

# Computer algebra independent integration tests

0\_Independent\_test\_suites/Hebisch\_Problems

Nasser M. Abbasi

November 25, 2018

Compiled on November 25, 2018 at 10:22pm

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>detailed summary tables of results</b>	<b>9</b>
<b>3</b>	<b>Listing of integrals</b>	<b>11</b>
<b>4</b>	<b>Listing of Grading functions</b>	<b>25</b>

# 1 Introduction

This report gives the result of running the computer algebra independent integration problems. The listing of the problems are maintained by and can be downloaded from Albert Rich Rubi web site.

## 1.1 Listing of CAS systems tested

The following systems were tested at this time.

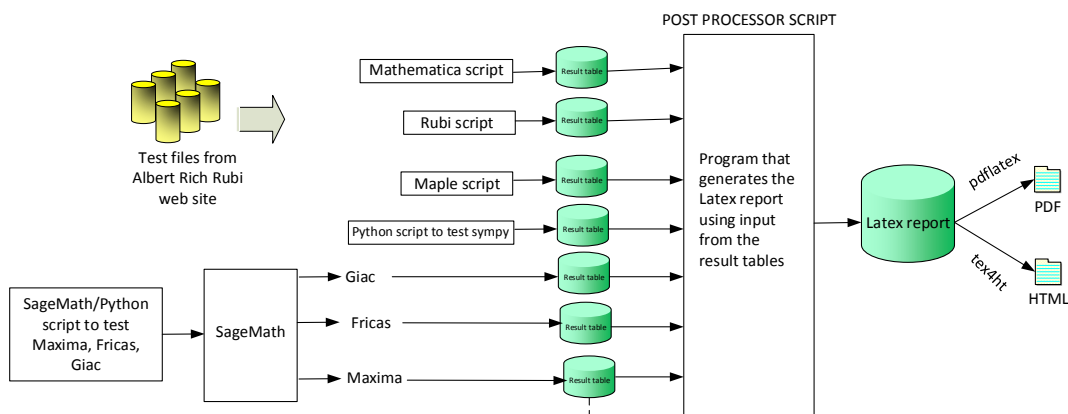
1. Mathematica 11.3 (64 bit).
2. Rubi 4.15.2 in Mathematica 11.3.
3. Rubi in Sympy (Version 1.3) under Python 3.7.0 using Anaconda distribution.
4. Maple 2018.1 (64 bit).
5. Maxima 5.41 Using Lisp ECL 16.1.2.
6. Fricas 1.3.4.
7. Sympy 1.3 under Python 3.7.0 using Anaconda distribution.
8. Giac/Xcas 1.4.9.

Maxima, Fricas and Giac/Xcas were called from inside SageMath version 8.3. This was done using SageMath integrate command by changing the name of the algorithm to use the different CAS systems.

Sympy was called directly using Python. Rubi in Sympy was also called directly using sympy 1.3 in python.

## 1.2 Design of the test system

The following diagram gives a high level view of the current test build system.



One record (line) per one integral result. The line is comma delimited. It contains 12 fields. This is description of each record (line)

1. integer, the problem number.
2. integer. 0 or 1 for failed or passed. (this is not the grade field)
3. integer. Leaf size of result.
4. integer. Leaf size of the optimal antiderivative.
5. number. CPU time used to solve this integral. 0 if failed.
6. string. The integral in Latex format
7. string. The input used in CAS own syntx.
8. string. The result (antiderivative) produced by CAS in Latex format
9. string. The optimal antiderivative in Latex format.
10. integer. 0 or 1. Indicates if problem has known antiderivative or not
11. String. The result (antiderivative) in CAS own syntax.
12. String. The grade of the antiderivative. Can be "A", "B", "C", or "F"

**High level overview of the CAS independent integration test build system**

Nasser M. Abbasi  
June 22, 2018

## 1.3 Timing

The command `AbsoluteTiming[ ]` was used in Mathematica to obtain the elapsed time for each integrate call. In Maple, the command `Usage` was used as in the following example

```
cpu_time := Usage(assign ('result_of _int', int(expr, x)), output='realtime')
```

For all other CAS systems, the elapsed time to complete each integral was found by taking the difference between the time after the call has completed from the time before the call was made. This was done using Python's `time.time()` call.

All elapsed times shown are in seconds. A time limit of 3 minutes was used for each integral. If the `integrate` command did not complete within this time limit, the integral was aborted and considered to have failed and assigned an F grade. The time used by failed integrals due to time out is not counted in the final statistics.

## 1.4 Verification

A verification phase was applied on the result of integration for Rubi and Mathematica. Future version of this report will implement verification for the other CAS systems. For the integrals whose result was not run through a verification phase, it is assumed that the antiderivative produced was correct.

Verification phase has 3 minutes time out. An integral whose result was not verified could still be correct. Further investigation is needed on those integrals which failed verifications. Such integrals are marked in the summary table below and also in each integral separate section so they are easy to identify and locate.

## 1.5 Important notes about some of the results

Important note about Maxima results Since these integrals are run in a batch mode, using an automated script, and by using `sagemath` (SageMath uses Maxima), then any integral where Maxima needs an interactive response from the user to answer a question during evaluation of the integral in order to complete the integration, will fail and is counted as failed.

The exception raised is `ValueError`. Therefore Maxima result below is lower than what could result if Maxima was run directly and each question Maxima asks was answered correctly.

The percentage of such failures were not counted for each test file, but for an example, for the Timofeev test file, there were about 30 such integrals out of total 705, or about 4 percent. This percentage can be higher or lower depending on the specific input test file.

Such integrals can be indentified by looking at the output of the integration in each section for Maxima. If the output was an exception `ValueError` then this is most likely due to this reason.

Maxima `integrate` was run using SageMath with the following settings set by default

```
'besselexpand : true'
'display2d : false'
'domain : complex'
'keepfloat : true'
'load(to_poly_solve)'
'load(simplify_sum)'
'load(abs_integrate)' 'load(diag)'
```

SageMath loading of Maxima `abs_integrate` was found to cause some problem. So the following code was added to disable this effect.

```
from sage.interfaces.maxima_lib import maxima_lib
maxima_lib.set('extra_definite_integration_methods', '[]')
maxima_lib.set('extra_integration_methods', '[]')
```

See <https://ask.sagemath.org/question/43088/integrate-results-that-are-different-from-using-maxima/> for reference.

Important note about FriCAS and Giac/XCAS results There are Few integrals which failed due to SageMath not able to translate the result back to SageMath syntax and not because these CAS system were not able to do the integrations.

These will fail With error `Exception raised: NotImplementedError`

The number of such cases seems to be very small. About 1 or 2 percent of all integrals.

Hopefully the next version of SageMath will have complete translation of FriCAS and XCAS syntax and I will re-run all the tests again when this happens.

Important note about finding leaf size of antiderivative For Mathematica, Rubi and Maple, the builtin system function `LeafSize` is used to find the leaf size of each antiderivative.

The other CAS systems (SageMath and Sympy) do not have special builtin function for this purpose at this time. Therefore the leaf size is determined as follows.

For Fricas, Giac and Maxima (all called via sagemath) the following code is used

```
#see https://stackoverflow.com/questions/25202346/how-to-obtain-leaf-count-express
def tree(expr):
    if expr.operator() is None:
        return expr
    else:
        return [expr.operator()+map(tree, expr.operands())

try:
    # 1.35 is a fudge factor since this estimate of leaf count is bit lower than
    #what it should be compared to Mathematica's
    leafCount = round(1.35*len(flatten(tree(anti))))
except Exception as ee:
    leafCount =1
```

For Sympy, called directly from Python, the following code is used

```
try:
    # 1.7 is a fudge factor since it is low side from actual leaf count
    leafCount = round(1.7*count_ops(anti))

    except Exception as ee:
        leafCount =1
```

When these cas systems implement a builtin function to find the leaf size of expressions, it will be used instead, and these tests run again.

