

# Computer Algebra Independent Integration Tests

Summer 2024

7-Inverse-hyperbolic-functions/7.2-Inverse-hyperbolic-  
cosine/331-7.2.1

Nasser M. Abbasi

May 18, 2024

Compiled on May 18, 2024 at 5:37am

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Listing of CAS systems tested . . . . .	4
1.2	Results . . . . .	5
1.3	Time and leaf size Performance . . . . .	9
1.4	Performance based on number of rules Rubi used . . . . .	11
1.5	Performance based on number of steps Rubi used . . . . .	12
1.6	Solved integrals histogram based on leaf size of result . . . . .	13
1.7	Solved integrals histogram based on CPU time used . . . . .	14
1.8	Leaf size vs. CPU time used . . . . .	15
1.9	list of integrals with no known antiderivative . . . . .	16
1.10	List of integrals solved by CAS but has no known antiderivative . . . . .	16
1.11	list of integrals solved by CAS but failed verification . . . . .	16
1.12	Timing . . . . .	17
1.13	Verification . . . . .	17
1.14	Important notes about some of the results . . . . .	18
1.15	Current tree layout of integration tests . . . . .	21
1.16	Design of the test system . . . . .	22
<b>2</b>	<b>detailed summary tables of results</b>	<b>23</b>
2.1	List of integrals sorted by grade for each CAS . . . . .	24
2.2	Detailed conclusion table per each integral for all CAS systems . . . . .	28
2.3	Detailed conclusion table specific for Rubi results . . . . .	32
<b>3</b>	<b>Listing of integrals</b>	<b>33</b>
3.1	$\int (a + \operatorname{barccosh}(cx))^4 dx$ . . . . .	34
3.2	$\int (a + \operatorname{barccosh}(cx))^3 dx$ . . . . .	41
3.3	$\int (a + \operatorname{barccosh}(cx))^2 dx$ . . . . .	47
3.4	$\int (a + \operatorname{barccosh}(cx)) dx$ . . . . .	53
3.5	$\int \frac{1}{a + \operatorname{barccosh}(cx)} dx$ . . . . .	58
3.6	$\int \frac{1}{(a + \operatorname{barccosh}(cx))^2} dx$ . . . . .	65

3.7	$\int \frac{1}{(a+b\operatorname{arccosh}(cx))^3} dx$	72
3.8	$\int (a + \operatorname{arccosh}(cx))^{5/2} dx$	81
3.9	$\int (a + \operatorname{arccosh}(cx))^{3/2} dx$	90
3.10	$\int \sqrt{a + \operatorname{arccosh}(cx)} dx$	98
3.11	$\int \frac{1}{\sqrt{a+b\operatorname{arccosh}(cx)}} dx$	105
3.12	$\int \frac{1}{(a+b\operatorname{arccosh}(cx))^{3/2}} dx$	112
3.13	$\int \frac{1}{(a+b\operatorname{arccosh}(cx))^{5/2}} dx$	120
3.14	$\int \frac{1}{(a+b\operatorname{arccosh}(cx))^{7/2}} dx$	128
3.15	$\int (a + \operatorname{arccosh}(cx))^n dx$	138
<b>4</b>	<b>Appendix</b>	<b>144</b>
4.1	Listing of Grading functions	144
4.2	Links to plain text integration problems used in this report for each CAS162	

# CHAPTER 1

## INTRODUCTION

1.1	Listing of CAS systems tested . . . . .	4
1.2	Results . . . . .	5
1.3	Time and leaf size Performance . . . . .	9
1.4	Performance based on number of rules Rubi used . . . . .	11
1.5	Performance based on number of steps Rubi used . . . . .	12
1.6	Solved integrals histogram based on leaf size of result . . . . .	13
1.7	Solved integrals histogram based on CPU time used . . . . .	14
1.8	Leaf size vs. CPU time used . . . . .	15
1.9	list of integrals with no known antiderivative . . . . .	16
1.10	List of integrals solved by CAS but has no known antiderivative . . . . .	16
1.11	list of integrals solved by CAS but failed verification . . . . .	16
1.12	Timing . . . . .	17
1.13	Verification . . . . .	17
1.14	Important notes about some of the results . . . . .	18
1.15	Current tree layout of integration tests . . . . .	21
1.16	Design of the test system . . . . .	22

This report gives the result of running the computer algebra independent integration test. The download section in on the main webpage contains links to download the problems in plain text format used for all CAS systems. The number of integrals in this report is [ 15 ]. This is test number [ 331 ].

## 1.1 Listing of CAS systems tested

The following are the CAS systems tested:

1. Mathematica 14 (January 9, 2024) on windows 10 pro.
2. Rubi 4.17.3 (Sept 25, 2023) on Mathematica 14 on windows 10m pro.
3. Maple 2024 (March 1, 2024) on windows 10 pro.
4. Maxima 5.47 (June 1, 2023) using Lisp SBCL 2.4.0 on Linux Manjaro 23.1.2 KDE via sagemath 10.3.
5. FriCAS 1.3.10 built with sbcl 2.3.11 (January 10, 2024) on Linux Manjaro 23.1.2 KDE via sagemath 10.3.
6. Giac/Xcas 1.9.0-99 on Linux via sagemath 10.3.
7. Sympy 1.12 using Python 3.11.6 (Nov 14 2023, 09:36:21) [GCC 13.2.1 20230801] on Linux Manjaro 23.1.2 KDE.
8. Mupad using Matlab 2021a with Symbolic Math Toolbox Version 8.7 on windows 10.
9. Reduce CSL rev 6687 (January 9, 2024) on Linux Manjaro 23.1.2 KDE.

Maxima and Fricas and Giac are called using Sagemath. This was done using Sagemath `integrate` command by changing the name of the algorithm to use the different CAS systems.

Sympy was run directly in Python not via sagemath.

Reduce was called directly.

## 1.2 Results

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	100.00 ( 15 )	0.00 ( 0 )
Mathematica	100.00 ( 15 )	0.00 ( 0 )
Maple	46.67 ( 7 )	53.33 ( 8 )
Fricas	26.67 ( 4 )	73.33 ( 11 )
Giac	26.67 ( 4 )	73.33 ( 11 )
Maxima	26.67 ( 4 )	73.33 ( 11 )
Mupad	6.67 ( 1 )	93.33 ( 14 )
Reduce	6.67 ( 1 )	93.33 ( 14 )
Sympy	0.00 ( 0 )	100.00 ( 15 )

Table 1.1: Percentage solved for each CAS

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 <ol 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> </ol>
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.

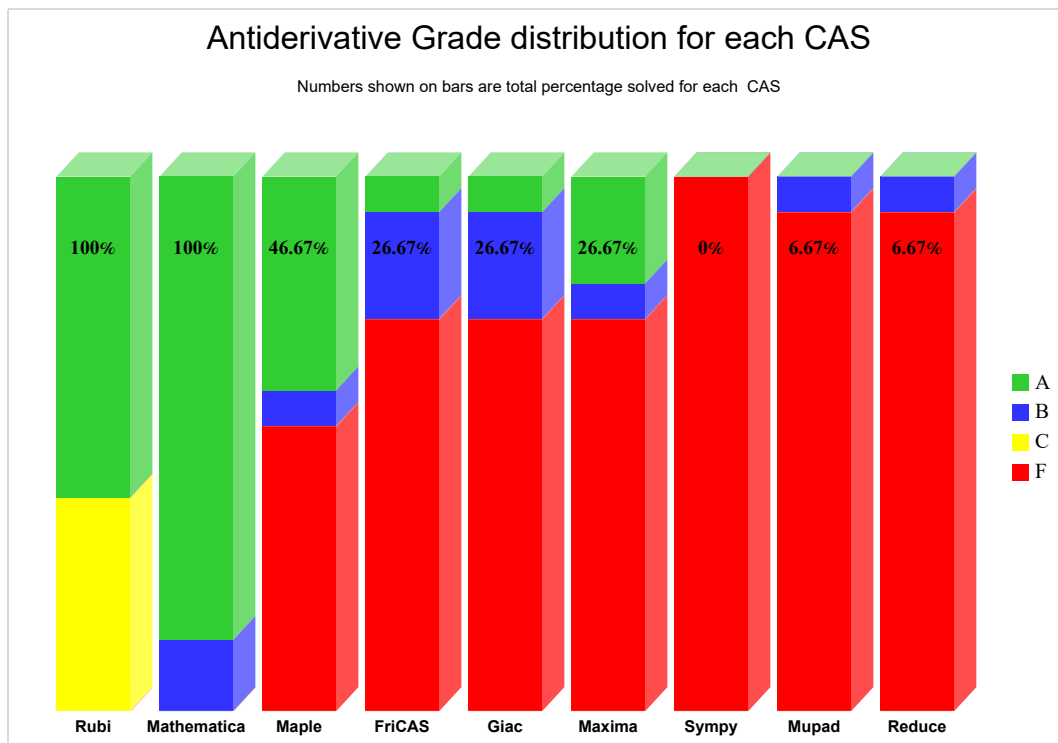
Table 1.2: Description of grading applied to integration result

Grading is implemented for all CAS systems. Based on the above, the following table summarizes the grading for this test suite.

System	% A grade	% B grade	% C grade	% F grade
Mathematica	86.667	13.333	0.000	0.000
Rubi	60.000	0.000	40.000	0.000
Maple	40.000	6.667	0.000	53.333
Maxima	20.000	6.667	0.000	73.333
Fricas	6.667	20.000	0.000	73.333
Giac	6.667	20.000	0.000	73.333
Mupad	0.000	6.667	0.000	93.333
Reduce	0.000	6.667	0.000	93.333
Sympy	0.000	0.000	0.000	100.000

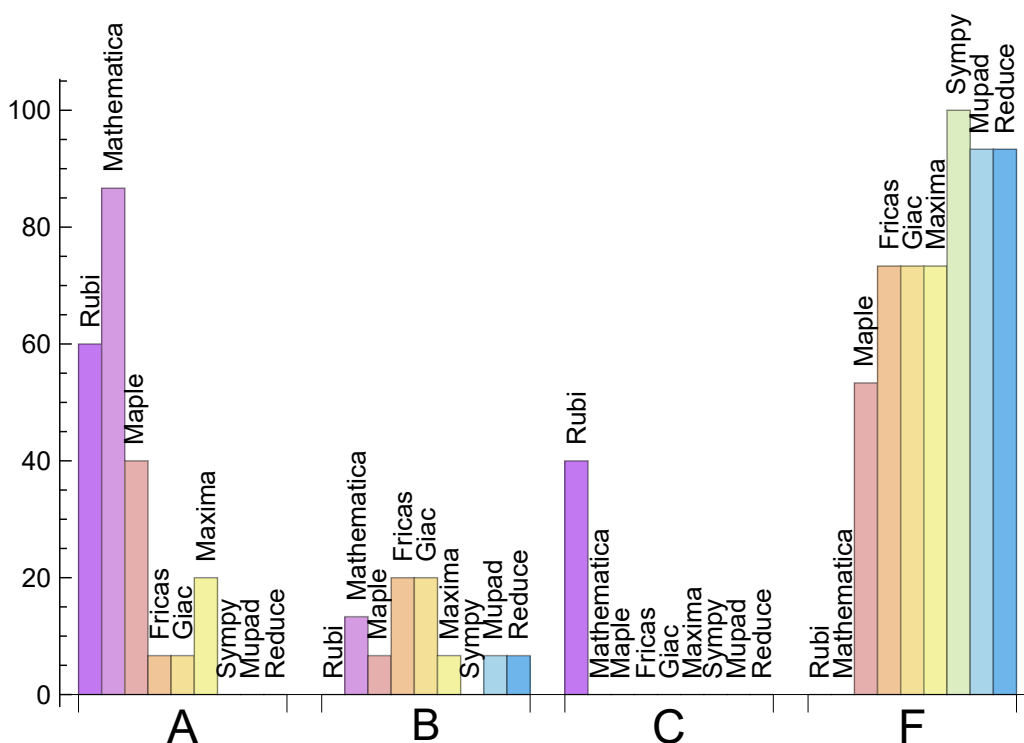
Table 1.3: Antiderivative Grade distribution of each CAS

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





The figure below compares the grades of the CAS systems.



The following table shows the distribution of the different types of failures for each CAS. There are 3 types failures. The first is when CAS returns the input within the time limit, which means it could not solve it. This is the typical failure and given as **F**.

The second failure is due to time out. CAS could not solve the integral within the 3 minutes time limit which is assigned. This is assigned **F(-1)**.

The third is due to an exception generated, indicated as **F(-2)**. This most likely indicates an interface problem between sagemath and the CAS (applicable only to FriCAS, Maxima and Giac) or it could be an indication of an internal error in the CAS itself. This type of error requires more investigation to determine the cause.

System	Number failed	Percentage normal failure	Percentage time-out failure	Percentage exception failure
Rubi	0	0.00	0.00	0.00
Mathematica	0	0.00	0.00	0.00
Maple	8	100.00	0.00	0.00
Fricas	11	36.36	0.00	63.64
Giac	11	90.91	0.00	9.09
Maxima	11	100.00	0.00	0.00
Mupad	14	0.00	100.00	0.00
Reduce	14	100.00	0.00	0.00
Sympy	15	86.67	13.33	0.00

Table 1.4: Failure statistics for each CAS

## 1.3 Time and leaf size Performance

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

Mean size is the average leaf size produced by the CAS (before any normalization). The Normalized mean is relative to the mean size of the optimal anti-derivative given in the input files.

For example, if CAS has **Normalized mean** of 3, then the mean size of its leaf size is 3 times as large as the mean size of the optimal leaf size.

Median size is value of leaf size where half the values are larger than this and half are smaller (before any normalization). i.e. The Middle value.

Similarly the **Normalized median** is relative to the median leaf size of the optimal.

For example, if a CAS has Normalized median of 1.2, then its median is 1.2 as large as the median leaf size of the optimal.

