log(f(exp(z))) ,z,1));
        exp(diff( log(f(exp(z))) ,z,1));
        exp( a * diff(f(a),a,1) / f(a) );
        exp('diff( log(f(a)) ,log(a),1));

a = \%e^z \quad (\% \text{ o168})
```

$$\%e^{\frac{d}{dz} \log(f(\%e^z))} \quad (\% \text{ o169})$$

$$\%e^{\frac{\frac{d}{da} f(\%e^z)}{f(\%e^z)}} \quad (\% \text{ o170})$$

$$\%e^{\frac{a \left( \frac{d}{da} f(a) \right)}{f(a)}} \quad (\% \text{ o171})$$

$$\%e^{\frac{d}{d \log(a)} \log(f(a))} \quad (\% \text{ o172})$$

This leads to an alternative expression for the "normal" derivative,  $d/dx$ , in terms of the down derivative:

```
(%      exp(x + 'Df_Dx(x) - f(x));
i176)   z = log(x); g(x) = log(f(x));
        exp('Dg_Dz(x));
```

$$\%e^{Df\_Dx(x)-f(x)+x} \quad (\% \text{ o173})$$

$$z = \log(x) \quad (\% \text{ o174})$$

$$g(x) = \log(f(x)) \quad (\% \text{ o175})$$

$$\%e^{Dg\_Dz(x)} \quad (\% \text{ o176})$$

## 4 Up2

Now to be distributive over "up":  $\text{up2}(a, \text{up}(b,c)) = \text{up}(\text{up2}(a,b), \text{up2}(a,c)) \dots$

```
(%      up2(a,b) := radcan( exp( up(log(a),log(b)) ) );
i177)
```

$$\text{up2}(a, b) := \text{radcan}(\exp(\text{up}(\log(a), \log(b)))) \quad (\% \text{ o177})$$

```
(%      up(up2(a,b),up2(a,c));
i179)   up2(a,up(b,c));
```

$$\%e^{\log(a)^{\log(\log(c))+\log(\log(b))}} \quad (\% \text{ o178})$$

$$\%e^{\log(a)^{\log(\log(c))+\log(\log(b))}} \quad (\% \text{ o179})$$

Bingo! And obviously also commutative, unlike say, tetration ( $\hat{y}^x$  or  $\hat{x}^y$ ), which is neither. Identity for up2 (or up2\_times):

```
(%      up2(e^ e,a);
i180)
```

$$a \quad (\% \text{ o180})$$

... or base to the power of the base. So one can imagine that for up3 it would be  $e^{e^e}$  ... etc.

```
(%      diff(up2(a,b),a);
i181)
```

$$\frac{\log(a)^{\log(\log(b))-1} e^{\log(a)^{\log(\log(b))}} \log(\log(b))}{a} \quad (\% \text{ o181})$$

```
(%      up2(x,x);
i184)  exp(log(x)^ log(x));
      x^ (log(x)^ (log(x)-1));
```

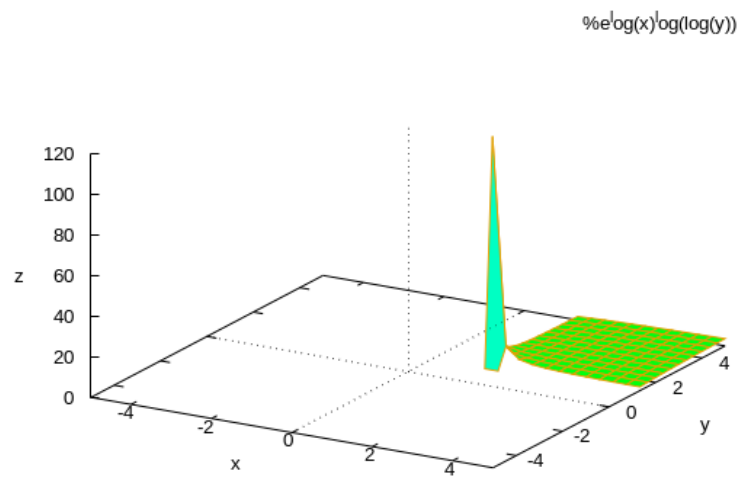
$$e^{e^{\log(\log(x))^2}} \quad (\% \text{ o182})$$

$$e^{\log(x)^{\log(x)}} \quad (\% \text{ o183})$$

$$x^{\log(x)^{\log(x)-1}} \quad (\% \text{ o184})$$

which is  $\text{up}(x,x)[\log(x)(\log(x)-2)]$ . Taking a look...