## 1.6 Grading of results

The table below summarizes the grading of each CAS system.

Important note: A number of problems in this test suite have no antiderivative in closed form. This means the antiderivative of these integrals can not be expressed in terms of elementary, special functions or Hypergeometric2F1 functions. `RootSum` and `RootOf` are not allowed.

If a CAS returns the above integral unevaluated within the time limit, then the result is counted as passed and assigned an A grade.

However, if CAS times out, then it is assigned an F grade even if the integral is not integrable, as this implies CAS could not determine that the integral is not integrable in the time limit.

If a CAS returns an antiderivative to such an integral, it is assigned an A grade automatically and this special result is listed in the introduction section of each individual test report to make it easy to identify as this can be important result to investigate.

The results given in in the table below reflects the above.

System	solved	Failed
Rubi	% 42.86 ( 3 )	% 57.14 ( 4 )
Rubi in Sympy	% 28.57 ( 2 )	% 71.43 ( 5 )
Mathematica	% 85.71 ( 6 )	% 14.29 ( 1 )
Maple	% 71.43 ( 5 )	% 28.57 ( 2 )
Maxima	% 42.86 ( 3 )	% 57.14 ( 4 )
Fricas	% 100. ( 7 )	% 0. ( 0 )
Sympy	% 57.14 ( 4 )	% 42.86 ( 3 )
Giac	% 42.86 ( 3 )	% 57.14 ( 4 )

The table below gives additional break down of the grading of quality of the antiderivatives generated by each CAS. The grading is given using the letters A,B,C and F with A being the best quality. The grading is accomplished by comparing the antiderivative generated with the optimal antiderivatives included in the test suite. The following table describes the meaning of these grades.

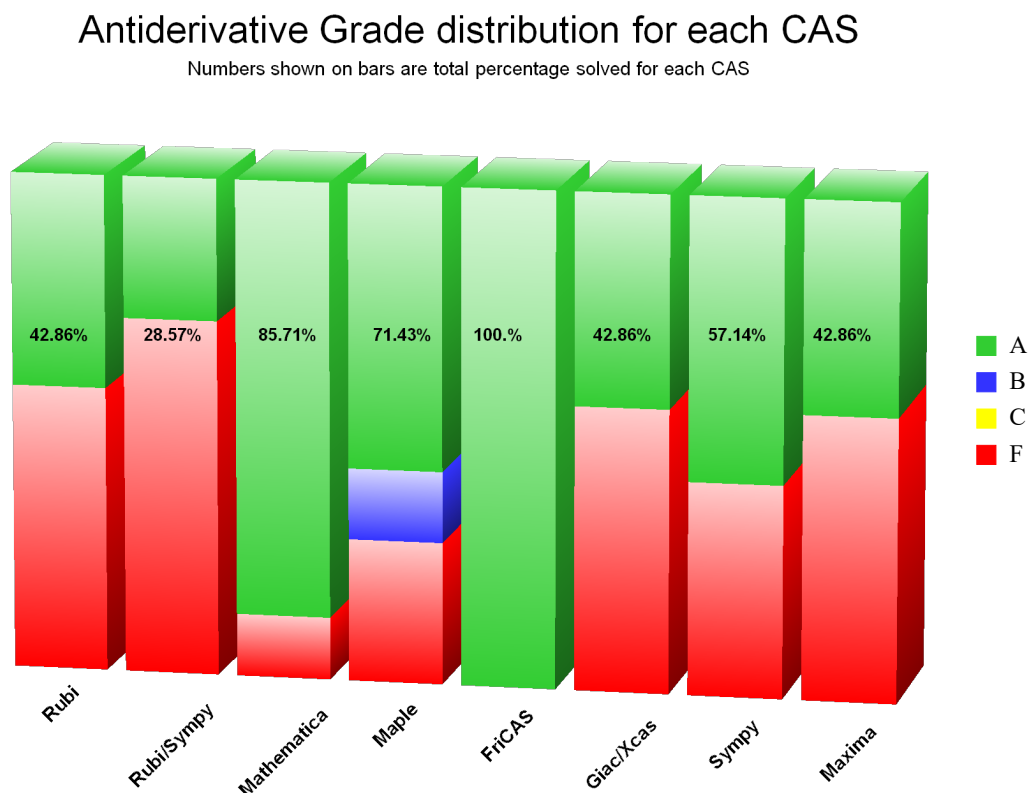
grade	description
A	Integral was solved and antiderivative is optimal in quality and leaf size.
B	Integral was solved and antiderivative is optimal in quality but leaf size is larger than twice the optimal antiderivatives leaf size.
C	Integral was solved and antiderivative is non-optimal in quality. This can be due to one or more of the following reasons <ul style="list-style-type: none"> <li>1. antiderivative contains a hypergeometric function and the optimal antiderivative does not.</li> <li>2. antiderivative contains a special function and the optimal antiderivative does not.</li> <li>3. antiderivative contains the imaginary unit and the optimal antiderivative does not.</li> </ul>
F	Integral was not solved. Either the integral was returned unevaluated within the time limit, or it timed out, or CAS hanged or crashed or an exception was raised.

Grading is currently implemented only for for Mathematica, Rubi and Maple results. For all other CAS systems (Maxima, Fricas, Sympy, Giac, Rubi in sympy), the grading function is not yet implemented. For these systems, a grade of A is assigned if the integrate command completes successfully and a grade of F otherwise.

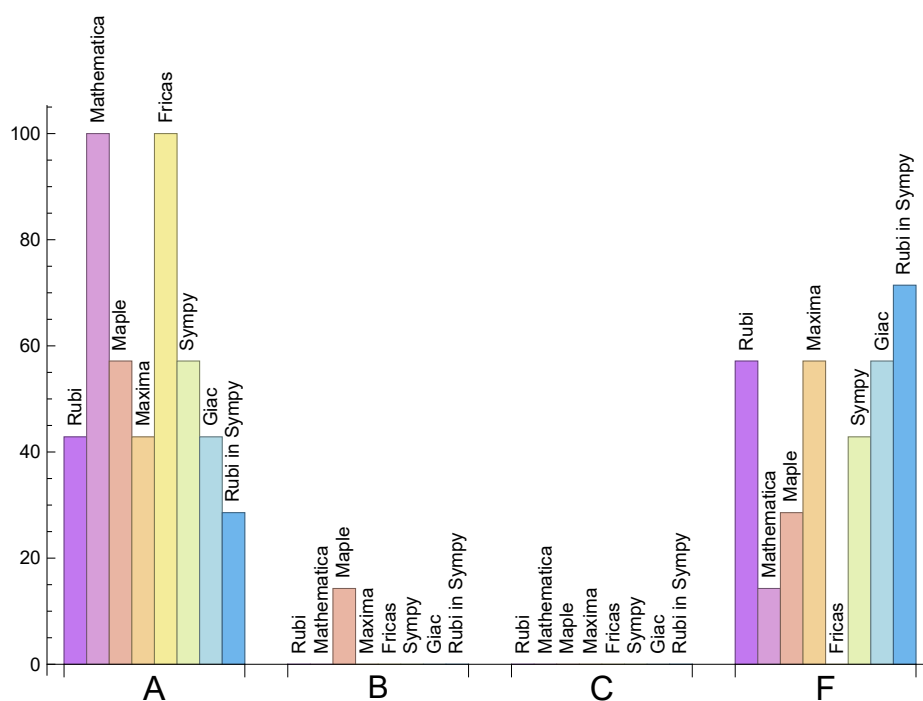
Based on the above, the following table summarizes the grading for this test suite.