System	Mean time (sec)
Maxima	0.03
Maple	0.11
Fricas	0.15
Reduce	0.17
Mathematica	0.30
Giac	0.34
Rubi	0.88
Mupad	3.20
Sympy	-nan(ind)

Table 1.5: Time performance for each CAS

System	Mean size	Normalized mean	Median size	Normalized median
Mupad	31.00	0.89	31.00	0.89
Reduce	32.00	0.91	32.00	0.91
Rubi	114.13	1.06	111.00	1.08
Maxima	117.75	1.53	106.50	1.52
Maple	119.00	1.46	125.00	1.53
Fricas	133.25	1.81	128.50	1.87
Mathematica	150.40	1.35	100.00	1.14
Giac	179.75	2.31	164.00	2.34
Sympy	-nan(ind)	-nan(ind)	nan	nan

Table 1.6: Leaf size performance for each CAS

## 1.4 Performance based on number of rules Rubi used

This section shows how each CAS performed based on the number of rules Rubi needed to solve the same integral. One diagram is given for each CAS.

On the  $y$  axis is the percentage solved which Rubi itself needed the number of rules given the  $x$  axis. These plots show that as more rules are needed then most CAS system percentage of solving decreases which indicates the integral is becoming more complicated to solve.

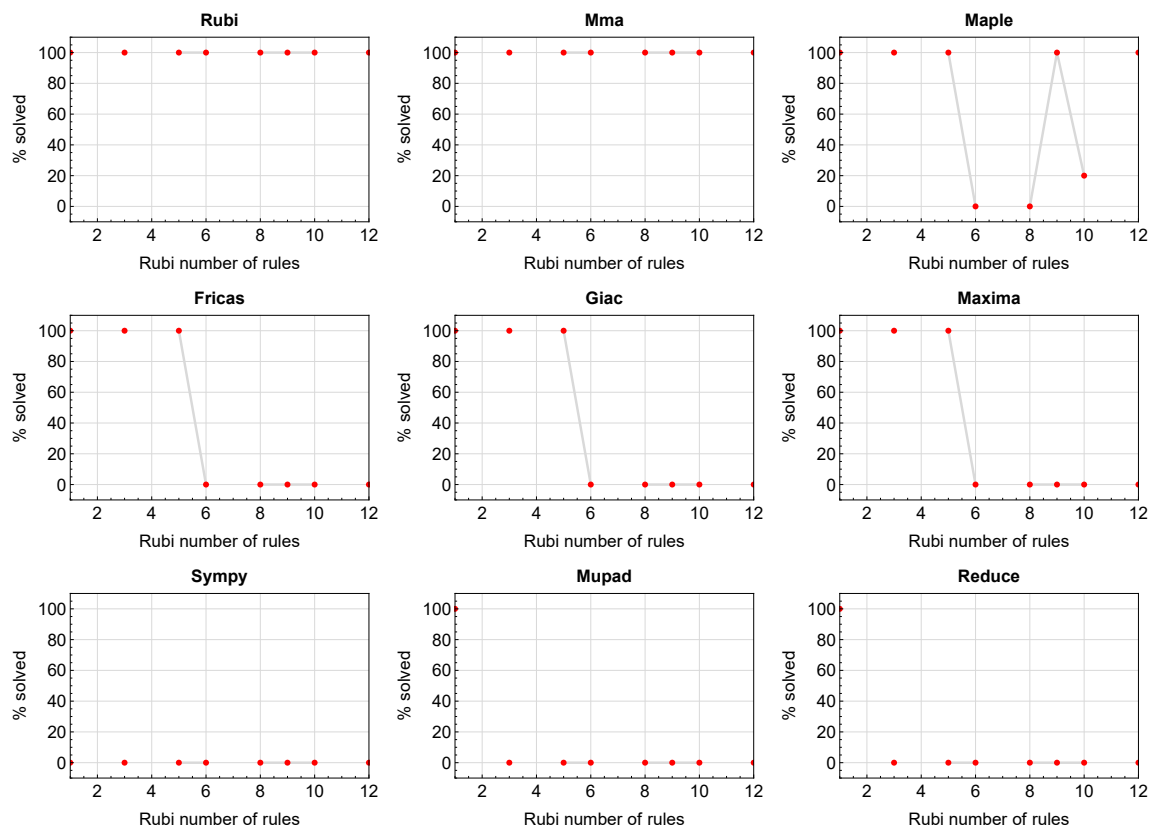


Figure 1.1: Solving statistics per number of Rubi rules used

## 1.5 Performance based on number of steps Rubi used

This section shows how each CAS performed based on the number of steps Rubi needed to solve the same integral. Note that the number of steps Rubi needed can be much higher than the number of rules, as the same rule could be used more than once.

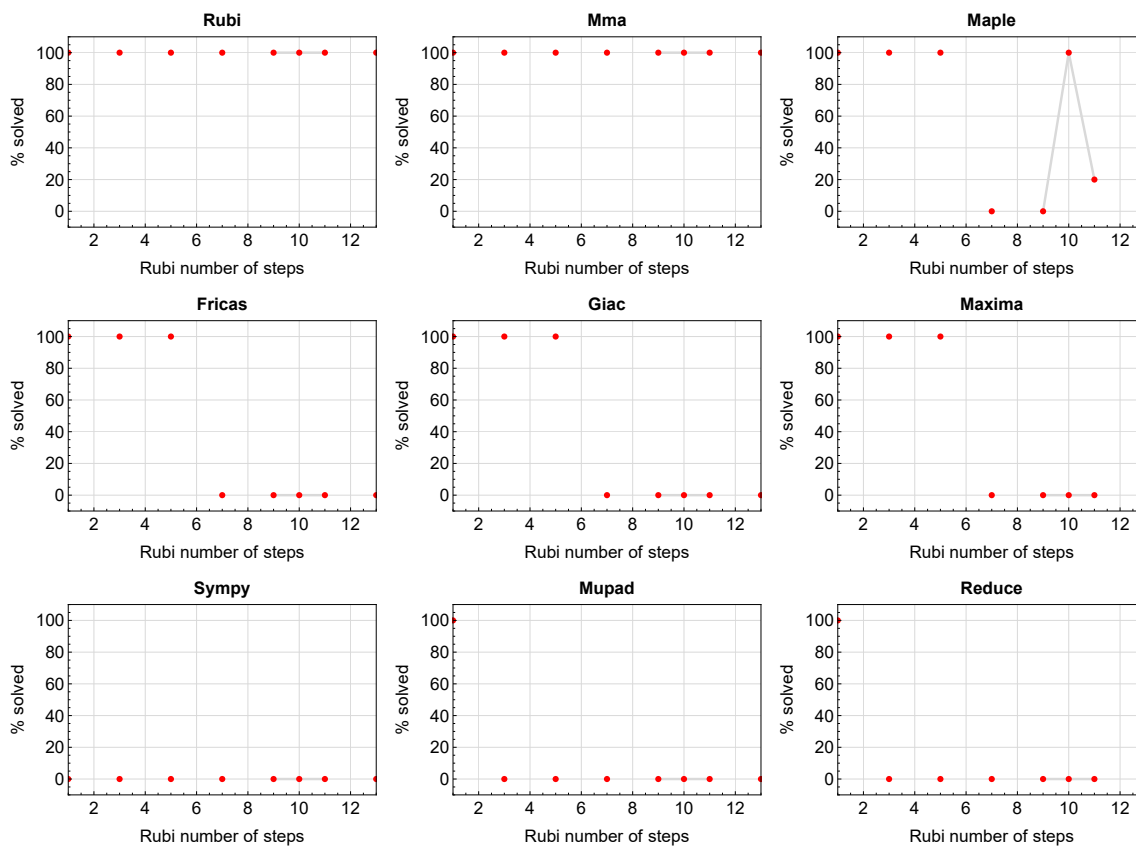


Figure 1.2: Solving statistics per number of Rubi steps used

The above diagram shows that the percentage of solved integrals decreases for most CAS systems as the number of steps increases. As expected, for integrals that required less steps by Rubi, CAS systems had more success which indicates the integral was not as hard to solve. As Rubi needed more steps to solve the integral, the solved percentage decreased for most CAS systems which indicates the integral is becoming harder to solve.

## 1.6 Solved integrals histogram based on leaf size of result

The following shows the distribution of solved integrals for each CAS system based on leaf size of the antiderivatives produced by each CAS. It shows that most integrals solved produced leaf size less than about 100 to 150. The bin size used is 40.

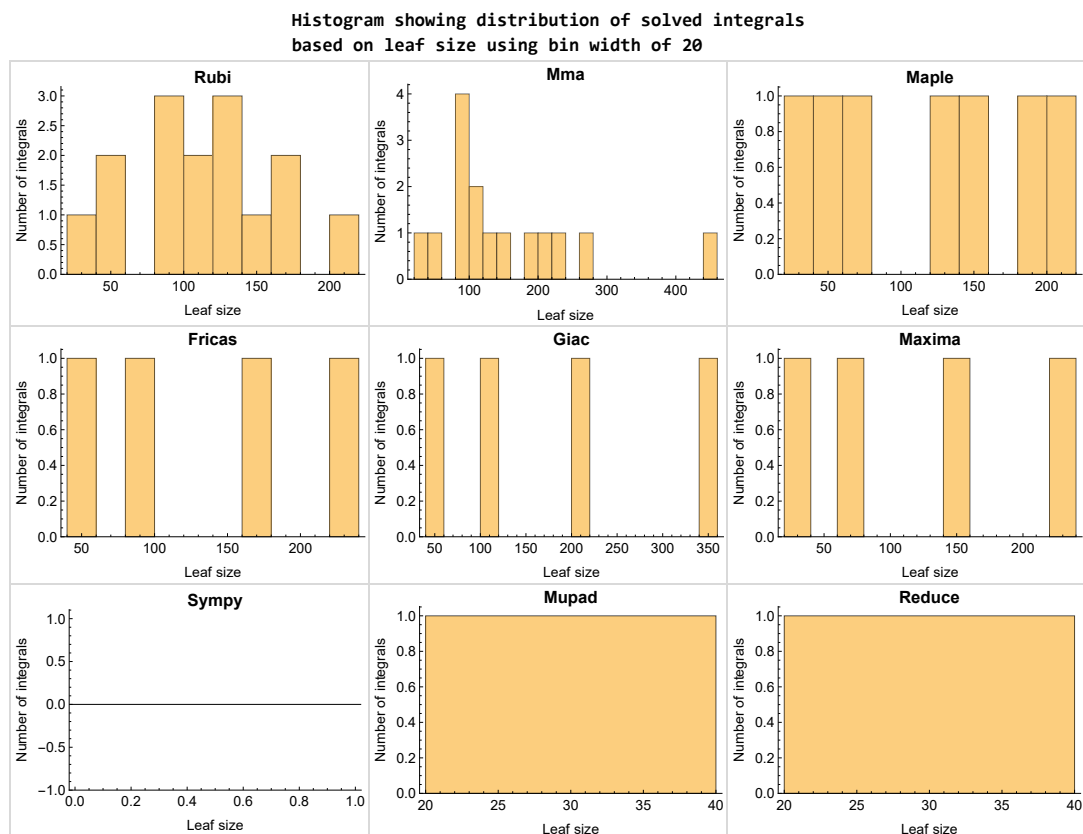


Figure 1.3: Solved integrals based on leaf size distribution

## 1.7 Solved integrals histogram based on CPU time used

The following shows the distribution of solved integrals for each CAS system based on CPU time used in seconds. The bin size used is 0.1 second.

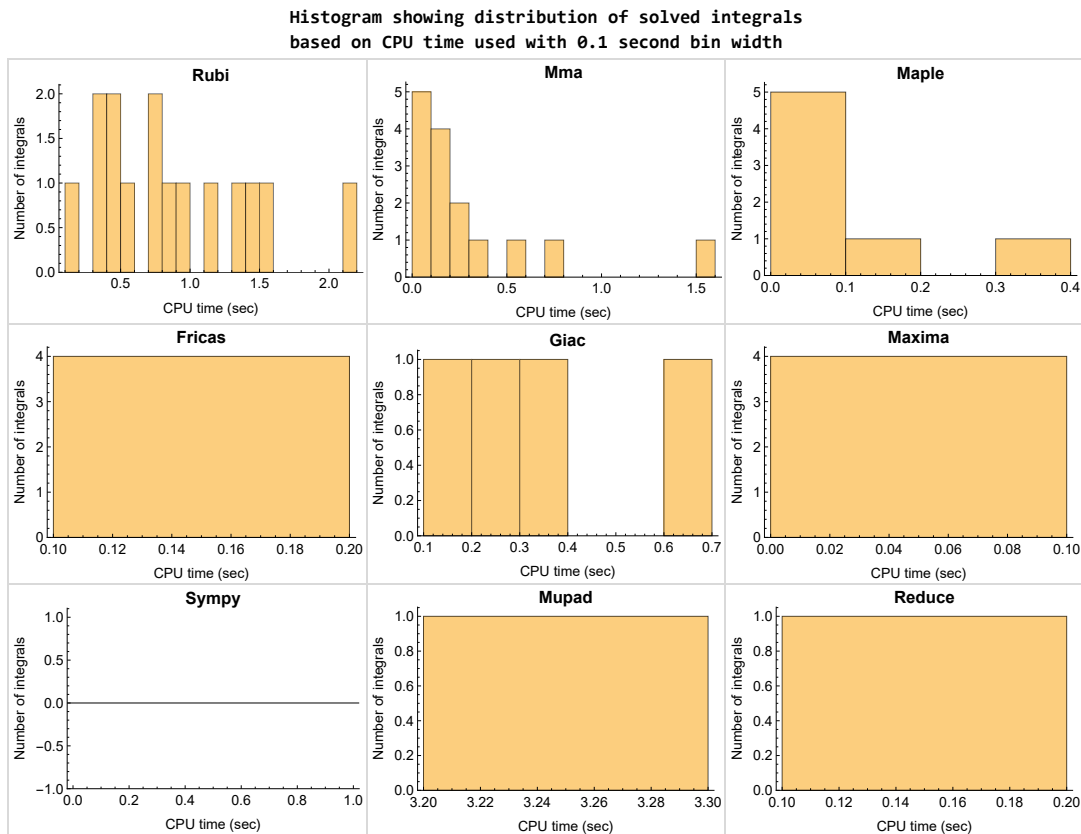


Figure 1.4: Solved integrals histogram based on CPU time used

## 1.8 Leaf size vs. CPU time used

The following shows the relation between the CPU time used to solve an integral and the leaf size of the antiderivative.