```
(% wxplot3d (up2(x,y), [x, -5, 5], [y, -5, 5])$
i185)
```

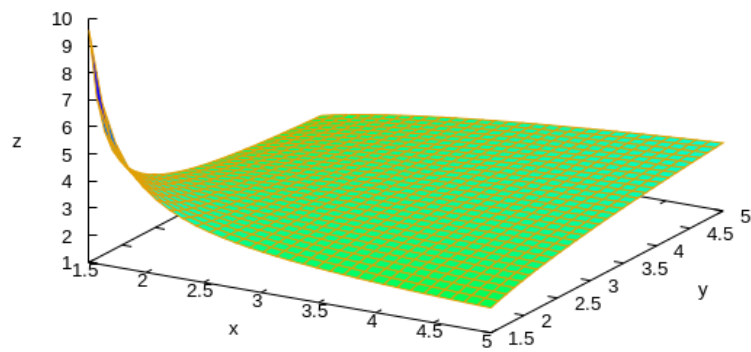


```
(% t185)
```

Problems now at  $x$  or  $y = 1$ .

```
(% wxplot3d (up2(x,y), [x, 1.5, 5], [y, 1.5, 5])$
i186)
```

```
%elog(x)log(log(y))
```

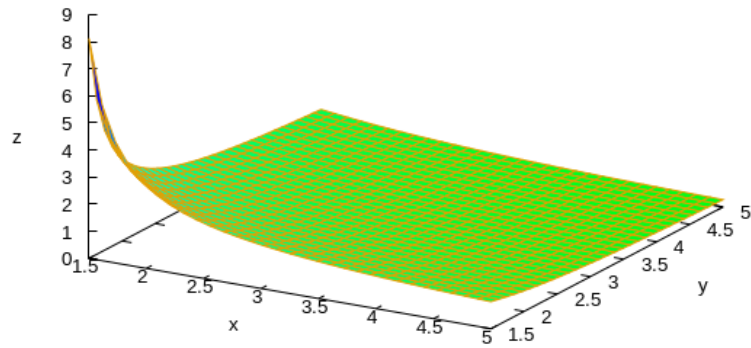


```
(% t186)
```

Comparing up2(x,y) with up(x,y) :

```
(% wxplot3d (up2(x,y)/up(x,y), [x, 1.5, 5], [y, 1.5, 5])$
i187)
```

```
%elog(x)log(log(y))/xlog(y)
```



```
(% t187)
```

```
(% up2(x,x)/up(x,x);
i191) log(up2(x,x)/up(x,x));
log(x)^log(log(x))-log(x)^2;
wxplot2d (log(up2(x,x)/up(x,x)), [x, 1.5, 5000])$
```