System	% A grade	% B grade	% C grade	% F grade
Rubi	42.86	0.	0.	57.14
Rubi in Sympy	28.57	0.	0.	71.43
Mathematica	100.	0.	0.	14.29
Maple	57.14	14.29	0.	28.57
Maxima	42.86	0.	0.	57.14
Fricas	100.	0.	0.	0.
Sympy	57.14	0.	0.	42.86
Giac	42.86	0.	0.	57.14

The following is a Bar chart illustration of the data in the above table.



The figure below compares the CAS systems for each grade level.



## 1.7 Performance

The table below summarizes the performance of each CAS system in terms of CPU time and leaf size of results.

System	Mean time (sec)	Mean size	Normalized mean	Median size	Normalized median
Rubi	0.16	22.33	1.	10.	1.
Rubi in Sympy	5.59	29.5	0.9	29.5	0.9
Mathematica	0.02	13.5	0.94	10.	1.
Maple	0.01	15.6	1.16	13.	1.
Maxima	1.49	52.67	1.68	12.	1.2
Fricas	0.21	21.43	1.34	16.	1.29
Sympy	0.69	14.25	0.74	9.	0.78
Giac	0.2	30.	1.71	41.	1.17

## 1.8 list of integrals that has no closed form antiderivative

{}

## 1.9 list of integrals not solved by each system

Not solved by Rubi {2, 3, 5, 7}

Not solved by Rubi in Sympy {2, 3, 4, 5, 7}

Not solved by Mathematica {3}

Not solved by Maple {2, 3}

Not solved by Maxima {2, 3, 4, 5}

Not solved by Fricas {}

Not solved by Sympy {2, 3, 4}

Not solved by Giac {2, 3, 6, 7}

## 1.10 list of integrals solved by CAS but has no known antiderivative

Rubi {}

Rubi in Sympy {}

Mathematica {}

Maple {}

Maxima {}

Fricas {}

Sympy {}

Giac {}

## 1.11 list of integrals solved by CAS but failed verification

The following are integrals solved by CAS but the verification phase failed to verify the anti-derivative produced is correct. This does not mean necessarily that the anti-derivative is wrong, as additional methods of verification might be needed, or more time is needed (3 minutes time limit was used). These integrals are listed here to make it easier to do further investigation to determine why it was not possible to verify the result produced.

Rubi {}

**Mathematica** {}

**Maple** Verification phase not implemented yet.

**Maxima** Verification phase not implemented yet.

**Fricas** Verification phase not implemented yet.

**Sympy** Verification phase not implemented yet.

**Giac** Verification phase not implemented yet.

**Rubi in Sympy** Verification phase not implemented yet.



## 2 detailed summary tables of results

### 2.1 Detailed conclusion table per each integral for all CAS systems

Detailed conclusion table per each integral is given by table below. The elapsed time is in seconds. For failed result it is given as F(-1) if the failure was due to timeout. It is given as F(-2) if the failure was due to an exception being raised, which could indicate a bug in the system. If the failure was due to integral not being evaluated within the time limit, then it is given just an F.

In this table, the column **normalized size** is defined as  $\frac{\text{antiderivative leaf size}}{\text{optimal antiderivative leaf size}}$

Problem 1	Optimal	Rubi	Mathematica	Maple	Maxima	Fricas	Sympy	Giac	Rubi in Sympy
grade	A	A	A	A	A	A	A	A	A
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	51	51	32	32	134	42	31	42	51
normalized size	1	1.	0.63	0.63	2.63	0.82	0.61	0.82	1.
time (sec)	N/A	0.269	0.01	0.004	1.374	0.212	0.095	0.206	8.505

Problem 2	Optimal	Rubi	Mathematica	Maple	Maxima	Fricas	Sympy	Giac	Rubi in Sympy
grade	A	F	A	F	F	A	F	F	F(-1)
verified	N/A	N/A	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	10	0	10	0	0	14	0	0	0
normalized size	1	0.	1.	0.	0.	1.4	0.	0.	0.
time (sec)	N/A	0.647	0.029	0.089	0.	0.217	0.	0.	0.

Problem 3	Optimal	Rubi	Mathematica	Maple	Maxima	Fricas	Sympy	Giac	Rubi in Sympy
grade	A	F	A	F	F	A	F	F	F(-1)
verified	N/A	N/A	N/A	TBD	TBD	TBD	TBD	TBD	TBD
size	28	0	0	0	0	36	0	0	0
normalized size	1	0.	0.	0.	0.	1.29	0.	0.	0.
time (sec)	N/A	0.786	0.504	0.064	0.	0.21	0.	0.	0.

Problem 4	Optimal	Rubi	Mathematica	Maple	Maxima	Fricas	Sympy	Giac	Rubi in Sympy
grade	A	A	A	B	F	A	F	A	F
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	6	6	6	13	0	7	0	7	0
normalized size	1	1.	1.	2.17	0.	1.17	0.	1.17	0.
time (sec)	N/A	0.17	0.006	0.01	0.	0.203	0.	0.2	0.

Problem 5	Optimal	Rubi	Mathematica	Maple	Maxima	Fricas	Sympy	Giac	Rubi in Sympy
grade	A	F	A	A	F	A	A	A	F(-1)
verified	N/A	N/A	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	13	0	13	13	0	16	10	41	0
normalized size	1	0.	1.	1.	0.	1.23	0.77	3.15	0.
time (sec)	N/A	0.618	0.029	0.007	0.	0.198	0.229	0.204	0.

Problem 6	Optimal	Rubi	Mathematica	Maple	Maxima	Fricas	Sympy	Giac	Rubi in Sympy
grade	A	A	A	A	A	A	A	F	A
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	10	10	10	10	12	16	8	0	8
normalized size	1	1.	1.	1.	1.2	1.6	0.8	0.	0.8
time (sec)	N/A	0.042	0.012	0.016	1.522	0.204	1.14	0.	2.674

Problem 7	Optimal	Rubi	Mathematica	Maple	Maxima	Fricas	Sympy	Giac	Rubi in Sympy
grade	A	F	A	A	A	A	A	F	F
verified	N/A	N/A	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	10	0	10	10	12	19	8	0	0
normalized size	1	0.	1.	1.	1.2	1.9	0.8	0.	0.
time (sec)	N/A	0.303	0.021	0.022	1.571	0.209	1.309	0.	0.

## 2.2 Detailed conclusion table specific for Rubi results

The following table is specific to Rubi. It gives additional statistics for each integral. the column **steps** is the number of steps used by Rubi to obtain the antiderivative. The **rules** column is the number of unique rules used. The **integrand size** column is the leaf size of the integrand. Finally the ratio  $\frac{\text{number of rules}}{\text{integrand size}}$  is given. The larger this ratio is, the harder the integral was to solve. In this test, problem number [1] had the largest ratio of [ 0.1364 ]

Table 1: Rubi specific breakdown of results for each integral

#	grade	number of steps used	number of unique rules	normalized antiderivative leaf size	integrand leaf size	$\frac{\text{number of rules}}{\text{integrand leaf size}}$
1	A	25	3	1.	22	0.136
2	F	0	0	N/A	0	N/A
3	F	0	0	N/A	0	N/A
4	A	2	2	1.	20	0.1
5	F	0	0	N/A	0	N/A
6	A	1	1	1.	19	0.053
7	F	0	0	N/A	0	N/A

### 3 Listing of integrals

#### 3.1 $\int e^x (1 - x^3 + x^4 - x^5 + x^6) dx$

**Optimal.** Leaf size=51

$$e^x x^6 - 7e^x x^5 + 36e^x x^4 - 145e^x x^3 + 435e^x x^2 - 870e^x x + 871e^x$$

[Out]  $871 * E^x - 870 * E^x * x + 435 * E^x * x^2 - 145 * E^x * x^3 + 36 * E^x * x^4 - 7 * E^x * x^5 + E^x * x^6$

**Rubi [A]** time = 0.268847, antiderivative size = 51, normalized size of antiderivative = 1., number of steps used = 25, number of rules used = 3, integrand size = 22,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.136$

$$e^x x^6 - 7e^x x^5 + 36e^x x^4 - 145e^x x^3 + 435e^x x^2 - 870e^x x + 871e^x$$

Antiderivative was successfully verified.