The result for Fracas, Maxima and Giac is shifted more to the right than the other CAS system due to the use of sagemath to call them, which causes an initial slight delay in the timing to start the integration due to overhead of starting a new process each time. This should also be taken into account when looking at the timing of these three CAS systems. Direct calls not using sagemath would result in faster timings, but current implementation uses sagemath as this makes testing much easier to do.

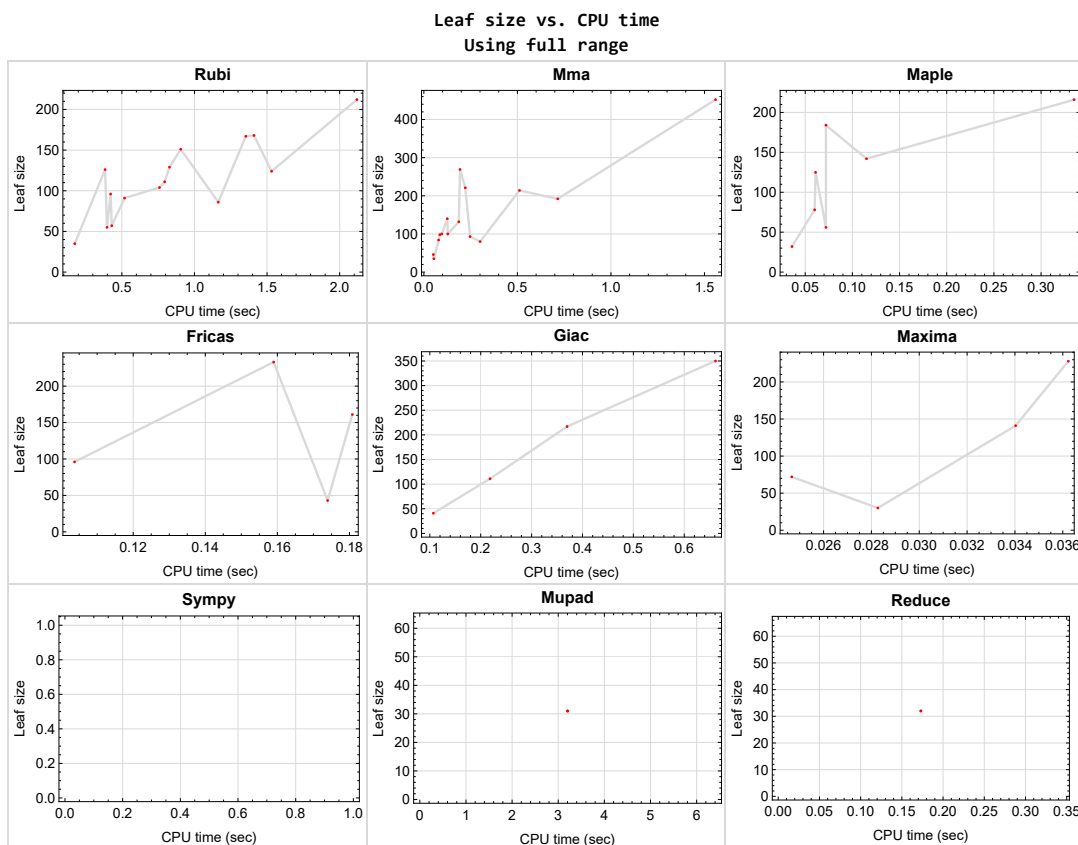


Figure 1.5: Leaf size vs. CPU time. Full range



## 1.9 list of integrals with no known antiderivative

{}

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

Rubi {}

Mathematica {}

Maple {}

Maxima {}

Fricas {}

Sympy {}

Giac {}

Reduce {}

Mupad {}

## 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 necessarily mean 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 possible to do further investigation to determine why the result could not be verified.

Rubi {}

Mathematica {6, 8, 9, 12, 13, 14}

Maple {}

**Maxima** Verification phase not currently implemented.

**Fricas** Verification phase not currently implemented.

**Sympy** Verification phase not currently implemented.

**Giac** Verification phase not currently implemented.

**Reduce** Verification phase not currently implemented.

**Mupad** Verification phase not currently implemented.

## 1.12 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 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 CPU 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 was not counted in the final statistics.

## 1.13 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 was correct.

Verification phase also had 3 minutes time out. An integral whose result was not verified could still be correct, but further investigation is needed on those integrals. These integrals were marked in the summary table below and also in each integral separate section so they are easy to identify and locate.

## 1.14 Important notes about some of the results

### Important note about Maxima results

Since tests were run in a batch mode, and using an automated script, then any integral where Maxima needed an interactive response from the user to answer a question during the evaluation of the integral will fail.

The exception raised is `ValueError`. Therefore Maxima results is lower than what would result if Maxima was run directly and each question 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 14 such integrals out of total 705, or about 2 percent. This percentage can be higher or lower depending on the specific input test file.

Such integrals can be identified by looking at the output of the integration in each section for Maxima. The exception message will indicate the cause of error.

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 automatic loading of Maxima `abs_integrate` was found to cause some problems. 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 result

There were few integrals which failed due to SageMath interface and not because FriCAS system could not do the integration.

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. These can be identified by looking at the exception message given in the result.

## Important note about finding leaf size of antiderivative

For Mathematica, Rubi, and Maple, the builtin system function `LeafSize` was 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 for Fricas and Sympy antiderivative was determined using the following function, thanks to user `slelievre` at [https://ask.sagemath.org/question/57123/could-we-have-a-leaf\\_count-function-in-base-sagemath/](https://ask.sagemath.org/question/57123/could-we-have-a-leaf_count-function-in-base-sagemath/)

```
def tree_size(expr):
    r"""
    Return the tree size of this expression.
    """
    if expr not in SR:
        # deal with lists, tuples, vectors
        return 1 + sum(tree_size(a) for a in expr)
    expr = SR(expr)
    x, aa = expr.operator(), expr.operands()
    if x is None:
        return 1
    else:
        return 1 + sum(tree_size(a) for a in aa)
```

For Sympy, which was called directly from Python, the following code was used to obtain the leafsize of its result

```
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
```

### Important note about Mupad results

Matlab's symbolic toolbox does not have a leaf count function to measure the size of the antiderivative. Maple was used to determine the leaf size of Mupad output by post processing Mupad result.

Currently no grading of the antiderivative for Mupad is implemented. If it can integrate the problem, it was assigned a B grade automatically as a placeholder. In the future, when grading function is implemented for Mupad, the tests will be rerun again.

The following is an example of using Matlab's symbolic toolbox (Mupad) to solve an integral

```
integrand = evalin(symengine, 'cos(x)*sin(x)')  
the_variable = evalin(symengine, 'x')  
anti = int(integrand, the_variable)
```

Which gives  $\sin(x)^2/2$

# 1.15 Current tree layout of integration tests

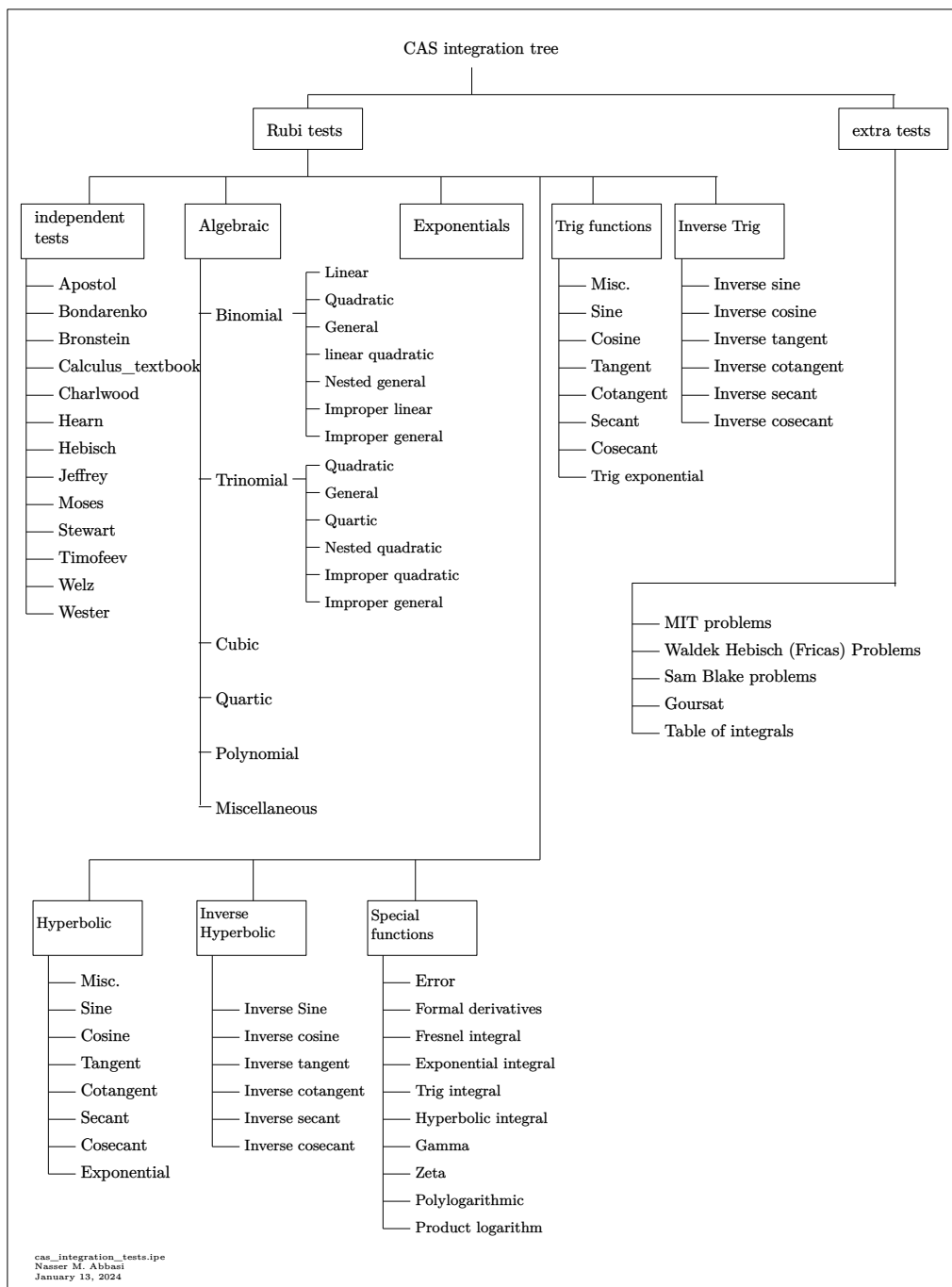
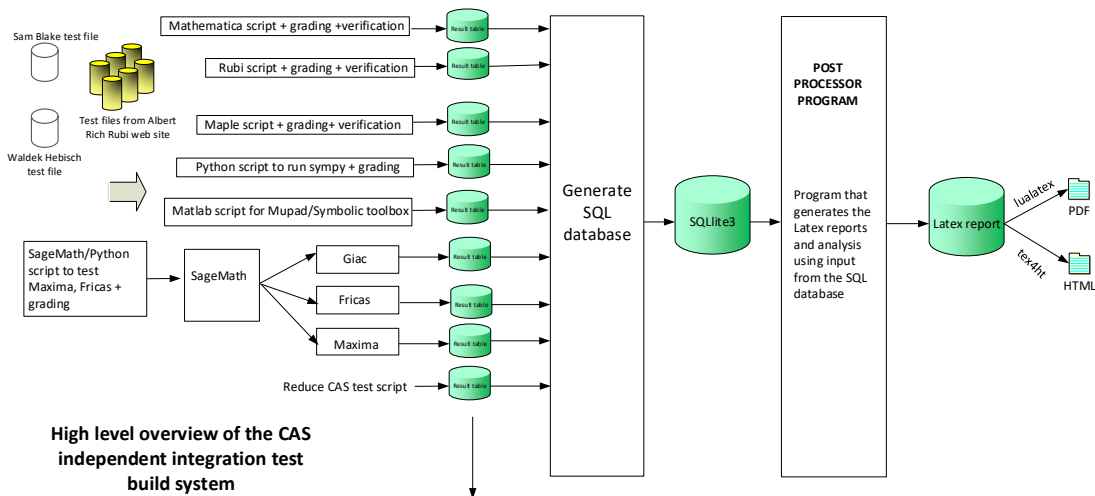


Figure 1.6: CAS integration tests tree

# 1.16 Design of the test system

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



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

One record (line) per one integral result. The line is CSV comma separated. This is description of each record

1. integer, the problem number.
2. integer. 0 for failed, 1 for passed, -1 for timeout, -2 for CAS specific exception. (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 syntax.
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 "E"
13. String. Small string description of why the grade was given.
14. integer. 1 if result was verified or 0 if not verified. (For mma, rubi and maple only)

*The following fields are present only in Rubi Table file*

15. integer. Number of steps used.
16. integer. Number of rules used.
17. integer. Integrand leaf size.
18. real number. Ratio. Field 16 over field 17
19. String of form "{n,n,...}" which is list of the rules used by Rubi
20. String. The optimal antiderivative in Mathematica syntax