```
%e%elog(log(x))2-log(x)2
```

(% o188)

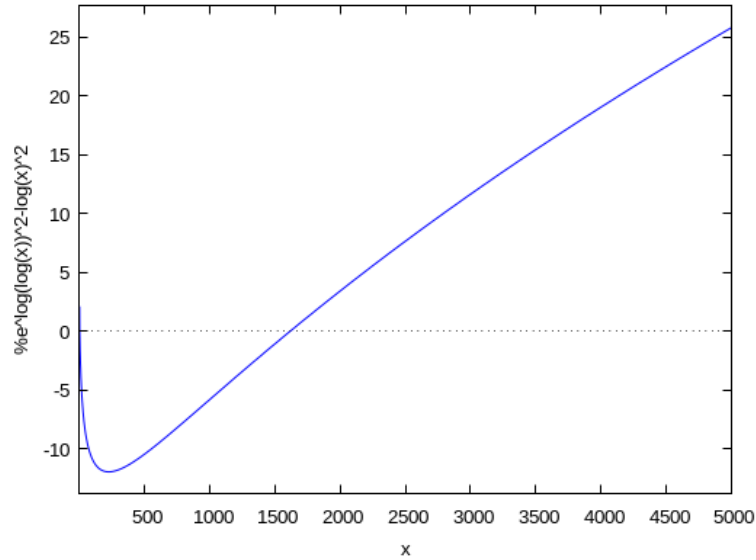
```
%elog(log(x))2-log(x)2
```

(% o189)

```
log(x)log(log(x))-log(x)2
```

(% o190)





(% t191)

```
(%      factor(diff(exp(log(x)^ log(log(x))-log(x)^ 2),x,1));
i192)
```

$$\frac{2e^{\log(x)\log(\log(x))-\log(x)^2} \left( \log(x)^{\log(\log(x))} \log(\log(x)) - \log(x)^2 \right)}{x \log(x)} \quad (\% \text{ o192})$$

```
(%      find_root('(%), x, 100, 300);
i194)  up2(%,%);
```

225.248329275192rat : replaced 17.3688717996767by 103253618/5944751 = 17.36887179967672  
(% o193)

$$e^{\frac{103253618}{5944751}} \quad (\% \text{ o194})$$

Note also that up2(x,x) does eventually win over up(x,x)... at  $x=e^{\hat{e}^2}$ , where up2(x,x) is...

```
(%      up2(exp(exp(2)),exp(exp(2)));
i196)  float(%), numer;
```

$$e^{e^4} \quad (\% \text{ o195})$$

5.14843556263455710<sup>23</sup> (% o196)

Holy Avogadro, batman! Using up2\_div for the "ratio"...

```
(%
i198) up_div(a,b) := radcan( exp(log(a) / log(b)) );
      up2_div(a,b) := radcan( exp( up_div(log(a),log(b)) ) );
```

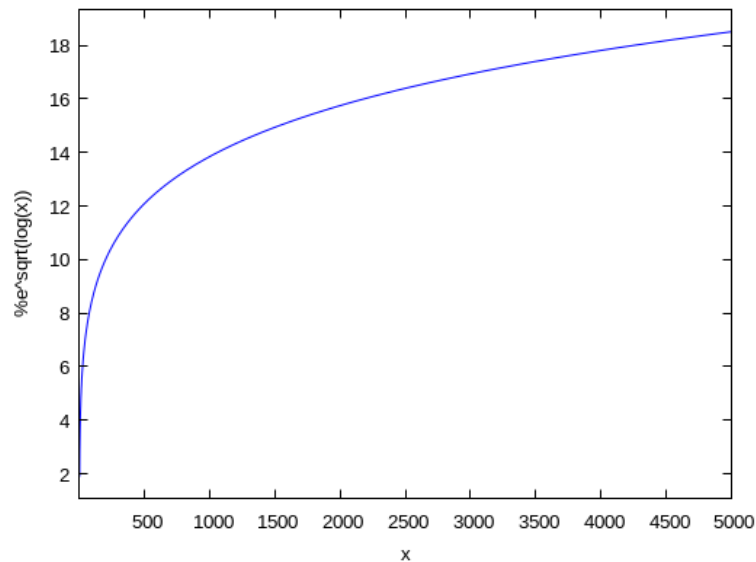
$$\text{up\_div}(a,b) := \text{radcan} \left( \exp \left( \frac{\log(a)}{\log(b)} \right) \right) \quad (\% \text{ o197})$$

$$\text{up2\_div}(a,b) := \text{radcan} \left( \exp \left( \text{up\_div}(\log(a), \log(b)) \right) \right) \quad (\% \text{ o198})$$