[In] Int[E^x\*(1 - x^3 + x^4 - x^5 + x^6), x]

[Out]  $871 * E^x - 870 * E^x * x + 435 * E^x * x^2 - 145 * E^x * x^3 + 36 * E^x * x^4 - 7 * E^x * x^5 + E^x * x^6$

**Rubi in Sympy [A]** time = 8.50467, size = 51, normalized size = 1.

$$x^6 e^x - 7x^5 e^x + 36x^4 e^x - 145x^3 e^x + 435x^2 e^x - 870x e^x + 871e^x$$

Verification of antiderivative is not currently implemented for this CAS.

[In] rubi\_integrate((x\*\*6-x\*\*5+x\*\*4-x\*\*3+1)\*exp(x), x)

[Out]  $x**6*exp(x) - 7*x**5*exp(x) + 36*x**4*exp(x) - 145*x**3*exp(x) + 435*x**2*exp(x) - 870*x*exp(x) + 871*exp(x)$

**Mathematica [A]** time = 0.00955885, size = 32, normalized size = 0.63

$$e^x (x^6 - 7x^5 + 36x^4 - 145x^3 + 435x^2 - 870x + 871)$$

Antiderivative was successfully verified.

[In] Integrate[E^x\*(1 - x^3 + x^4 - x^5 + x^6), x]

[Out]  $E^x*(871 - 870*x + 435*x^2 - 145*x^3 + 36*x^4 - 7*x^5 + x^6)$

**Maple [A]** time = 0.004, size = 32, normalized size = 0.6

$$(x^6 - 7x^5 + 36x^4 - 145x^3 + 435x^2 - 870x + 871) e^x$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((x^6-x^5+x^4-x^3+1)\*exp(x), x)

[Out]  $(x^6 - 7x^5 + 36x^4 - 145x^3 + 435x^2 - 870x + 871) \cdot \exp(x)$

**Maxima [A]** time = 1.37429, size = 134, normalized size = 2.63

$$(x^6 - 6x^5 + 30x^4 - 120x^3 + 360x^2 - 720x + 720)e^x - (x^5 - 5x^4 + 20x^3 - 60x^2 + 120x - 120)e^x + (x^4 - 4x^3 + 12x^2 - 24x + 24)e^x - (x^3 - 3x^2 + 6x - 6)e^x + e^x$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((x^6 - x^5 + x^4 - x^3 + 1)*e^x,x, algorithm="maxima")`

[Out]  $(x^6 - 6x^5 + 30x^4 - 120x^3 + 360x^2 - 720x + 720) \cdot e^x - (x^5 - 5x^4 + 20x^3 - 60x^2 + 120x - 120) \cdot e^x + (x^4 - 4x^3 + 12x^2 - 24x + 24) \cdot e^x - (x^3 - 3x^2 + 6x - 6) \cdot e^x + e^x$

**Fricas [A]** time = 0.211803, size = 42, normalized size = 0.82

$$(x^6 - 7x^5 + 36x^4 - 145x^3 + 435x^2 - 870x + 871)e^x$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((x^6 - x^5 + x^4 - x^3 + 1)*e^x,x, algorithm="fricas")`

[Out]  $(x^6 - 7x^5 + 36x^4 - 145x^3 + 435x^2 - 870x + 871) \cdot e^x$

**Sympy [A]** time = 0.095039, size = 31, normalized size = 0.61

$$(x^6 - 7x^5 + 36x^4 - 145x^3 + 435x^2 - 870x + 871) e^x$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((x**6-x**5+x**4-x**3+1)*exp(x),x)`

[Out]  $(x^{**6} - 7x^{**5} + 36x^{**4} - 145x^{**3} + 435x^{**2} - 870x + 871) \cdot \exp(x)$

**GIAC/XCAS [A]** time = 0.206139, size = 42, normalized size = 0.82

$$(x^6 - 7x^5 + 36x^4 - 145x^3 + 435x^2 - 870x + 871)e^x$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((x^6 - x^5 + x^4 - x^3 + 1)*e^x,x, algorithm="giac")`

[Out]  $(x^6 - 7x^5 + 36x^4 - 145x^3 + 435x^2 - 870x + 871) \cdot e^x$

$$3.2 \quad \int \frac{e^{\frac{x}{2+x^2}} (2-x^2)}{2x+x^3} dx$$

**Optimal.** Leaf size=10

$$\text{ExpIntegralEi}\left(\frac{x}{x^2+2}\right)$$

[Out] ExpIntegralEi[x/(2 + x^2)]

**Rubi [F]** time = 0.646906, antiderivative size = 0, normalized size of antiderivative = 0., number of rules used = 0, number of rules used = 0, integrand size = 0,  $\frac{\text{number of rules}}{\text{integrand size}} = 0$ .

$$\text{Int}\left(\frac{e^{\frac{x}{2+x^2}} (2-x^2)}{2x+x^3}, x\right)$$

Verification is Not applicable to the result.

[In] Int[(E^(x/(2 + x^2)))\*(2 - x^2))/(2\*x + x^3), x]

[Out] Defer[Int][E^(x/(2 + x^2))/(I\*Sqrt[2] - x), x] + Defer[Int][E^(x/(2 + x^2))/x, x] - Defer[Int][E^(x/(2 + x^2))/(I\*Sqrt[2] + x), x]

**Rubi in Sympy [F(-1)]** time = 0., size = 0, normalized size = 0.

Timed out

Verification of antiderivative is not currently implemented for this CAS.

[In] rubi\_integrate((-x\*\*2+2)\*exp(x/(x\*\*2+2))/(x\*\*3+2\*x), x)

[Out] Timed out

**Mathematica [A]** time = 0.0291441, size = 10, normalized size = 1.

$$\text{ExpIntegralEi}\left(\frac{x}{x^2+2}\right)$$

Antiderivative was successfully verified.