Nasser M. Abbasi  
January 13, 2024  
Design note

# CHAPTER 2

## DETAILED SUMMARY TABLES OF RESULTS

2.1	List of integrals sorted by grade for each CAS . . . . .	24
2.2	Detailed conclusion table per each integral for all CAS systems . . . . .	28
2.3	Detailed conclusion table specific for Rubi results . . . . .	32



## 2.1 List of integrals sorted by grade for each CAS

Rubi . . . . .	24
Mma . . . . .	24
Maple . . . . .	25
Fricas . . . . .	25
Maxima . . . . .	25
Giac . . . . .	26
Mupad . . . . .	26
Sympy . . . . .	26
Reduce . . . . .	27

### Rubi

**A grade** { 1, 2, 3, 4, 6, 8, 10, 12, 14 }

**B grade** { }

**C grade** { 5, 7, 9, 11, 13, 15 }

**F normal fail** { }

**F(-1) timeout fail** { }

**F(-2) exception fail** { }

### Mma

**A grade** { 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15 }

**B grade** { 1, 8 }

**C grade** { }

**F normal fail** { }

**F(-1) timeout fail** { }

**F(-2) exception fail** { }

## Maple

A grade { 2, 3, 4, 5, 6, 7 }

B grade { 1 }

C grade { }

F normal fail { 8, 9, 10, 11, 12, 13, 14, 15 }

F(-1) timedout fail { }

F(-2) exception fail { }

## Fricas

A grade { 4 }

B grade { 1, 2, 3 }

C grade { }

F normal fail { 5, 6, 7, 15 }

F(-1) timedout fail { }

F(-2) exception fail { 8, 9, 10, 11, 12, 13, 14 }

## Maxima

A grade { 2, 3, 4 }

B grade { 1 }

C grade { }

F normal fail { 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 }

F(-1) timedout fail { }

F(-2) exception fail { }

## Giac

A grade { 4 }

B grade { 1, 2, 3 }

C grade { }

F normal fail { 5, 6, 7, 9, 10, 11, 12, 13, 14, 15 }

F(-1) timeout fail { }

F(-2) exception fail { 8 }

## Mupad

A grade { }

B grade { 4 }

C grade { }

F normal fail { }

F(-1) timeout fail { 1, 2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 }

F(-2) exception fail { }

## Sympy

A grade { }

B grade { }

C grade { }

F normal fail { 1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 13, 15 }

F(-1) timeout fail { 8, 14 }

F(-2) exception fail { }

## Reduce

A grade { }

B grade { 4 }

C grade { }

F normal fail { 1, 2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 }

F(-1) timedout fail { }

F(-2) exception fail { }

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

Detailed conclusion table per each integral is given by the 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 as **F**.

In this table, the column **N.S.** means **normalized size** and is defined as  $\frac{\text{antiderivative leaf size}}{\text{optimal antiderivative leaf size}}$ . To make the table fit the page, the name **Mathematica** was abbreviated to **MMA**.

Problem 1	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	A	B	B	B	B	<b>F</b>	B	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	103	111	221	216	228	233	0	350	86	0
N.S.	1	1.08	2.15	2.10	2.21	2.26	0.00	3.40	0.83	0.00
time (sec)	N/A	0.794	0.221	0.335	0.036	0.159	0.000	0.661	0.205	0.000

Problem 2	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	A	A	A	A	B	<b>F</b>	B	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	87	91	140	142	141	161	0	217	69	0
N.S.	1	1.05	1.61	1.63	1.62	1.85	0.00	2.49	0.79	0.00
time (sec)	N/A	0.519	0.125	0.115	0.034	0.181	0.000	0.370	0.201	0.000

Problem 3	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	A	A	A	A	B	<b>F</b>	B	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	51	55	84	78	72	96	0	111	50	0
N.S.	1	1.08	1.65	1.53	1.41	1.88	0.00	2.18	0.98	0.00
time (sec)	N/A	0.398	0.079	0.060	0.025	0.104	0.000	0.218	0.183	0.000

Problem 4	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	A	A	A	A	A	<b>F</b>	A	B	B
verified	N/A	Yes	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	35	35	35	32	30	43	0	41	32	31
N.S.	1	1.00	1.00	0.91	0.86	1.23	0.00	1.17	0.91	0.89
time (sec)	N/A	0.176	0.053	0.036	0.028	0.174	0.000	0.107	0.173	3.202

Problem 5	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	C	A	A	<b>F</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	54	57	46	56	0	0	0	0	12	0
N.S.	1	1.06	0.85	1.04	0.00	0.00	0.00	0.00	0.22	0.00
time (sec)	N/A	0.430	0.051	0.072	0.000	0.000	0.000	0.000	0.196	0.000

Problem 6	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	A	A	A	<b>F</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	<b>No</b>	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	90	86	80	125	0	0	0	0	26	0
N.S.	1	0.96	0.89	1.39	0.00	0.00	0.00	0.00	0.29	0.00
time (sec)	N/A	1.164	0.301	0.061	0.000	0.000	0.000	0.000	0.190	0.000

Problem 7	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	C	A	A	<b>F</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	115	124	93	184	0	0	0	0	40	0
N.S.	1	1.08	0.81	1.60	0.00	0.00	0.00	0.00	0.35	0.00
time (sec)	N/A	1.531	0.247	0.072	0.000	0.000	0.000	0.000	0.200	0.000

Problem 8	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	A	B	<b>F</b>	<b>F</b>	<b>F(-2)</b>	<b>F(-1)</b>	<b>F(-2)</b>	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	<b>No</b>	N/A	TBD	TBD	TBD	TBD	TBD	TBD
size	160	168	452	0	0	0	0	0	58	0
N.S.	1	1.05	2.82	0.00	0.00	0.00	0.00	0.00	0.36	0.00
time (sec)	N/A	1.410	1.559	0.000	0.000	0.000	0.000	0.000	0.503	0.000

Problem 9	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	C	A	<b>F</b>	<b>F</b>	<b>F(-2)</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	<b>No</b>	N/A	TBD	TBD	TBD	TBD	TBD	TBD
size	140	151	269	0	0	0	0	0	32	0
N.S.	1	1.08	1.92	0.00	0.00	0.00	0.00	0.00	0.23	0.00
time (sec)	N/A	0.906	0.193	0.000	0.000	0.000	0.000	0.000	0.390	0.000

Problem 10	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	A	A	<b>F</b>	<b>F</b>	<b>F(-2)</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	Yes	N/A	TBD	TBD	TBD	TBD	TBD	TBD
size	102	104	100	0	0	0	0	0	11	0
N.S.	1	1.02	0.98	0.00	0.00	0.00	0.00	0.00	0.11	0.00
time (sec)	N/A	0.759	0.128	0.000	0.000	0.000	0.000	0.000	0.220	0.000

Problem 11	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	C	A	<b>F</b>	<b>F</b>	<b>F(-2)</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	Yes	N/A	TBD	TBD	TBD	TBD	TBD	TBD
size	88	96	100	0	0	0	0	0	22	0
N.S.	1	1.09	1.14	0.00	0.00	0.00	0.00	0.00	0.25	0.00
time (sec)	N/A	0.423	0.096	0.000	0.000	0.000	0.000	0.000	0.211	0.000

Problem 12	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	A	A	<b>F</b>	<b>F</b>	<b>F(-2)</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	<b>No</b>	N/A	TBD	TBD	TBD	TBD	TBD	TBD
size	120	129	132	0	0	0	0	0	842	0
N.S.	1	1.08	1.10	0.00	0.00	0.00	0.00	0.00	7.02	0.00
time (sec)	N/A	0.828	0.187	0.000	0.000	0.000	0.000	0.000	0.729	0.000

Problem 13	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	C	A	<b>F</b>	<b>F</b>	<b>F(-2)</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	<b>No</b>	N/A	TBD	TBD	TBD	TBD	TBD	TBD
size	148	167	192	0	0	0	0	0	0	0
N.S.	1	1.13	1.30	0.00	0.00	0.00	0.00	0.00	0.00	0.00
time (sec)	N/A	1.354	0.716	0.000	0.000	0.000	0.000	0.000	0.814	0.000

Problem 14	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	A	A	<b>F</b>	<b>F</b>	<b>F(-2)</b>	<b>F(-1)</b>	<b>F</b>	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	<b>No</b>	N/A	TBD	TBD	TBD	TBD	TBD	TBD
size	188	212	214	0	0	0	0	0	0	0
N.S.	1	1.13	1.14	0.00	0.00	0.00	0.00	0.00	0.00	0.00
time (sec)	N/A	2.118	0.511	0.000	0.000	0.000	0.000	0.000	0.861	0.000

Problem 15	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Reduce	Mupad
grade	N/A	C	A	<b>F</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F</b>	<b>F(-1)</b>
verified	N/A	Yes	Yes	N/A	TBD	TBD	TBD	TBD	TBD	TBD
size	116	126	98	0	0	0	0	0	12	0
N.S.	1	1.09	0.84	0.00	0.00	0.00	0.00	0.00	0.10	0.00
time (sec)	N/A	0.384	0.085	0.000	0.000	0.000	0.000	0.000	0.164	0.000



## 2.3 Detailed conclusion table specific for Rubi results

The following table is specific to Rubi only. 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 also given. The larger this ratio is, the harder the integral is to solve. In this test file, problem number [7] had the largest ratio of [1.1999999999999996]

Table 2.1: 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	5	5	1.08	10	0.500
2	A	3	3	1.05	10	0.300
3	A	3	3	1.08	10	0.300
4	A	1	1	1.00	8	0.125
5	C	11	10	1.06	10	1.000
6	A	10	9	0.96	10	0.900
7	C	13	12	1.08	10	1.200
8	A	11	10	1.05	12	0.833
9	C	11	10	1.08	12	0.833
10	A	9	8	1.02	12	0.667
11	C	9	8	1.09	12	0.667
12	A	9	8	1.08	12	0.667
13	C	11	10	1.13	12	0.833
14	A	11	10	1.13	12	0.833
15	C	7	6	1.09	10	0.600

# CHAPTER 3

## LISTING OF INTEGRALS

3.1	$\int (a + \operatorname{barccosh}(cx))^4 dx$ . . . . .	34
3.2	$\int (a + \operatorname{barccosh}(cx))^3 dx$ . . . . .	41
3.3	$\int (a + \operatorname{barccosh}(cx))^2 dx$ . . . . .	47
3.4	$\int (a + \operatorname{barccosh}(cx)) dx$ . . . . .	53
3.5	$\int \frac{1}{a + \operatorname{barccosh}(cx)} dx$ . . . . .	58
3.6	$\int \frac{1}{(a + \operatorname{barccosh}(cx))^2} dx$ . . . . .	65
3.7	$\int \frac{1}{(a + \operatorname{barccosh}(cx))^3} dx$ . . . . .	72
3.8	$\int (a + \operatorname{barccosh}(cx))^{5/2} dx$ . . . . .	81
3.9	$\int (a + \operatorname{barccosh}(cx))^{3/2} dx$ . . . . .	90
3.10	$\int \sqrt{a + \operatorname{barccosh}(cx)} dx$ . . . . .	98
3.11	$\int \frac{1}{\sqrt{a + \operatorname{barccosh}(cx)}} dx$ . . . . .	105
3.12	$\int \frac{1}{(a + \operatorname{barccosh}(cx))^{3/2}} dx$ . . . . .	112
3.13	$\int \frac{1}{(a + \operatorname{barccosh}(cx))^{5/2}} dx$ . . . . .	120
3.14	$\int \frac{1}{(a + \operatorname{barccosh}(cx))^{7/2}} dx$ . . . . .	128
3.15	$\int (a + \operatorname{barccosh}(cx))^n dx$ . . . . .	138

### 3.1 $\int (a + \operatorname{barccosh}(cx))^4 dx$

Optimal result	34
Mathematica [B] (verified)	34
Rubi [A] (verified)	35
Maple [B] (verified)	37
Fricas [B] (verification not implemented)	37
Sympy [F]	38
Maxima [B] (verification not implemented)	38
Giac [B] (verification not implemented)	39
Mupad [F(-1)]	40
Reduce [F]	40

#### Optimal result

Integrand size = 10, antiderivative size = 103

$$\int (a + \operatorname{barccosh}(cx))^4 dx = 24b^4x - \frac{24b^3\sqrt{-1+cx}\sqrt{1+cx}(a + \operatorname{barccosh}(cx))}{c} + 12b^2x(a + \operatorname{barccosh}(cx))^2 - \frac{4b\sqrt{-1+cx}\sqrt{1+cx}(a + \operatorname{barccosh}(cx))^3}{c} + x(a + \operatorname{barccosh}(cx))^4$$

output

```
24*b^4*x-24*b^3*(c*x-1)^(1/2)*(c*x+1)^(1/2)*(a+b*arccosh(c*x))/c+12*b^2*x*(a+b*arccosh(c*x))^2-4*b*(c*x-1)^(1/2)*(c*x+1)^(1/2)*(a+b*arccosh(c*x))^3/c+x*(a+b*arccosh(c*x))^4
```

#### Mathematica [B] (verified)

Leaf count is larger than twice the leaf count of optimal. 221 vs. 2(103) = 206.