```
(%
i199) up2_div(up2(x,x),up(x,x));
```

$$e^{\sqrt{\log(x)}} \quad (\% \text{ o199})$$

```
(%
i200) wxplot2d (up2_div(up2(x,x),up(x,x)), [x, 1.5, 5000])$
```

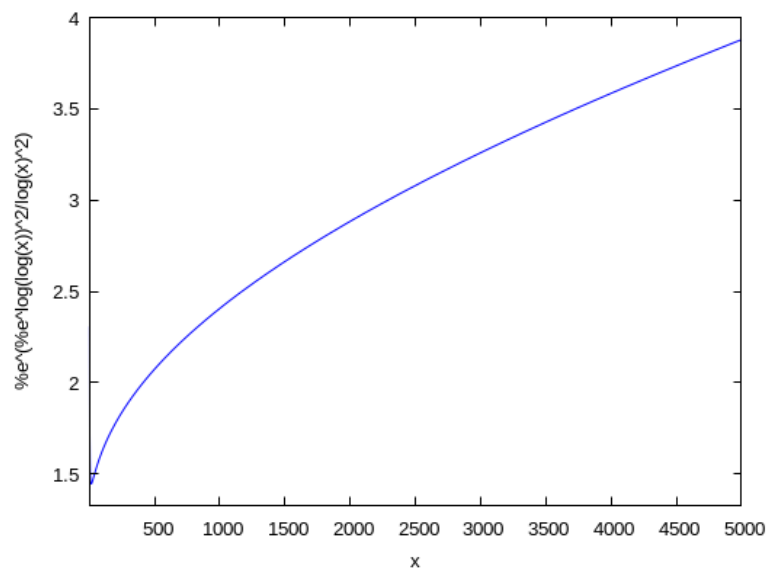


(% t200)

```
(%
i203) up_div(up2(x,x),up(x,x));
exp(log(x)^(log(log(x))-2));
wxplot2d (up_div(up2(x,x),up(x,x)), [x, 3, 5000])$
```

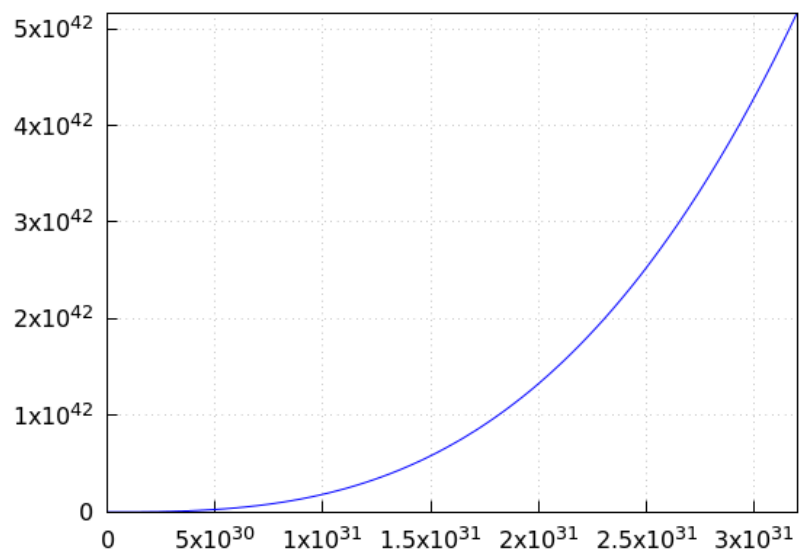
```
%e  $\frac{e^{\log(\log(x))^2}}{\log(x)^2}$  (% o201)
```

```
%e  $\log(x)^{\log(\log(x))-2}$  (% o202)
```



```
(% t203)
```

```
(% wxdraw2d( grid = true, nticks = 1000,
i204) parametric( up(t,t) , up2(t,t) ,
t,1.5,5000 ) )$
```



(% t204)