[In] Integrate[(E^(x/(2 + x^2)))\*(2 - x^2))/(2\*x + x^3), x]

[Out] ExpIntegralEi[x/(2 + x^2)]

**Maple [F]** time = 0.089, size = 0, normalized size = 0.

$$\int \frac{-x^2+2}{x^3+2x} e^{\frac{x}{x^2+2}} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((-x^2+2)\*exp(x/(x^2+2))/(x^3+2\*x), x)

[Out] int((-x^2+2)\*exp(x/(x^2+2))/(x^3+2\*x), x)

---

**Maxima [F]** time = 0., size = 0, normalized size = 0.

$$-\int \frac{(x^2 - 2)e^{\left(\frac{x}{x^2+2}\right)}}{x^3 + 2x} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate(-(x^2 - 2)\*e^(x/(x^2 + 2))/(x^3 + 2\*x), x, algorithm="maxima")

[Out] -integrate((x^2 - 2)\*e^(x/(x^2 + 2))/(x^3 + 2\*x), x)

---

**Fricas [A]** time = 0.216999, size = 14, normalized size = 1.4

$$\text{Ei}\left(\frac{x}{x^2 + 2}\right)$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate(-(x^2 - 2)\*e^(x/(x^2 + 2))/(x^3 + 2\*x), x, algorithm="fricas")

[Out] Ei(x/(x^2 + 2))

---

**Sympy [F]** time = 0., size = 0, normalized size = 0.

$$-\int \left( -\frac{2e^{\frac{x}{x^2+2}}}{x^3 + 2x} \right) dx - \int \frac{x^2 e^{\frac{x}{x^2+2}}}{x^3 + 2x} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((-x\*\*2+2)\*exp(x/(x\*\*2+2))/(x\*\*3+2\*x), x)

[Out] -Integral(-2\*exp(x/(x\*\*2 + 2))/(x\*\*3 + 2\*x), x) - Integral(x\*\*2\*exp(x/(x\*\*2 + 2))/(x\*\*3 + 2\*x), x)

---

**GIAC/XCAS [F]** time = 0., size = 0, normalized size = 0.

$$\int -\frac{(x^2 - 2)e^{\left(\frac{x}{x^2+2}\right)}}{x^3 + 2x} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate(-(x^2 - 2)\*e^(x/(x^2 + 2))/(x^3 + 2\*x), x, algorithm="giac")

[Out] integrate(-(x^2 - 2)\*e^(x/(x^2 + 2))/(x^3 + 2\*x), x)

$$3.3 \quad \int \frac{e^{\frac{x}{2+x^2}} (2+2x+3x^2-x^3+2x^4)}{2x+x^3} dx$$

**Optimal.** Leaf size=28

$$\text{ExpIntegralEi}\left(\frac{x}{x^2+2}\right) + e^{\frac{x}{x^2+2}} (x^2+2)$$

[Out]  $E^{(x/(2+x^2))} * (2+x^2) + \text{ExpIntegralEi}[x/(2+x^2)]$

**Rubi [F]** time = 0.786464, antiderivative size = 0, normalized size of antiderivative = 0., number of steps used = 0, number of rules used = 0, integrand size = 0,  $\frac{\text{number of rules}}{\text{integrand size}} = 0$ .

$$\text{Int}\left(\frac{e^{\frac{x}{2+x^2}} (2+2x+3x^2-x^3+2x^4)}{2x+x^3}, x\right)$$

Verification is Not applicable to the result.

[In]  $\text{Int}[(E^{(x/(2+x^2))}) * (2+2*x+3*x^2-x^3+2*x^4)] / (2*x+x^3), x]$

[Out]  $-\text{Defer}[\text{Int}[E^{(x/(2+x^2))}, x] + (1+I*\text{Sqrt}[2]) * \text{Defer}[\text{Int}[E^{(x/(2+x^2))} / (I*\text{Sqrt}[2]-x), x] + \text{Defer}[\text{Int}[E^{(x/(2+x^2))} / x, x] + 2 * \text{Defer}[\text{Int}[E^{(x/(2+x^2))} * x, x] - (1-I*\text{Sqrt}[2]) * \text{Defer}[\text{Int}[E^{(x/(2+x^2))} / (I*\text{Sqrt}[2]+x), x]$

**Rubi in Sympy [F-1)]** time = 0., size = 0, normalized size = 0.

Timed out

Verification of antiderivative is not currently implemented for this CAS.

[In]  $\text{rubi\_integrate}((2*x**4-x**3+3*x**2+2*x+2) * \exp(x/(x**2+2)) / (x**3+2*x), x)$

[Out] Timed out

**Mathematica [A]** time = 0.503685, size = 0, normalized size = 0.

$$\int \frac{e^{\frac{x}{2+x^2}} (2+2x+3x^2-x^3+2x^4)}{2x+x^3} dx$$

Verification is Not applicable to the result.

[In]  $\text{Integrate}[(E^{(x/(2+x^2))}) * (2+2*x+3*x^2-x^3+2*x^4)] / (2*x+x^3), x]$

[Out]  $\text{Integrate}[(E^{(x/(2+x^2))}) * (2+2*x+3*x^2-x^3+2*x^4)] / (2*x+x^3), x]$

**Maple [F]** time = 0.064, size = 0, normalized size = 0.

$$\int \frac{2x^4-x^3+3x^2+2x+2}{x^3+2x} e^{\frac{x}{x^2+2}} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `int((2*x^4-x^3+3*x^2+2*x+2)*exp(x/(x^2+2))/(x^3+2*x),x)`

[Out] `int((2*x^4-x^3+3*x^2+2*x+2)*exp(x/(x^2+2))/(x^3+2*x),x)`

**Maxima [F]** time = 0., size = 0, normalized size = 0.

$$\int \frac{(2x^4 - x^3 + 3x^2 + 2x + 2)e^{\left(\frac{x}{x^2+2}\right)}}{x^3 + 2x} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((2*x^4 - x^3 + 3*x^2 + 2*x + 2)*e^(x/(x^2 + 2))/(x^3 + 2*x),x, algorithm`

[Out] `integrate((2*x^4 - x^3 + 3*x^2 + 2*x + 2)*e^(x/(x^2 + 2))/(x^3 + 2*x), x)`

**Fricas [A]** time = 0.21013, size = 36, normalized size = 1.29

$$(x^2 + 2)e^{\left(\frac{x}{x^2+2}\right)} + \text{Ei}\left(\frac{x}{x^2 + 2}\right)$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((2*x^4 - x^3 + 3*x^2 + 2*x + 2)*e^(x/(x^2 + 2))/(x^3 + 2*x),x, algorithm`

[Out] `(x^2 + 2)*e^(x/(x^2 + 2)) + Ei(x/(x^2 + 2))`

**Sympy [F]** time = 0., size = 0, normalized size = 0.

$$\int \frac{(2x^4 - x^3 + 3x^2 + 2x + 2)e^{\frac{x}{x^2+2}}}{x(x^2 + 2)} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((2*x**4-x**3+3*x**2+2*x+2)*exp(x/(x**2+2))/(x**3+2*x),x)`

[Out] `Integral((2*x**4 - x**3 + 3*x**2 + 2*x + 2)*exp(x/(x**2 + 2))/(x*(x**2 + 2)), x)`

**GIAC/XCAS [F]** time = 0., size = 0, normalized size = 0.

$$\int \frac{(2x^4 - x^3 + 3x^2 + 2x + 2)e^{\left(\frac{x}{x^2+2}\right)}}{x^3 + 2x} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((2*x^4 - x^3 + 3*x^2 + 2*x + 2)*e^(x/(x^2 + 2))/(x^3 + 2*x),x, algorithm`

[Out] `integrate((2*x^4 - x^3 + 3*x^2 + 2*x + 2)*e^(x/(x^2 + 2))/(x^3 + 2*x), x)`



$$3.4 \quad \int \frac{e^{e^x+x}(1+e^x)}{e^x+x} dx$$

**Optimal.** Leaf size=6

$$\text{ExpIntegralEi}(x + e^x)$$

[Out] ExpIntegralEi[E^x + x]

**Rubi [A]** time = 0.170419, antiderivative size = 6, normalized size of antiderivative = 1., number of steps used = 2, number of rules used = 2, integrand size = 20,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.1$

$$\text{ExpIntegralEi}(x + e^x)$$

Antiderivative was successfully verified.

[In] Int[(E^(E^x + x) \* (1 + E^x)) / (E^x + x), x]

[Out] ExpIntegralEi[E^x + x]

**Rubi in Sympy [F]** time = 0., size = 0, normalized size = 0.

$$\int \frac{(e^x + 1)e^{x+e^x}}{x + e^x} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] rubi\_integrate((1+exp(x))\*exp(x+exp(x))/(x+exp(x)), x)

[Out] Integral((exp(x) + 1)\*exp(x + exp(x))/(x + exp(x)), x)

**Mathematica [A]** time = 0.00638302, size = 6, normalized size = 1.

$$\text{ExpIntegralEi}(x + e^x)$$

Antiderivative was successfully verified.