Time = 0.22 (sec) , antiderivative size = 221, normalized size of antiderivative = 2.15

$$\int (a + \operatorname{barccosh}(cx))^4 dx = \frac{(a^4 + 12a^2b^2 + 24b^4)cx - 4ab(a^2 + 6b^2)\sqrt{-1+cx}\sqrt{1+cx} + 4b(a^3cx + 6ab^2cx - 3a^2b\sqrt{-1+cx}\sqrt{1+cx} + c^2x^2)}{c^2}$$

input `Integrate[(a + b*ArcCosh[c*x])^4,x]`

output 
$$\frac{((a^4 + 12*a^2*b^2 + 24*b^4)*c*x - 4*a*b*(a^2 + 6*b^2)*\sqrt{-1 + c*x}*\sqrt{1 + c*x} + 4*b*(a^3*c*x + 6*a*b^2*c*x - 3*a^2*b*\sqrt{-1 + c*x}*\sqrt{1 + c*x}) - 6*b^3*\sqrt{-1 + c*x}*\sqrt{1 + c*x})*\text{ArcCosh}[c*x] + 6*b^2*(a^2*c*x + 2*b^2*c*x - 2*a*b*\sqrt{-1 + c*x}*\sqrt{1 + c*x})*\text{ArcCosh}[c*x]^2 + 4*b^3*(a*c*x - b*\sqrt{-1 + c*x}*\sqrt{1 + c*x})*\text{ArcCosh}[c*x]^3 + b^4*c*x*\text{ArcCosh}[c*x]^4}{c}$$

### Rubi [A] (verified)

Time = 0.79 (sec) , antiderivative size = 111, normalized size of antiderivative = 1.08, number of steps used = 5, number of rules used = 5,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.500$ , Rules used = {6294, 6330, 6294, 6330, 24}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\int (a + \text{barccosh}(cx))^4 dx$$

$$\downarrow 6294$$

$$x(a + \text{barccosh}(cx))^4 - 4bc \int \frac{x(a + \text{barccosh}(cx))^3}{\sqrt{cx-1}\sqrt{cx+1}} dx$$

$$\downarrow 6330$$

$$4bc \left( \frac{x(a + \text{barccosh}(cx))^4 - \sqrt{cx-1}\sqrt{cx+1}(a + \text{barccosh}(cx))^3}{c^2} - \frac{3b \int (a + \text{barccosh}(cx))^2 dx}{c} \right)$$

$$\downarrow 6294$$

$$4bc \left( \frac{x(a + \text{barccosh}(cx))^4 - \sqrt{cx-1}\sqrt{cx+1}(a + \text{barccosh}(cx))^3}{c^2} - \frac{3b \left( x(a + \text{barccosh}(cx))^2 - 2bc \int \frac{x(a + \text{barccosh}(cx))}{\sqrt{cx-1}\sqrt{cx+1}} dx \right)}{c} \right)$$

$$\downarrow 6330$$

$$4bc \left( \frac{x(a + \operatorname{barccosh}(cx))^4 - \sqrt{cx-1}\sqrt{cx+1}(a + \operatorname{barccosh}(cx))^3}{c^2} - \frac{3b(x(a + \operatorname{barccosh}(cx))^2 - 2bc \left( \frac{\sqrt{cx-1}\sqrt{cx+1}(a + \operatorname{barccosh}(cx))}{c^2} - \frac{bx}{c} \right))}{c} \right)$$

↓ 24

$$4bc \left( \frac{x(a + \operatorname{barccosh}(cx))^4 - \sqrt{cx-1}\sqrt{cx+1}(a + \operatorname{barccosh}(cx))^3}{c^2} - \frac{3b(x(a + \operatorname{barccosh}(cx))^2 - 2bc \left( \frac{\sqrt{cx-1}\sqrt{cx+1}(a + \operatorname{barccosh}(cx))}{c^2} - \frac{bx}{c} \right))}{c} \right)$$

input `Int[(a + b*ArcCosh[c*x])^4,x]`

output `x*(a + b*ArcCosh[c*x])^4 - 4*b*c*((Sqrt[-1 + c*x]*Sqrt[1 + c*x]*(a + b*ArcCosh[c*x])^3)/c^2 - (3*b*(x*(a + b*ArcCosh[c*x])^2 - 2*b*c*(-((b*x)/c) + (Sqrt[-1 + c*x]*Sqrt[1 + c*x]*(a + b*ArcCosh[c*x]))) / c^2)) / c)`

### Defintions of rubi rules used

rule 24 `Int[a_, x_Symbol] := Simp[a*x, x] /; FreeQ[a, x]`

rule 6294 `Int[((a_.) + ArcCosh[(c_.)*(x_.)]*(b_.))^n, x_Symbol] := Simp[x*(a + b*ArcCosh[c*x])^n, x] - Simp[b*c*n Int[x*(a + b*ArcCosh[c*x])^(n-1)/(Sqrt[1 + c*x]*Sqrt[-1 + c*x]), x], x] /; FreeQ[{a, b, c}, x] && GtQ[n, 0]`

rule 6330 `Int[((a_.) + ArcCosh[(c_.)*(x_.)]*(b_.))^n*(x_)*((d1_) + (e1_.)*(x_))^(p_)*((d2_) + (e2_.)*(x_))^(q_), x_Symbol] := Simp[(d1 + e1*x)^(p+1)*(d2 + e2*x)^(q+1)*((a + b*ArcCosh[c*x])^n/(2*e1*e2*(p+1))), x] - Simp[b*(n/(2*c*(p+1)))*Simp[(d1 + e1*x)^p/(1 + c*x)^p]*Simp[(d2 + e2*x)^q/(-1 + c*x)^p] Int[(1 + c*x)^(p+1/2)*(-1 + c*x)^(q+1/2)*(a + b*ArcCosh[c*x])^(n-1), x], x] /; FreeQ[{a, b, c, d1, e1, d2, e2, p}, x] && EqQ[e1, c*d1] && EqQ[e2, (-c)*d2] && GtQ[n, 0] && NeQ[p, -1]`

**Maple [B] (verified)**

Leaf count of result is larger than twice the leaf count of optimal. 215 vs.  $2(95) = 190$ .

Time = 0.34 (sec) , antiderivative size = 216, normalized size of antiderivative = 2.10

method	result
derivativedivides	$\frac{cx a^4 + b^4 (\operatorname{arccosh}(cx)^4 cx - 4 \operatorname{arccosh}(cx)^3 \sqrt{cx-1} \sqrt{cx+1} + 12 \operatorname{arccosh}(cx)^2 cx - 24 \operatorname{arccosh}(cx) \sqrt{cx-1} \sqrt{cx+1} + 24 cx)}{c}$
default	$\frac{cx a^4 + b^4 (\operatorname{arccosh}(cx)^4 cx - 4 \operatorname{arccosh}(cx)^3 \sqrt{cx-1} \sqrt{cx+1} + 12 \operatorname{arccosh}(cx)^2 cx - 24 \operatorname{arccosh}(cx) \sqrt{cx-1} \sqrt{cx+1} + 24 cx)}{c}$
parts	$x a^4 + \frac{b^4 (\operatorname{arccosh}(cx)^4 cx - 4 \operatorname{arccosh}(cx)^3 \sqrt{cx-1} \sqrt{cx+1} + 12 \operatorname{arccosh}(cx)^2 cx - 24 \operatorname{arccosh}(cx) \sqrt{cx-1} \sqrt{cx+1} + 24 cx)}{c}$
oring	$x(a + b \operatorname{arccosh}(cx))^4 + \frac{8(a+b \operatorname{arccosh}(cx))^3 b}{c \sqrt{cx-1} \sqrt{cx+1}} + \frac{(5c^2 x^2 - 2)x \left( \frac{12b^2 c^2 (a+b \operatorname{arccosh}(cx))^2}{(cx-1)(cx+1)} - \frac{2(a+b \operatorname{arccosh}(cx))}{(cx-1)^{\frac{3}{2}} \sqrt{cx+1}} \right)}{c^2}$

input `int((a+b*arccosh(c*x))^4,x,method=_RETURNVERBOSE)`

output 
$$\begin{aligned} & 1/c*(c*x*a^4+b^4*(\operatorname{arccosh}(c*x))^4*c*x-4*\operatorname{arccosh}(c*x)^3*(c*x-1)^{(1/2)}*(c*x+1) \\ & ^{(1/2)}+12*\operatorname{arccosh}(c*x)^2*c*x-24*\operatorname{arccosh}(c*x)*(c*x-1)^{(1/2)}*(c*x+1)^{(1/2)}+ \\ & 24*c*x)+4*a*b^3*(\operatorname{arccosh}(c*x))^3*c*x-3*\operatorname{arccosh}(c*x)^2*(c*x-1)^{(1/2)}*(c*x+1) \\ & ^{(1/2)}+6*c*x*\operatorname{arccosh}(c*x)-6*(c*x-1)^{(1/2)}*(c*x+1)^{(1/2}))+6*a^2*b^2*(\operatorname{arccos} \\ & h(c*x)^2*c*x-2*\operatorname{arccosh}(c*x)*(c*x-1)^{(1/2)}*(c*x+1)^{(1/2)}+2*c*x)+4*a^3*b*(c* \\ & x*\operatorname{arccosh}(c*x)-(c*x-1)^{(1/2)}*(c*x+1)^{(1/2}))) \end{aligned}$$

**Fricas [B] (verification not implemented)**

Leaf count of result is larger than twice the leaf count of optimal. 233 vs.  $2(95) = 190$ .

Time = 0.16 (sec) , antiderivative size = 233, normalized size of antiderivative = 2.26

$$\int (a + b \operatorname{arccosh}(cx))^4 dx = \frac{b^4 cx \log(cx + \sqrt{c^2 x^2 - 1})^4 + 4(ab^3 cx - \sqrt{c^2 x^2 - 1} b^4) \log(cx + \sqrt{c^2 x^2 - 1})^3 + (a^4 + 12 a^2 b^2 + 24 b^4) cx}{c}$$

input `integrate((a+b*arccosh(c*x))^4,x, algorithm="fricas")`

output

```
(b^4*c*x*log(c*x + sqrt(c^2*x^2 - 1))^4 + 4*(a*b^3*c*x - sqrt(c^2*x^2 - 1)
*b^4)*log(c*x + sqrt(c^2*x^2 - 1))^3 + (a^4 + 12*a^2*b^2 + 24*b^4)*c*x - 6
*(2*sqrt(c^2*x^2 - 1)*a*b^3 - (a^2*b^2 + 2*b^4)*c*x)*log(c*x + sqrt(c^2*x^
2 - 1))^2 + 4*((a^3*b + 6*a*b^3)*c*x - 3*(a^2*b^2 + 2*b^4)*sqrt(c^2*x^2 -
1))*log(c*x + sqrt(c^2*x^2 - 1)) - 4*(a^3*b + 6*a*b^3)*sqrt(c^2*x^2 - 1))/
c
```

**Sympy [F]**

$$\int (a + b \operatorname{arccosh}(cx))^4 dx = \int (a + b \operatorname{acosh}(cx))^4 dx$$

input

```
integrate((a+b*acosh(c*x))**4,x)
```

output

```
Integral((a + b*acosh(c*x))**4, x)
```

**Maxima [B] (verification not implemented)**

Leaf count of result is larger than twice the leaf count of optimal. 228 vs. 2(95) = 190.

Time = 0.04 (sec) , antiderivative size = 228, normalized size of antiderivative = 2.21

$$\begin{aligned} & \int (a + b \operatorname{arccosh}(cx))^4 dx \\ &= b^4 x \operatorname{arccosh}(cx)^4 + 4 ab^3 x \operatorname{arccosh}(cx)^3 + 6 a^2 b^2 x \operatorname{arccosh}(cx)^2 \\ & \quad - 12 \left( \frac{\sqrt{c^2 x^2 - 1} \operatorname{arccosh}(cx)^2}{c} - \frac{2 (cx \operatorname{arccosh}(cx) - \sqrt{c^2 x^2 - 1})}{c} \right) ab^3 \\ & \quad - 4 \left( \frac{\sqrt{c^2 x^2 - 1} \operatorname{arccosh}(cx)^3}{c} - 3 \left( \frac{x \operatorname{arccosh}(cx)^2}{c} + \frac{2 \left( x - \frac{\sqrt{c^2 x^2 - 1} \operatorname{arccosh}(cx)}{c} \right)}{c} \right) c \right) b^4 \\ & \quad + 12 a^2 b^2 \left( x - \frac{\sqrt{c^2 x^2 - 1} \operatorname{arccosh}(cx)}{c} \right) + a^4 x + \frac{4 (cx \operatorname{arccosh}(cx) - \sqrt{c^2 x^2 - 1}) a^3 b}{c} \end{aligned}$$

input

```
integrate((a+b*arccosh(c*x))^4,x, algorithm="maxima")
```

output

```

b^4*x*arccosh(c*x)^4 + 4*a*b^3*x*arccosh(c*x)^3 + 6*a^2*b^2*x*arccosh(c*x)
^2 - 12*(sqrt(c^2*x^2 - 1)*arccosh(c*x)^2/c - 2*(c*x*arccosh(c*x) - sqrt(c
^2*x^2 - 1))/c)*a*b^3 - 4*(sqrt(c^2*x^2 - 1)*arccosh(c*x)^3/c - 3*(x*arcco
sh(c*x)^2/c + 2*(x - sqrt(c^2*x^2 - 1)*arccosh(c*x)/c)/c)*c)*b^4 + 12*a^2*
b^2*(x - sqrt(c^2*x^2 - 1)*arccosh(c*x)/c) + a^4*x + 4*(c*x*arccosh(c*x) -
sqrt(c^2*x^2 - 1))*a^3*b/c

```

### Giac [B] (verification not implemented)

Leaf count of result is larger than twice the leaf count of optimal. 350 vs.  $2(95) = 190$ .

Time = 0.66 (sec) , antiderivative size = 350, normalized size of antiderivative = 3.40

$$\begin{aligned}
 \int (a + b \operatorname{arccosh}(cx))^4 dx &= 4 \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right) - \frac{\sqrt{c^2 x^2 - 1}}{c} \right) a^3 b \\
 &+ 6 \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right)^2 + 2c \left( \frac{x}{c} - \frac{\sqrt{c^2 x^2 - 1} \log \left( cx + \sqrt{c^2 x^2 - 1} \right)}{c^2} \right) \right) a^2 b^2 \\
 &+ 4 \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right)^3 - 3c \left( \frac{\sqrt{c^2 x^2 - 1} \log \left( cx + \sqrt{c^2 x^2 - 1} \right)^2}{c^2} - \frac{2 \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right) - \frac{\sqrt{c^2 x^2 - 1}}{c} \right)}{c} \right) \right) a b^3 \\
 &+ \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right)^4 - 4 \left( \frac{\sqrt{c^2 x^2 - 1} \log \left( cx + \sqrt{c^2 x^2 - 1} \right)^3}{c^2} - \frac{3 \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right)^2 + 2 \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right) - \frac{\sqrt{c^2 x^2 - 1}}{c} \right)}{c} \right)}{c} \right) \right) a^2 b^2 \\
 &+ a^4 x
 \end{aligned}$$

input

```
integrate((a+b*arccosh(c*x))^4,x, algorithm="giac")
```

output

```

4*(x*log(c*x + sqrt(c^2*x^2 - 1)) - sqrt(c^2*x^2 - 1)/c)*a^3*b + 6*(x*log(c
*x + sqrt(c^2*x^2 - 1))^2 + 2*c*(x/c - sqrt(c^2*x^2 - 1)*log(c*x + sqrt(c
^2*x^2 - 1))/c^2))*a^2*b^2 + 4*(x*log(c*x + sqrt(c^2*x^2 - 1))^3 - 3*c*(sq
rt(c^2*x^2 - 1)*log(c*x + sqrt(c^2*x^2 - 1))^2/c^2 - 2*(x*log(c*x + sqrt(c
^2*x^2 - 1)) - sqrt(c^2*x^2 - 1)/c)/c)*a*b^3 + (x*log(c*x + sqrt(c^2*x^2
- 1))^4 - 4*(sqrt(c^2*x^2 - 1)*log(c*x + sqrt(c^2*x^2 - 1))^3/c^2 - 3*(x*1
og(c*x + sqrt(c^2*x^2 - 1))^2 + 2*c*(x/c - sqrt(c^2*x^2 - 1)*log(c*x + sq
rt(c^2*x^2 - 1))/c^2))/c)*c)*b^4 + a^4*x

```



**Mupad [F(-1)]**

Timed out.

$$\int (a + b \operatorname{arccosh}(cx))^4 dx = \int (a + b \operatorname{acosh}(cx))^4 dx$$

input `int((a + b*acosh(c*x))^4,x)`output `int((a + b*acosh(c*x))^4, x)`**Reduce [F]**

$$\int (a + b \operatorname{arccosh}(cx))^4 dx$$

$$= \frac{4 \operatorname{acosh}(cx) a^3 b c x - 4 \sqrt{cx+1} \sqrt{cx-1} a^3 b + (\int \operatorname{acosh}(cx)^4 dx) b^4 c + 4 (\int \operatorname{acosh}(cx)^3 dx) a b^3 c + 6 (\int \operatorname{acosh}(cx)^2 dx) a^2 b^2 c + a^4 c x}{c}$$

input `int((a+b*acosh(c*x))^4,x)`output `(4*acosh(c*x)*a**3*b*c*x - 4*sqrt(c*x + 1)*sqrt(c*x - 1)*a**3*b + int(acosh(c*x)**4,x)*b**4*c + 4*int(acosh(c*x)**3,x)*a*b**3*c + 6*int(acosh(c*x)**2,x)*a**2*b**2*c + a**4*c*x)/c`

### 3.2 $\int (a + \operatorname{barccosh}(cx))^3 dx$

Optimal result	41
Mathematica [A] (verified)	41
Rubi [A] (verified)	42
Maple [A] (verified)	43
Fricas [B] (verification not implemented)	44
Sympy [F]	44
Maxima [A] (verification not implemented)	45
Giac [B] (verification not implemented)	45
Mupad [F(-1)]	46
Reduce [F]	46

#### Optimal result

Integrand size = 10, antiderivative size = 87

$$\int (a + \operatorname{barccosh}(cx))^3 dx = -\frac{6b^3\sqrt{-1+cx}\sqrt{1+cx}}{c} + 6b^2x(a + \operatorname{barccosh}(cx)) - \frac{3b\sqrt{-1+cx}\sqrt{1+cx}(a + \operatorname{barccosh}(cx))^2}{c} + x(a + \operatorname{barccosh}(cx))^3$$

output

$$-6*b^3*(c*x-1)^{(1/2)}*(c*x+1)^{(1/2)}/c+6*b^2*x*(a+b*\operatorname{arccosh}(c*x))-3*b*(c*x-1)^{(1/2)}*(c*x+1)^{(1/2)}*(a+b*\operatorname{arccosh}(c*x))^2/c+x*(a+b*\operatorname{arccosh}(c*x))^3$$

#### Mathematica [A] (verified)

Time = 0.13 (sec) , antiderivative size = 140, normalized size of antiderivative = 1.61

$$\int (a + \operatorname{barccosh}(cx))^3 dx = \frac{a(a^2 + 6b^2)cx - 3b(a^2 + 2b^2)\sqrt{-1+cx}\sqrt{1+cx} + 3b(a^2cx + 2b^2cx - 2ab\sqrt{-1+cx}\sqrt{1+cx})\operatorname{arccosh}(cx)}{c}$$

input

$$\operatorname{Integrate}[(a + b*\operatorname{ArcCosh}[c*x])^3, x]$$

output

```
(a*(a^2 + 6*b^2)*c*x - 3*b*(a^2 + 2*b^2)*Sqrt[-1 + c*x]*Sqrt[1 + c*x] + 3*
b*(a^2*c*x + 2*b^2*c*x - 2*a*b*Sqrt[-1 + c*x]*Sqrt[1 + c*x])*ArcCosh[c*x]
+ 3*b^2*(a*c*x - b*Sqrt[-1 + c*x]*Sqrt[1 + c*x])*ArcCosh[c*x]^2 + b^3*c*x*
ArcCosh[c*x]^3)/c
```

**Rubi [A] (verified)**

Time = 0.52 (sec) , antiderivative size = 91, normalized size of antiderivative = 1.05, number of steps used = 3, number of rules used = 3,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.300$ , Rules used = {6294, 6330, 2009}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\int (a + b \operatorname{arccosh}(cx))^3 dx$$

$$\downarrow 6294$$

$$x(a + b \operatorname{arccosh}(cx))^3 - 3bc \int \frac{x(a + b \operatorname{arccosh}(cx))^2}{\sqrt{cx-1}\sqrt{cx+1}} dx$$

$$\downarrow 6330$$

$$x(a + b \operatorname{arccosh}(cx))^3 - 3bc \left( \frac{\sqrt{cx-1}\sqrt{cx+1}(a + b \operatorname{arccosh}(cx))^2}{c^2} - \frac{2b \int (a + b \operatorname{arccosh}(cx)) dx}{c} \right)$$

$$\downarrow 2009$$

$$3bc \left( \frac{x(a + b \operatorname{arccosh}(cx))^3 - \sqrt{cx-1}\sqrt{cx+1}(a + b \operatorname{arccosh}(cx))^2}{c^2} - \frac{2b(ax + b \operatorname{arccosh}(cx) - \frac{b\sqrt{cx-1}\sqrt{cx+1}}{c})}{c} \right)$$

input

```
Int[(a + b*ArcCosh[c*x])^3,x]
```

output

```
x*(a + b*ArcCosh[c*x])^3 - 3*b*c*((Sqrt[-1 + c*x]*Sqrt[1 + c*x]*(a + b*Arc
Cosh[c*x])^2)/c^2 - (2*b*(a*x - (b*Sqrt[-1 + c*x]*Sqrt[1 + c*x])/c + b*x*A
rcCosh[c*x]))/c)
```

Defintions of rubi rules used

```
rule 2009 Int[u_, x_Symbol] := Simp[IntSum[u, x], x] /; SumQ[u]
```

```
rule 6294 Int[((a_.) + ArcCosh[(c_.)*(x_)]*(b_.))^n, x_Symbol] := Simp[x*(a + b*ArcCosh[c*x])^n, x] - Simp[b*c*n Int[x*(a + b*ArcCosh[c*x])^(n - 1)/(Sqrt[1 + c*x]*Sqrt[-1 + c*x]), x], x] /; FreeQ[{a, b, c}, x] && GtQ[n, 0]
```

```
rule 6330 Int[((a_.) + ArcCosh[(c_.)*(x_)]*(b_.))^n*(x_)*((d1_) + (e1_.)*(x_))^(p_)*((d2_) + (e2_.)*(x_))^(q_), x_Symbol] := Simp[(d1 + e1*x)^(p + 1)*(d2 + e2*x)^(q + 1)*((a + b*ArcCosh[c*x])^n/(2*e1*e2*(p + 1))), x] - Simp[b*(n/(2*c*(p + 1)))*Simp[(d1 + e1*x)^p/(1 + c*x)^p]*Simp[(d2 + e2*x)^q/(-1 + c*x)^q] Int[(1 + c*x)^(p + 1/2)*(-1 + c*x)^(q + 1/2)*(a + b*ArcCosh[c*x])^(n - 1), x], x] /; FreeQ[{a, b, c, d1, e1, d2, e2, p}, x] && EqQ[e1, c*d1] && EqQ[e2, (-c)*d2] && GtQ[n, 0] && NeQ[p, -1]
```

Maple [A] (verified)

Time = 0.12 (sec) , antiderivative size = 142, normalized size of antiderivative = 1.63

method	result
derivativedivides	$\frac{cx a^3 + b^3 (\operatorname{arccosh}(cx)^3 cx - 3 \operatorname{arccosh}(cx)^2 \sqrt{cx-1} \sqrt{cx+1} + 6cx \operatorname{arccosh}(cx) - 6\sqrt{cx-1} \sqrt{cx+1}) + 3a b^2 (\operatorname{arccosh}(cx)^2 c}{c}$
default	$\frac{cx a^3 + b^3 (\operatorname{arccosh}(cx)^3 cx - 3 \operatorname{arccosh}(cx)^2 \sqrt{cx-1} \sqrt{cx+1} + 6cx \operatorname{arccosh}(cx) - 6\sqrt{cx-1} \sqrt{cx+1}) + 3a b^2 (\operatorname{arccosh}(cx)^2 c}{c}$
parts	$x a^3 + \frac{b^3 (\operatorname{arccosh}(cx)^3 cx - 3 \operatorname{arccosh}(cx)^2 \sqrt{cx-1} \sqrt{cx+1} + 6cx \operatorname{arccosh}(cx) - 6\sqrt{cx-1} \sqrt{cx+1})}{c} + \frac{3a^2 b (cx \operatorname{arccos}}{c}$
oring	$x(a + b \operatorname{arccosh}(cx))^3 - \frac{3(c^2 x^2 - 2)(a + b \operatorname{arccosh}(cx))^2 b}{c \sqrt{cx-1} \sqrt{cx+1}} - \frac{2x(cx-1)(cx+1) \left( \frac{6b^2 c^2 (a + b \operatorname{arccosh}(cx))}{(cx-1)(cx+1)} - \frac{3(a}{c} \right)}{c}$

```
input int((a+b*arccosh(c*x))^3,x,method=_RETURNVERBOSE)
```

output

```
1/c*(c*x*a^3+b^3*(arccosh(c*x)^3*c*x-3*arccosh(c*x)^2*(c*x-1)^(1/2)*(c*x+1)^(1/2)+6*c*x*arccosh(c*x)-6*(c*x-1)^(1/2)*(c*x+1)^(1/2))+3*a*b^2*(arccosh(c*x)^2*c*x-2*arccosh(c*x)*(c*x-1)^(1/2)*(c*x+1)^(1/2)+2*c*x)+3*a^2*b*(c*x*arccosh(c*x)-(c*x-1)^(1/2)*(c*x+1)^(1/2)))
```

### Fricas [B] (verification not implemented)

Leaf count of result is larger than twice the leaf count of optimal. 161 vs.  $2(79) = 158$ .

Time = 0.18 (sec) , antiderivative size = 161, normalized size of antiderivative = 1.85

$$\int (a + b \operatorname{arccosh}(cx))^3 dx$$

$$= \frac{b^3 cx \log(cx + \sqrt{c^2 x^2 - 1})^3 + (a^3 + 6ab^2)cx + 3(ab^2 cx - \sqrt{c^2 x^2 - 1}b^3) \log(cx + \sqrt{c^2 x^2 - 1})^2 - 3(2\sqrt{c^2 x^2 - 1}) \log(cx + \sqrt{c^2 x^2 - 1}) + (a^2 b + 2b^3)cx}{c}$$

input

```
integrate((a+b*arccosh(c*x))^3,x, algorithm="fricas")
```

output

```
(b^3*c*x*log(c*x + sqrt(c^2*x^2 - 1))^3 + (a^3 + 6*a*b^2)*c*x + 3*(a*b^2*c*x - sqrt(c^2*x^2 - 1)*b^3)*log(c*x + sqrt(c^2*x^2 - 1))^2 - 3*(2*sqrt(c^2*x^2 - 1)*a*b^2 - (a^2*b + 2*b^3)*c*x)*log(c*x + sqrt(c^2*x^2 - 1)) - 3*sqrt(c^2*x^2 - 1)*(a^2*b + 2*b^3))/c
```

### Sympy [F]

$$\int (a + b \operatorname{arccosh}(cx))^3 dx = \int (a + b \operatorname{acosh}(cx))^3 dx$$

input

```
integrate((a+b*acosh(c*x))**3,x)
```

output

```
Integral((a + b*acosh(c*x))**3, x)
```

**Maxima [A] (verification not implemented)**

Time = 0.03 (sec) , antiderivative size = 141, normalized size of antiderivative = 1.62

$$\int (a + b \operatorname{arccosh}(cx))^3 dx$$

$$= b^3 x \operatorname{arccosh}(cx)^3 + 3ab^2 x \operatorname{arccosh}(cx)^2$$

$$- 3 \left( \frac{\sqrt{c^2 x^2 - 1} \operatorname{arccosh}(cx)^2}{c} - \frac{2(cx \operatorname{arccosh}(cx) - \sqrt{c^2 x^2 - 1})}{c} \right) b^3$$

$$+ 6ab^2 \left( x - \frac{\sqrt{c^2 x^2 - 1} \operatorname{arccosh}(cx)}{c} \right) + a^3 x + \frac{3(cx \operatorname{arccosh}(cx) - \sqrt{c^2 x^2 - 1})a^2 b}{c}$$

input `integrate((a+b*arccosh(c*x))^3,x, algorithm="maxima")`

output `b^3*x*arccosh(c*x)^3 + 3*a*b^2*x*arccosh(c*x)^2 - 3*(sqrt(c^2*x^2 - 1)*arccosh(c*x)^2/c - 2*(c*x*arccosh(c*x) - sqrt(c^2*x^2 - 1))/c)*b^3 + 6*a*b^2*(x - sqrt(c^2*x^2 - 1)*arccosh(c*x)/c) + a^3*x + 3*(c*x*arccosh(c*x) - sqrt(c^2*x^2 - 1))*a^2*b/c`

**Giac [B] (verification not implemented)**

Leaf count of result is larger than twice the leaf count of optimal. 217 vs. 2(79) = 158.