[In] Integrate[(E^(E^x + x) \* (1 + E^x)) / (E^x + x), x]

[Out] ExpIntegralEi[E^x + x]

**Maple [B]** time = 0.01, size = 13, normalized size = 2.2

$$-Ei(1, -e^x - x)$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((1+exp(x))\*exp(exp(x)+x)/(exp(x)+x), x)

[Out] -Ei(1, -exp(x)-x)

**Maxima [F]** time = 0., size = 0, normalized size = 0.

$$\frac{(e^x + 1)e^{(e^x)}}{x + e^x} - \int \frac{((x - 2)e^x - 1)e^{(e^x)}}{x^2 + 2xe^x + e^{(2x)}} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((e^x + 1)\*e^(x + e^x)/(x + e^x), x, algorithm="maxima")

[Out] (e^x + 1)\*e^(e^x)/(x + e^x) - integrate(((x - 2)\*e^x - 1)\*e^(e^x)/(x^2 + 2\*x\*e^x + e^(2\*x)), x)

**Fricas [A]** time = 0.202567, size = 7, normalized size = 1.17

$$\text{Ei}(x + e^x)$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((e^x + 1)\*e^(x + e^x)/(x + e^x), x, algorithm="fricas")

[Out] Ei(x + e^x)

**Sympy [F]** time = 0., size = 0, normalized size = 0.

$$\int \frac{(e^x + 1)e^x e^{e^x}}{x + e^x} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((1+exp(x))\*exp(x+exp(x))/(x+exp(x)), x)

[Out] Integral((exp(x) + 1)\*exp(x)\*exp(exp(x))/(x + exp(x)), x)

**GIAC/XCAS [A]** time = 0.200332, size = 7, normalized size = 1.17

$$\text{Ei}(x + e^x)$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((e^x + 1)\*e^(x + e^x)/(x + e^x), x, algorithm="giac")

[Out] Ei(x + e^x)

$$3.5 \quad \int \frac{e^{-1+x^2} (1-3x-x^2+x^3)}{1-x-x^2+x^3} dx$$

**Optimal.** Leaf size=13

$$e^{\frac{1}{x^2-1}}(x+1)$$

[Out]  $E^{(-1 + x^2)^{-1}}(1 + x)$

**Rubi [F]** time = 0.617887, antiderivative size = 0, normalized size of antiderivative = 0., number of steps used = 0, number of rules used = 0, integrand size = 0,  $\frac{\text{number of rules}}{\text{integrand size}} = 0$ .

$$\text{Int} \left( \frac{e^{-1+x^2} (1-3x-x^2+x^3)}{1-x-x^2+x^3}, x \right)$$

Verification is Not applicable to the result.

[In]  $\text{Int}[(E^{(-1 + x^2)^{-1}}(1 - 3x - x^2 + x^3))/(1 - x - x^2 + x^3), x]$

[Out]  $\text{Defer}[\text{Int}][E^{(-1 + x^2)^{-1}}, x] + \text{Defer}[\text{Int}][E^{(-1 + x^2)^{-1}}/(1 - x), x]/2 - \text{Defer}[\text{Int}][E^{(-1 + x^2)^{-1}}/(-1 + x)^2, x] + \text{Defer}[\text{Int}][E^{(-1 + x^2)^{-1}}/(1 + x), x]/2$

**Rubi in Sympy [F(-1)]** time = 0., size = 0, normalized size = 0.

Timed out

Verification of antiderivative is not currently implemented for this CAS.

[In]  $\text{rubi\_integrate}((x^3-x^2-3x+1)*\exp(1/(x^2-1))/(x^3-x^2-x+1), x)$

[Out] Timed out

**Mathematica [A]** time = 0.0293725, size = 13, normalized size = 1.

$$e^{\frac{1}{x^2-1}}(x+1)$$

Antiderivative was successfully verified.

[In]  $\text{Integrate}[(E^{(-1 + x^2)^{-1}}(1 - 3x - x^2 + x^3))/(1 - x - x^2 + x^3), x]$

[Out]  $E^{(-1 + x^2)^{-1}}(1 + x)$

**Maple [A]** time = 0.007, size = 13, normalized size = 1.

$$e^{(x^2-1)^{-1}}(1+x)$$

Verification of antiderivative is not currently implemented for this CAS.

[In]  $\text{int}((x^3-x^2-3x+1)*\exp(1/(x^2-1))/(x^3-x^2-x+1), x)$

[Out]  $\exp(1/(x^2-1))*(1+x)$

---

**Maxima [F]** time = 0., size = 0, normalized size = 0.

$$\int \frac{(x^3 - x^2 - 3x + 1)e^{\left(\frac{1}{x^2-1}\right)}}{x^3 - x^2 - x + 1} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((x^3 - x^2 - 3\*x + 1)\*e^(1/(x^2 - 1))/(x^3 - x^2 - x + 1), x, algorithm="m

[Out] integrate((x^3 - x^2 - 3\*x + 1)\*e^(1/(x^2 - 1))/(x^3 - x^2 - x + 1), x)

---

**Fricas [A]** time = 0.197782, size = 16, normalized size = 1.23

$$(x + 1)e^{\left(\frac{1}{x^2-1}\right)}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((x^3 - x^2 - 3\*x + 1)\*e^(1/(x^2 - 1))/(x^3 - x^2 - x + 1), x, algorithm="f

[Out] (x + 1)\*e^(1/(x^2 - 1))

---

**Sympy [A]** time = 0.229493, size = 10, normalized size = 0.77

$$(x + 1)e^{\frac{1}{x^2-1}}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((x\*\*3-x\*\*2-3\*x+1)\*exp(1/(x\*\*2-1))/(x\*\*3-x\*\*2-x+1), x)

[Out] (x + 1)\*exp(1/(x\*\*2 - 1))

---

**GIAC/XCAS [A]** time = 0.203606, size = 41, normalized size = 3.15

$$\left(xe^{\left(\frac{x^2}{x^2-1}\right)} + e^{\left(\frac{x^2}{x^2-1}\right)}\right)e^{(-1)}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((x^3 - x^2 - 3\*x + 1)\*e^(1/(x^2 - 1))/(x^3 - x^2 - x + 1), x, algorithm="g

[Out] (x\*e^(x^2/(x^2 - 1)) + e^(x^2/(x^2 - 1)))\*e^(-1)

$$3.6 \quad \int \frac{e^{1+\frac{1}{\log(x)}}(-1+\log^2(x))}{\log^2(x)} dx$$

**Optimal.** Leaf size=10

$$xe^{\frac{1}{\log(x)}+1}$$

[Out]  $E^{(1 + \text{Log}[x]^{(-1)})} * x$

**Rubi [A]** time = 0.0423875, antiderivative size = 10, normalized size of antiderivative = 1., number of steps used = 1, number of rules used = 1, integrand size = 19,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.053$

$$xe^{\frac{1}{\log(x)}+1}$$

Antiderivative was successfully verified.

[In]  $\text{Int}[(E^{(1 + \text{Log}[x]^{(-1)})} * (-1 + \text{Log}[x]^2))/\text{Log}[x]^2, x]$

[Out]  $E^{(1 + \text{Log}[x]^{(-1)})} * x$

**Rubi in Sympy [A]** time = 2.67433, size = 8, normalized size = 0.8

$$xe^{1+\frac{1}{\log(x)}}$$

Verification of antiderivative is not currently implemented for this CAS.