Time = 0.37 (sec) , antiderivative size = 217, normalized size of antiderivative = 2.49

$$\int (a + b \operatorname{arccosh}(cx))^3 dx = 3 \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right) - \frac{\sqrt{c^2 x^2 - 1}}{c} \right) a^2 b$$

$$+ 3 \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right)^2 + 2c \left( \frac{x}{c} - \frac{\sqrt{c^2 x^2 - 1} \log \left( cx + \sqrt{c^2 x^2 - 1} \right)}{c^2} \right) \right) ab^2$$

$$+ \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right)^3 - 3c \left( \frac{\sqrt{c^2 x^2 - 1} \log \left( cx + \sqrt{c^2 x^2 - 1} \right)^2}{c^2} - \frac{2 \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right) - \frac{\sqrt{c^2 x^2 - 1}}{c} \right)}{c} \right) \right) a^2 b$$

$$+ a^3 x$$

input `integrate((a+b*arccosh(c*x))^3,x, algorithm="giac")`

output

```
3*(x*log(c*x + sqrt(c^2*x^2 - 1)) - sqrt(c^2*x^2 - 1)/c)*a^2*b + 3*(x*log(
c*x + sqrt(c^2*x^2 - 1))^2 + 2*c*(x/c - sqrt(c^2*x^2 - 1))*log(c*x + sqrt(c
^2*x^2 - 1))/c^2))*a*b^2 + (x*log(c*x + sqrt(c^2*x^2 - 1))^3 - 3*c*(sqrt(c
^2*x^2 - 1)*log(c*x + sqrt(c^2*x^2 - 1))^2/c^2 - 2*(x*log(c*x + sqrt(c^2*x
^2 - 1)) - sqrt(c^2*x^2 - 1)/c)/c))*b^3 + a^3*x
```

**Mupad [F(-1)]**

Timed out.

$$\int (a + b \operatorname{arccosh}(cx))^3 dx = \int (a + b \operatorname{acosh}(cx))^3 dx$$

input

```
int((a + b*acosh(c*x))^3,x)
```

output

```
int((a + b*acosh(c*x))^3, x)
```

**Reduce [F]**

$$\int (a + b \operatorname{arccosh}(cx))^3 dx = \frac{3 \operatorname{acosh}(cx) a^2 b c x - 3 \sqrt{cx + 1} \sqrt{cx - 1} a^2 b + \left( \int \operatorname{acosh}(cx)^3 dx \right) b^3 c + 3 \left( \int \operatorname{acosh}(cx)^2 dx \right) a b^2 c + a^3 c x}{c}$$

input

```
int((a+b*acosh(c*x))^3,x)
```

output

```
(3*acosh(c*x)*a**2*b*c*x - 3*sqrt(c*x + 1)*sqrt(c*x - 1)*a**2*b + int(acos
h(c*x)**3,x)*b**3*c + 3*int(acosh(c*x)**2,x)*a*b**2*c + a**3*c*x)/c
```

### 3.3 $\int (a + \operatorname{barccosh}(cx))^2 dx$

Optimal result	47
Mathematica [A] (verified)	47
Rubi [A] (verified)	48
Maple [A] (verified)	49
Fricas [B] (verification not implemented)	50
Sympy [F]	50
Maxima [A] (verification not implemented)	50
Giac [B] (verification not implemented)	51
Mupad [F(-1)]	51
Reduce [F]	52

#### Optimal result

Integrand size = 10, antiderivative size = 51

$$\int (a + \operatorname{barccosh}(cx))^2 dx = 2b^2x - \frac{2b\sqrt{-1 + cx}\sqrt{1 + cx}(a + \operatorname{barccosh}(cx))}{c} + x(a + \operatorname{barccosh}(cx))^2$$

output

```
2*b^2*x-2*b*(c*x-1)^(1/2)*(c*x+1)^(1/2)*(a+b*arccosh(c*x))/c+x*(a+b*arccosh(c*x))^2
```

#### Mathematica [A] (verified)

Time = 0.08 (sec) , antiderivative size = 84, normalized size of antiderivative = 1.65

$$\int (a + \operatorname{barccosh}(cx))^2 dx = (a^2 + 2b^2)x - \frac{2ab\sqrt{-1 + cx}\sqrt{1 + cx}}{c} + \frac{2b(ax - b\sqrt{-1 + cx}\sqrt{1 + cx}) \operatorname{arccosh}(cx)}{c} + b^2x \operatorname{arccosh}(cx)^2$$

input

```
Integrate[(a + b*ArcCosh[c*x])^2,x]
```



output

$$(a^2 + 2b^2)x - (2ab\sqrt{-1 + cx}\sqrt{1 + cx})/c + (2b(ax - b\sqrt{-1 + cx}\sqrt{1 + cx})\operatorname{ArcCosh}[cx])/c + b^2x\operatorname{ArcCosh}[cx]^2$$
**Rubi [A] (verified)**

Time = 0.40 (sec) , antiderivative size = 55, normalized size of antiderivative = 1.08, number of steps used = 3, number of rules used = 3,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.300$ , Rules used = {6294, 6330, 24}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\int (a + b\operatorname{arccosh}(cx))^2 dx$$

$$\downarrow 6294$$

$$x(a + b\operatorname{arccosh}(cx))^2 - 2bc \int \frac{x(a + b\operatorname{arccosh}(cx))}{\sqrt{cx - 1}\sqrt{cx + 1}} dx$$

$$\downarrow 6330$$

$$x(a + b\operatorname{arccosh}(cx))^2 - 2bc \left( \frac{\sqrt{cx - 1}\sqrt{cx + 1}(a + b\operatorname{arccosh}(cx))}{c^2} - \frac{b \int 1 dx}{c} \right)$$

$$\downarrow 24$$

$$x(a + b\operatorname{arccosh}(cx))^2 - 2bc \left( \frac{\sqrt{cx - 1}\sqrt{cx + 1}(a + b\operatorname{arccosh}(cx))}{c^2} - \frac{bx}{c} \right)$$

input

$$\operatorname{Int}[(a + b\operatorname{ArcCosh}[c*x])^2, x]$$

output

$$x*(a + b\operatorname{ArcCosh}[c*x])^2 - 2*b*c*(-((b*x)/c) + (\sqrt{-1 + c*x}*\sqrt{1 + c*x}*(a + b\operatorname{ArcCosh}[c*x]))/c^2)$$

## Definitions of rubi rules used

rule 24 `Int[a_, x_Symbol] := Simp[a*x, x] /; FreeQ[a, x]`

rule 6294 `Int[((a_.) + ArcCosh[(c_.)*(x_.)]*(b_.))^n, x_Symbol] := Simp[x*(a + b*ArcCosh[c*x])^n, x] - Simp[b*c*n Int[x*(a + b*ArcCosh[c*x])^(n - 1)/(Sqrt[1 + c*x]*Sqrt[-1 + c*x]), x], x] /; FreeQ[{a, b, c}, x] && GtQ[n, 0]`

rule 6330 `Int[((a_.) + ArcCosh[(c_.)*(x_.)]*(b_.))^n*(x_)*((d1_) + (e1_.)*(x_))^(p_)*((d2_) + (e2_.)*(x_))^(q_), x_Symbol] := Simp[(d1 + e1*x)^(p + 1)*(d2 + e2*x)^(q + 1)*((a + b*ArcCosh[c*x])^n/(2*e1*e2*(p + 1))), x] - Simp[b*(n/(2*c*(p + 1))*Simp[(d1 + e1*x)^p/(1 + c*x)^p]*Simp[(d2 + e2*x)^q/(-1 + c*x)^q] Int[(1 + c*x)^(p + 1/2)*(-1 + c*x)^(q + 1/2)*(a + b*ArcCosh[c*x])^(n - 1), x], x] /; FreeQ[{a, b, c, d1, e1, d2, e2, p}, x] && EqQ[e1, c*d1] && EqQ[e2, (-c)*d2] && GtQ[n, 0] && NeQ[p, -1]`

## Maple [A] (verified)

Time = 0.06 (sec) , antiderivative size = 78, normalized size of antiderivative = 1.53

method	result
derivativedivides	$\frac{cx a^2 + b^2 \left( \operatorname{arccosh}(cx)^2 cx - 2 \operatorname{arccosh}(cx) \sqrt{cx-1} \sqrt{cx+1} + 2cx \right) + 2ab(cx \operatorname{arccosh}(cx) - \sqrt{cx-1} \sqrt{cx+1})}{c}$
default	$\frac{cx a^2 + b^2 \left( \operatorname{arccosh}(cx)^2 cx - 2 \operatorname{arccosh}(cx) \sqrt{cx-1} \sqrt{cx+1} + 2cx \right) + 2ab(cx \operatorname{arccosh}(cx) - \sqrt{cx-1} \sqrt{cx+1})}{c}$
parts	$x a^2 + \frac{b^2 \left( \operatorname{arccosh}(cx)^2 cx - 2 \operatorname{arccosh}(cx) \sqrt{cx-1} \sqrt{cx+1} + 2cx \right)}{c} + \frac{2ab(cx \operatorname{arccosh}(cx) - \sqrt{cx-1} \sqrt{cx+1})}{c}$
oring	$x(a + b \operatorname{arccosh}(cx))^2 + \frac{2(a + b \operatorname{arccosh}(cx))b}{c\sqrt{cx-1}\sqrt{cx+1}} + \frac{x(cx-1)(cx+1) \left( \frac{2b^2 e^2}{(cx-1)(cx+1)} - \frac{(a+b \operatorname{arccosh}(cx))b c^2}{(cx-1)^{\frac{3}{2}} \sqrt{cx+1}} - \frac{(a-b \operatorname{arccosh}(cx))b c^2}{(cx+1)^{\frac{3}{2}} \sqrt{cx-1}} \right)}{c^2}$

input `int((a+b*arccosh(c*x))^2,x,method=_RETURNVERBOSE)`

output `1/c*(c*x*a^2+b^2*(arccosh(c*x)^2*c*x-2*arccosh(c*x)*(c*x-1)^(1/2)*(c*x+1)^(1/2)+2*c*x)+2*a*b*(c*x*arccosh(c*x)-(c*x-1)^(1/2)*(c*x+1)^(1/2)))`

**Fricas [B] (verification not implemented)**

Leaf count of result is larger than twice the leaf count of optimal. 96 vs.  $2(47) = 94$ .

Time = 0.10 (sec) , antiderivative size = 96, normalized size of antiderivative = 1.88

$$\int (a + \operatorname{barccosh}(cx))^2 dx = \frac{b^2 cx \log(cx + \sqrt{c^2 x^2 - 1})^2 + (a^2 + 2b^2)cx - 2\sqrt{c^2 x^2 - 1}ab + 2(abcx - \sqrt{c^2 x^2 - 1}b^2) \log(cx + \sqrt{c^2 x^2 - 1})}{c}$$

input `integrate((a+b*arccosh(c*x))^2,x, algorithm="fricas")`

output `(b^2*c*x*log(c*x + sqrt(c^2*x^2 - 1))^2 + (a^2 + 2*b^2)*c*x - 2*sqrt(c^2*x^2 - 1)*a*b + 2*(a*b*c*x - sqrt(c^2*x^2 - 1)*b^2)*log(c*x + sqrt(c^2*x^2 - 1)))/c`

**Sympy [F]**

$$\int (a + \operatorname{barccosh}(cx))^2 dx = \int (a + b \operatorname{acosh}(cx))^2 dx$$

input `integrate((a+b*acosh(c*x))**2,x)`

output `Integral((a + b*acosh(c*x))**2, x)`

**Maxima [A] (verification not implemented)**

Time = 0.02 (sec) , antiderivative size = 72, normalized size of antiderivative = 1.41

$$\int (a + \operatorname{barccosh}(cx))^2 dx = b^2 x \operatorname{arcosh}(cx)^2 + 2b^2 \left( x - \frac{\sqrt{c^2 x^2 - 1} \operatorname{arcosh}(cx)}{c} \right) + a^2 x + \frac{2(cx \operatorname{arcosh}(cx) - \sqrt{c^2 x^2 - 1})ab}{c}$$

input `integrate((a+b*arccosh(c*x))^2,x, algorithm="maxima")`

output `b^2*x*arccosh(c*x)^2 + 2*b^2*(x - sqrt(c^2*x^2 - 1))*arccosh(c*x)/c + a^2*x + 2*(c*x*arccosh(c*x) - sqrt(c^2*x^2 - 1))*a*b/c`

### Giac [B] (verification not implemented)

Leaf count of result is larger than twice the leaf count of optimal. 111 vs.  $2(47) = 94$ .