[In]  $\text{rubi\_integrate}(\exp(1+1/\ln(x)) * (-1+\ln(x)**2)/\ln(x)**2, x)$

[Out]  $x * \exp(1 + 1/\log(x))$

**Mathematica [A]** time = 0.0119706, size = 10, normalized size = 1.

$$xe^{\frac{1}{\log(x)}+1}$$

Antiderivative was successfully verified.

[In]  $\text{Integrate}[(E^{(1 + \text{Log}[x]^{(-1)})} * (-1 + \text{Log}[x]^2))/\text{Log}[x]^2, x]$

[Out]  $E^{(1 + \text{Log}[x]^{(-1)})} * x$

**Maple [A]** time = 0.016, size = 10, normalized size = 1.

$$e^{1+(\ln(x))^{-1}} x$$

Verification of antiderivative is not currently implemented for this CAS.

[In]  $\text{int}(\exp(1+1/\ln(x)) * (-1+\ln(x)^2)/\ln(x)^2, x)$

[Out]  $\exp(1+1/\ln(x)) * x$

**Maxima [A]** time = 1.52221, size = 12, normalized size = 1.2

$$xe^{\left(\frac{1}{\log(x)}+1\right)}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((log(x)^2 - 1)\*e^(1/log(x) + 1)/log(x)^2,x, algorithm="maxima")

[Out] x\*e^(1/log(x) + 1)

---

**Fricas [A]** time = 0.203544, size = 16, normalized size = 1.6

$$xe^{\left(\frac{\log(x)+1}{\log(x)}\right)}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((log(x)^2 - 1)\*e^(1/log(x) + 1)/log(x)^2,x, algorithm="fricas")

[Out] x\*e^((log(x) + 1)/log(x))

---

**Sympy [A]** time = 1.14028, size = 8, normalized size = 0.8

$$xe^{1+\frac{1}{\log(x)}}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate(exp(1+1/ln(x))\*(-1+ln(x)\*\*2)/ln(x)\*\*2,x)

[Out] x\*exp(1 + 1/log(x))

---

**GIAC/XCAS [F]** time = 0., size = 0, normalized size = 0.

*undef*

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((log(x)^2 - 1)\*e^(1/log(x) + 1)/log(x)^2,x, algorithm="giac")

[Out] undef

$$3.7 \quad \int \frac{e^{x+\frac{1}{\log(x)}}(-1+(1+x)\log^2(x))}{\log^2(x)} dx$$

**Optimal.** Leaf size=10

$$xe^{x+\frac{1}{\log(x)}}$$

[Out]  $E^{(x + \text{Log}[x]^{-1})} * x$

**Rubi [F]** time = 0.303053, antiderivative size = 0, normalized size of antiderivative = 0., number of steps used = 0, number of rules used = 0, integrand size = 0,  $\frac{\text{number of rules}}{\text{integrand size}} = 0$ .

$$\text{Int}\left(\frac{e^{x+\frac{1}{\log(x)}}(-1+(1+x)\log^2(x))}{\log^2(x)}, x\right)$$

Verification is Not applicable to the result.

[In]  $\text{Int}[(E^{(x + \text{Log}[x]^{-1})}) * (-1 + (1 + x) * \text{Log}[x]^2)] / \text{Log}[x]^2, x]$

[Out]  $\text{Defer}[\text{Int}][E^{(x + \text{Log}[x]^{-1})}, x] + \text{Defer}[\text{Int}][E^{(x + \text{Log}[x]^{-1})} * x, x] - \text{Defer}[\text{Int}][E^{(x + \text{Log}[x]^{-1})} / \text{Log}[x]^2, x]$

**Rubi in Sympy [F]** time = 0., size = 0, normalized size = 0.

$$\int \frac{((x+1)\log(x)^2 - 1)e^{x+\frac{1}{\log(x)}}}{\log(x)^2} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In]  $\text{rubi\_integrate}(\exp(x+1/\ln(x)) * (-1+(1+x) * \ln(x)**2) / \ln(x)**2, x)$

[Out]  $\text{Integral}(((x + 1) * \log(x)**2 - 1) * \exp(x + 1/\log(x)) / \log(x)**2, x)$

**Mathematica [A]** time = 0.0206338, size = 10, normalized size = 1.

$$xe^{x+\frac{1}{\log(x)}}$$

Antiderivative was successfully verified.

[In]  $\text{Integrate}[(E^{(x + \text{Log}[x]^{-1})}) * (-1 + (1 + x) * \text{Log}[x]^2)] / \text{Log}[x]^2, x]$

[Out]  $E^{(x + \text{Log}[x]^{-1})} * x$

**Maple [A]** time = 0.022, size = 10, normalized size = 1.

$$e^{x+(\ln(x))^{-1}} x$$

Verification of antiderivative is not currently implemented for this CAS.

[In]  $\text{int}(\exp(x+1/\ln(x)) * (-1+(1+x) * \ln(x)^2) / \ln(x)^2, x)$

[Out]  $\exp(x+1/\ln(x)) * x$

**Maxima [A]** time = 1.57102, size = 12, normalized size = 1.2

$$xe^{\left(x + \frac{1}{\log(x)}\right)}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(((x + 1) * log(x)^2 - 1) * e^(x + 1/log(x))/log(x)^2,x, algorithm="maxima")`

[Out]  $x * e^{(x + 1/\log(x))}$

**Fricas [A]** time = 0.208516, size = 19, normalized size = 1.9

$$xe^{\left(\frac{x \log(x)+1}{\log(x)}\right)}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(((x + 1) * log(x)^2 - 1) * e^(x + 1/log(x))/log(x)^2,x, algorithm="fricas")`

[Out]  $x * e^{((x * \log(x) + 1)/\log(x))}$

**Sympy [A]** time = 1.30853, size = 8, normalized size = 0.8

$$xe^{x + \frac{1}{\log(x)}}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(exp(x+1/ln(x)) * (-1+(1+x) * ln(x)**2)/ln(x)**2,x)`

[Out]  $x * \exp(x + 1/\log(x))$

**GIAC/XCAS [F]** time = 0., size = 0, normalized size = 0.

*undef*

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(((x + 1) * log(x)^2 - 1) * e^(x + 1/log(x))/log(x)^2,x, algorithm="giac")`

[Out] undef



## 4 Listing of Grading functions

The following are the current version of the grading functions used for grading the quality of the antiderivative with reference to the optimal antiderivative included in the test suite.

There is a version for Mathematica/Rubi followed by one for Maple. The following are links to the source code.

The following are the listing of the above functions.

```
(* Original version thanks to Albert Rich emailed on 03/21/2017 *)
(* ::Package:: *)
(* ::Subsection:: *)
(*GradeAntiderivative[result, optimal]*)

(* ::Text:: *)
(*If result and optimal are mathematical expressions, *)
(*      GradeAntiderivative[result, optimal] returns*)
(* "F" if the result fails to integrate an expression that*)
(*   is integrable*)
(* "C" if result involves higher level functions than necessary*)
(* "B" if result is more than twice the size of the optimal*)
(*   antiderivative*)
(* "A" if result can be considered optimal*)

GradeAntiderivative[result_, optimal_] :=
  If[ExpnType[result]<=ExpnType[optimal],
    If[FreeQ[result, Complex] || Not[FreeQ[optimal, Complex]],
      If[LeafCount[result]<=2*LeafCount[optimal],
        "A",
        "B"],
      "C"],
    If[FreeQ[result, Integrate] && FreeQ[result, Int],
      "C",
      "F"]]

(* ::Text:: *)
(*The following summarizes the type number assigned an *)
(*expression based on the functions it involves*)
(*1 = rational function*)
(*2 = algebraic function*)
(*3 = elementary function*)
(*4 = special function*)
(*5 = hyperpergeometric function*)
(*6 = appell function*)
(*7 = rootsum function*)
(*8 = integrate function*)
(*9 = unknown function*)
```

```

ExpnType[expn_] :=
  If[AtomQ[expn], 1,
  If[ListQ[expn],
    Max[Map[ExpnType, expn]],
  If[Head[expn]==Power,
    If[IntegerQ[expn[[2]]],
      ExpnType[expn[[1]]],
    If[Head[expn[[2]]]==Rational,
      If[IntegerQ[expn[[1]]] || Head[expn[[1]]]==Rational, 1,
      Max[ExpnType[expn[[1]], 2]],
      Max[ExpnType[expn[[1]], ExpnType[expn[[2]], 3]],
  If[Head[expn]==Plus || Head[expn]==Times,
    Max[ExpnType[First[expn]], ExpnType[Rest[expn]]],
  If[ElementaryFunctionQ[Head[expn]],
    Max[3, ExpnType[expn[[1]]],
  If[SpecialFunctionQ[Head[expn]],
    Apply[Max, Append[Map[ExpnType, Apply[List, expn]], 4]],
  If[HypergeometricFunctionQ[Head[expn]],
    Apply[Max, Append[Map[ExpnType, Apply[List, expn]], 5]],
  If[AppellFunctionQ[Head[expn]],
    Apply[Max, Append[Map[ExpnType, Apply[List, expn]], 6]],
  If[Head[expn]==RootSum,
    Apply[Max, Append[Map[ExpnType, Apply[List, expn]], 7]],
  If[Head[expn]==Integrate || Head[expn]==Int,
    Apply[Max, Append[Map[ExpnType, Apply[List, expn]], 8]],
  9]]]]]]]]]

```

```

ElementaryFunctionQ[func_] :=
  MemberQ[{
    Exp, Log,
    Sin, Cos, Tan, Cot, Sec, Csc,
    ArcSin, ArcCos, ArcTan, ArcCot, ArcSec, ArcCsc,
    Sinh, Cosh, Tanh, Coth, Sech, Csch,
    ArcSinh, ArcCosh, ArcTanh, ArcCoth, ArcSech, ArcCsch
  }, func]

SpecialFunctionQ[func_] :=
  MemberQ[{
    Erf, Erfc, Erfi,
    FresnelS, FresnelC,
    ExpIntegralE, ExpIntegralEi, LogIntegral,
    SinIntegral, CosIntegral, SinhIntegral, CoshIntegral,
    Gamma, LogGamma, PolyGamma,
    Zeta, PolyLog, ProductLog,
    EllipticF, EllipticE, EllipticPi
  }, func]

HypergeometricFunctionQ[func_] := MemberQ[{Hypergeometric1F1, Hypergeometric2F1, HypergeometricPFQ}, func]
AppellFunctionQ[func_] := MemberQ[{AppellF1}, func]

```

```

# File: GradeAntiderivative.mpl Original version thanks to Albert Rich emailed on 03/21/2017
#Nasser 03/22/2017 Use Maple leaf count instead since buildin
#Nasser 03/23/2017 missing 'ln' for ElementaryFunctionQ added
#Nasser 03/24/2017 corrected the check for complex result
#Nasser 10/27/2017 check for leafsize and do not call ExpnType()
#
# if leaf size is "too large". Set at 500,000
GradeAntiderivative := proc(result,optimal)
local leaf_count_result, leaf_count_optimal,ExpnType_result,ExpnType_optimal;

    leaf_count_result:=leafcount(result);
    #do NOT call ExpnType() if leaf size is too large. Recursion problem
    if leaf_count_result > 500000 then
        return "B";
    fi;

    leaf_count_optimal:=leafcount(optimal);

    ExpnType_result:=ExpnType(result);
    ExpnType_optimal:=ExpnType(optimal);
    #This check below actually is not needed, since I only call this grading only for
    #passed integrals. i.e. I check for "F" before calling this.

    if not type(result,freeof('int')) then
        return "F";
    end if;

    if ExpnType_result<=ExpnType_optimal then
        if is_contains_complex(result) then
            if is_contains_complex(optimal) then
                #both result and optimal complex
                if leaf_count_result<=2*leaf_count_optimal then
                    return "A";
                else
                    return "B";
                end if
            else #result contains complex but optimal is not
                return "C";
            end if
        else # result do not contain complex
            # this assumes optimal do not as well
            if leaf_count_result<=2*leaf_count_optimal then
                return "A";
            else
                return "B";
            end if
        end if
    else #ExpnType(result) > ExpnType(optimal)
        return "C";
    end if
end proc:

```

```

# is_contains_complex(result) takes expressions and returns true if it contains "I"
#Nasser 032417
is_contains_complex:= proc(expression)
  return (has(expression,I));
end proc:

ExpnType := proc(expn)
  if type(expn,'atomic') then
    1
  elif type(expn,'list') then
    apply(max,map(ExpnType,expn))
  elif type(expn,'sqrt') then
    if type(op(1,expn),'rational') then
      1 else
      max(2,ExpnType(op(1,expn)))
    end if
  elif type(expn,``^``) then
    if type(op(2,expn),'integer') then
      ExpnType(op(1,expn))
    elif type(op(2,expn),'rational') then
      if type(op(1,expn),'rational') then
        1 else
        max(2,ExpnType(op(1,expn))) end if else
      max(3,ExpnType(op(1,expn)),ExpnType(op(2,expn)))
    end if
  elif type(expn,``+``) or type(expn,``*``) then
    max(ExpnType(op(1,expn)),max(ExpnType(rest(expn))))
  elif ElementaryFunctionQ(op(0,expn)) then
    max(3,ExpnType(op(1,expn)))
  elif SpecialFunctionQ(op(0,expn)) then
    max(4,apply(max,map(ExpnType,[op(expn)])))
  elif HypergeometricFunctionQ(op(0,expn)) then
    max(5,apply(max,map(ExpnType,[op(expn)])))
  elif AppellFunctionQ(op(0,expn)) then
    max(6,apply(max,map(ExpnType,[op(expn)])))
  elif op(0,expn)='int' or op(0,expn)='integrate' then
    max(8,apply(max,map(ExpnType,[op(expn)]))) else
    9
  end if
end proc:

```

```

ElementaryFunctionQ := proc(func)
  member(func,[exp,log,ln, sin,cos,tan,cot,sec,csc,
    arcsin,arccos,arctan,arccot,arcsec,arccsc,
    sinh,cosh,tanh,coth,sech,csch,
    arcsinh,arccosh,arctanh,arccoth,arcsech,arccsch])
end proc:

SpecialFunctionQ := proc(func)
  member(func,[erf,erfc,erfi,FresnelS,FresnelC,Ei,Ei,Li,Si,Ci,Shi,Chi,
    GAMMA,lnGAMMA,Psi,Zeta,polylog,LambertW,
    EllipticF,EllipticE,EllipticPi])
end proc:

HypergeometricFunctionQ := proc(func)
  member(func,[Hypergeometric1F1,hypergeom,HypergeometricPFQ])
end proc:

AppellFunctionQ := proc(func)
  member(func,[AppellF1])
end proc:

# u is a sum or product. rest(u) returns all but the first term or factor of u.
rest := proc(u) local v;
  if nops(u)=2 then
    op(2,u) else
    apply(op(0,u),op(2..nops(u),u))
  end if
end proc:

#leafcount(u) returns the number of nodes in u.
#Nasser 3/23/17 Replaced by build-in leafCount from package in Maple
leafcount := proc(u)
  MmaTranslator[Mma][LeafCount](u);
end proc:

```