Time = 0.22 (sec) , antiderivative size = 111, normalized size of antiderivative = 2.18

$$\begin{aligned} & \int (a + b \operatorname{arccosh}(cx))^2 dx \\ &= 2 \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right) - \frac{\sqrt{c^2 x^2 - 1}}{c} \right) ab \\ &+ \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right)^2 + 2c \left( \frac{x}{c} - \frac{\sqrt{c^2 x^2 - 1} \log \left( cx + \sqrt{c^2 x^2 - 1} \right)}{c^2} \right) \right) b^2 \\ &+ a^2 x \end{aligned}$$

input `integrate((a+b*arccosh(c*x))^2,x, algorithm="giac")`

output `2*(x*log(c*x + sqrt(c^2*x^2 - 1)) - sqrt(c^2*x^2 - 1)/c)*a*b + (x*log(c*x + sqrt(c^2*x^2 - 1))^2 + 2*c*(x/c - sqrt(c^2*x^2 - 1)*log(c*x + sqrt(c^2*x^2 - 1))/c^2)*b^2 + a^2*x`

### Mupad [F(-1)]

Timed out.

$$\int (a + b \operatorname{arccosh}(cx))^2 dx = \int (a + b \operatorname{acosh}(cx))^2 dx$$

input `int((a + b*acosh(c*x))^2,x)`

output `int((a + b*acosh(c*x))^2, x)`

**Reduce [F]**

$$\int (a + b \operatorname{arccosh}(cx))^2 dx$$

$$= \frac{2a \operatorname{cosh}(cx) abcx - 2\sqrt{cx+1}\sqrt{cx-1}ab + \left(\int a \operatorname{cosh}(cx)^2 dx\right) b^2c + a^2cx}{c}$$

input `int((a+b*acosh(c*x))^2,x)`

output `(2*acosh(c*x)*a*b*c*x - 2*sqrt(c*x + 1)*sqrt(c*x - 1)*a*b + int(acosh(c*x)**2,x)*b**2*c + a**2*c*x)/c`

### 3.4 $\int (a + b \operatorname{arccosh}(cx)) dx$

Optimal result	53
Mathematica [A] (verified)	53
Rubi [A] (verified)	54
Maple [A] (verified)	54
Fricas [A] (verification not implemented)	55
Sympy [F]	55
Maxima [A] (verification not implemented)	56
Giac [A] (verification not implemented)	56
Mupad [B] (verification not implemented)	56
Reduce [B] (verification not implemented)	57

#### Optimal result

Integrand size = 8, antiderivative size = 35

$$\int (a + b \operatorname{arccosh}(cx)) dx = ax - \frac{b\sqrt{-1+cx}\sqrt{1+cx}}{c} + b \operatorname{arccosh}(cx)$$

output

```
a*x-b*(c*x-1)^(1/2)*(c*x+1)^(1/2)/c+b*x*arccosh(c*x)
```

#### Mathematica [A] (verified)

Time = 0.05 (sec) , antiderivative size = 35, normalized size of antiderivative = 1.00

$$\int (a + b \operatorname{arccosh}(cx)) dx = ax - \frac{b\sqrt{-1+cx}\sqrt{1+cx}}{c} + b \operatorname{arccosh}(cx)$$

input

```
Integrate[a + b*ArcCosh[c*x], x]
```

output

```
a*x - (b*Sqrt[-1 + c*x]*Sqrt[1 + c*x])/c + b*x*ArcCosh[c*x]
```

**Rubi [A] (verified)**

Time = 0.18 (sec) , antiderivative size = 35, normalized size of antiderivative = 1.00, number of steps used = 1, number of rules used = 1,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.125$ , Rules used = {2009}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\int (a + b \operatorname{arccosh}(cx)) dx$$

$$\downarrow 2009$$

$$ax + b \operatorname{arccosh}(cx) - \frac{b\sqrt{cx-1}\sqrt{cx+1}}{c}$$

input `Int[a + b*ArcCosh[c*x], x]`

output `a*x - (b*Sqrt[-1 + c*x]*Sqrt[1 + c*x])/c + b*x*ArcCosh[c*x]`

**Defintions of rubi rules used**

rule 2009 `Int[u_, x_Symbol] :> Simp[IntSum[u, x], x] /; SumQ[u]`

**Maple [A] (verified)**

Time = 0.04 (sec) , antiderivative size = 32, normalized size of antiderivative = 0.91

method	result	size
orering	$x(a + b \operatorname{arccosh}(cx)) - \frac{b\sqrt{cx-1}\sqrt{cx+1}}{c}$	32
default	$ax + \frac{b(cx \operatorname{arccosh}(cx) - \sqrt{cx-1}\sqrt{cx+1})}{c}$	34
parts	$ax + \frac{b(cx \operatorname{arccosh}(cx) - \sqrt{cx-1}\sqrt{cx+1})}{c}$	34
derivativedivides	$\frac{cxa + b(cx \operatorname{arccosh}(cx) - \sqrt{cx-1}\sqrt{cx+1})}{c}$	36

input `int(a+b*arccosh(c*x),x,method=_RETURNVERBOSE)`

output `x*(a+b*arccosh(c*x))-b*(c*x-1)^(1/2)*(c*x+1)^(1/2)/c`

### Fricas [A] (verification not implemented)

Time = 0.17 (sec) , antiderivative size = 43, normalized size of antiderivative = 1.23

$$\int (a + b \operatorname{arccosh}(cx)) dx = \frac{bcx \log(cx + \sqrt{c^2x^2 - 1}) + acx - \sqrt{c^2x^2 - 1}b}{c}$$

input `integrate(a+b*arccosh(c*x),x, algorithm="fricas")`

output `(b*c*x*log(c*x + sqrt(c^2*x^2 - 1)) + a*c*x - sqrt(c^2*x^2 - 1)*b)/c`

### Sympy [F]

$$\int (a + b \operatorname{arccosh}(cx)) dx = \int (a + b \operatorname{acosh}(cx)) dx$$

input `integrate(a+b*acosh(c*x),x)`

output `Integral(a + b*acosh(c*x), x)`



**Maxima [A] (verification not implemented)**

Time = 0.03 (sec) , antiderivative size = 30, normalized size of antiderivative = 0.86

$$\int (a + b \operatorname{arccosh}(cx)) dx = ax + \frac{(cx \operatorname{arccosh}(cx) - \sqrt{c^2 x^2 - 1})b}{c}$$

input `integrate(a+b*arccosh(c*x),x, algorithm="maxima")`

output `a*x + (c*x*arccosh(c*x) - sqrt(c^2*x^2 - 1))*b/c`

**Giac [A] (verification not implemented)**

Time = 0.11 (sec) , antiderivative size = 41, normalized size of antiderivative = 1.17

$$\int (a + b \operatorname{arccosh}(cx)) dx = \left( x \log \left( cx + \sqrt{c^2 x^2 - 1} \right) - \frac{\sqrt{c^2 x^2 - 1}}{c} \right) b + ax$$

input `integrate(a+b*arccosh(c*x),x, algorithm="giac")`

output `(x*log(c*x + sqrt(c^2*x^2 - 1)) - sqrt(c^2*x^2 - 1)/c)*b + a*x`

**Mupad [B] (verification not implemented)**

Time = 3.20 (sec) , antiderivative size = 31, normalized size of antiderivative = 0.89

$$\int (a + b \operatorname{arccosh}(cx)) dx = ax + bx \operatorname{acosh}(cx) - \frac{b \sqrt{cx - 1} \sqrt{cx + 1}}{c}$$

input `int(a + b*acosh(c*x),x)`

output `a*x + b*x*acosh(c*x) - (b*(c*x - 1)^(1/2)*(c*x + 1)^(1/2))/c`

**Reduce [B] (verification not implemented)**

Time = 0.17 (sec) , antiderivative size = 32, normalized size of antiderivative = 0.91

$$\int (a + b \operatorname{arccosh}(cx)) dx = \frac{a \operatorname{cosh}(cx) bcx - \sqrt{cx + 1} \sqrt{cx - 1} b + acx}{c}$$

input

```
int(a+b*acosh(c*x),x)
```

output

```
(acosh(c*x)*b*c*x - sqrt(c*x + 1)*sqrt(c*x - 1)*b + a*c*x)/c
```

### 3.5 $\int \frac{1}{a+b\operatorname{arccosh}(cx)} dx$

Optimal result	58
Mathematica [A] (verified)	58
Rubi [C] (verified)	59
Maple [A] (verified)	62
Fricas [F]	62
Sympy [F]	62
Maxima [F]	63
Giac [F]	63
Mupad [F(-1)]	63
Reduce [F]	64

#### Optimal result

Integrand size = 10, antiderivative size = 54

$$\int \frac{1}{a + b\operatorname{arccosh}(cx)} dx = -\frac{\operatorname{Chi}\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right) \sinh\left(\frac{a}{b}\right)}{bc} + \frac{\cosh\left(\frac{a}{b}\right) \operatorname{Shi}\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{bc}$$

output

```
-Chi((a+b*arccosh(c*x))/b)*sinh(a/b)/b/c+cosh(a/b)*Shi((a+b*arccosh(c*x))/b)/b/c
```

#### Mathematica [A] (verified)

Time = 0.05 (sec) , antiderivative size = 46, normalized size of antiderivative = 0.85

$$\int \frac{1}{a + b\operatorname{arccosh}(cx)} dx = -\frac{\operatorname{Chi}\left(\frac{a}{b} + \operatorname{arccosh}(cx)\right) \sinh\left(\frac{a}{b}\right) - \cosh\left(\frac{a}{b}\right) \operatorname{Shi}\left(\frac{a}{b} + \operatorname{arccosh}(cx)\right)}{bc}$$

input

```
Integrate[(a + b*ArcCosh[c*x])^(-1), x]
```

output

```
-((CoshIntegral[a/b + ArcCosh[c*x]]*Sinh[a/b] - Cosh[a/b]*SinhIntegral[a/b + ArcCosh[c*x]])/(b*c))
```

**Rubi [C] (verified)**

Result contains complex when optimal does not.

Time = 0.43 (sec) , antiderivative size = 57, normalized size of antiderivative = 1.06, number of steps used = 11, number of rules used = 10,  $\frac{\text{number of rules}}{\text{integrand size}} = 1.000$ , Rules used = {6296, 25, 3042, 26, 3784, 26, 3042, 26, 3779, 3782}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\begin{aligned}
 & \int \frac{1}{a + \operatorname{barccosh}(cx)} dx \\
 & \quad \downarrow \text{6296} \\
 & \frac{\int -\frac{\sinh\left(\frac{a}{b} - \frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{a+b\operatorname{arccosh}(cx)} d(a + \operatorname{barccosh}(cx))}{bc} \\
 & \quad \downarrow \text{25} \\
 & \frac{\int \frac{\sinh\left(\frac{a}{b} - \frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{a+b\operatorname{arccosh}(cx)} d(a + \operatorname{barccosh}(cx))}{bc} \\
 & \quad \downarrow \text{3042} \\
 & \frac{\int -\frac{i \sin\left(\frac{ia}{b} - \frac{i(a+b\operatorname{arccosh}(cx))}{b}\right)}{a+b\operatorname{arccosh}(cx)} d(a + \operatorname{barccosh}(cx))}{bc} \\
 & \quad \downarrow \text{26} \\
 & \frac{i \int \frac{\sin\left(\frac{ia}{b} - \frac{i(a+b\operatorname{arccosh}(cx))}{b}\right)}{a+b\operatorname{arccosh}(cx)} d(a + \operatorname{barccosh}(cx))}{bc} \\
 & \quad \downarrow \text{3784} \\
 & \frac{i \left( i \sinh\left(\frac{a}{b}\right) \int \frac{\cosh\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{a+b\operatorname{arccosh}(cx)} d(a + \operatorname{barccosh}(cx)) + \cosh\left(\frac{a}{b}\right) \int -\frac{i \sinh\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{a+b\operatorname{arccosh}(cx)} d(a + \operatorname{barccosh}(cx)) \right)}{bc} \\
 & \quad \downarrow \text{26}
 \end{aligned}$$

$$i \left( i \sinh \left( \frac{a}{b} \right) \int \frac{\cosh \left( \frac{a+b \operatorname{arccosh}(cx)}{b} \right)}{a+b \operatorname{arccosh}(cx)} d(a + \operatorname{arccosh}(cx)) - i \cosh \left( \frac{a}{b} \right) \int \frac{\sinh \left( \frac{a+b \operatorname{arccosh}(cx)}{b} \right)}{a+b \operatorname{arccosh}(cx)} d(a + \operatorname{arccosh}(cx)) \right)$$

$bc$

↓ 3042

$$i \left( i \sinh \left( \frac{a}{b} \right) \int \frac{\sin \left( \frac{i(a+b \operatorname{arccosh}(cx))}{b} + \frac{\pi}{2} \right)}{a+b \operatorname{arccosh}(cx)} d(a + \operatorname{arccosh}(cx)) - i \cosh \left( \frac{a}{b} \right) \int \frac{i \sin \left( \frac{i(a+b \operatorname{arccosh}(cx))}{b} \right)}{a+b \operatorname{arccosh}(cx)} d(a + \operatorname{arccosh}(cx)) \right)$$

$bc$

↓ 26

$$i \left( i \sinh \left( \frac{a}{b} \right) \int \frac{\sin \left( \frac{i(a+b \operatorname{arccosh}(cx))}{b} + \frac{\pi}{2} \right)}{a+b \operatorname{arccosh}(cx)} d(a + \operatorname{arccosh}(cx)) - \cosh \left( \frac{a}{b} \right) \int \frac{\sin \left( \frac{i(a+b \operatorname{arccosh}(cx))}{b} \right)}{a+b \operatorname{arccosh}(cx)} d(a + \operatorname{arccosh}(cx)) \right)$$

$bc$

↓ 3779

$$i \left( i \sinh \left( \frac{a}{b} \right) \int \frac{\sin \left( \frac{i(a+b \operatorname{arccosh}(cx))}{b} + \frac{\pi}{2} \right)}{a+b \operatorname{arccosh}(cx)} d(a + \operatorname{arccosh}(cx)) - i \cosh \left( \frac{a}{b} \right) \operatorname{Shi} \left( \frac{a+b \operatorname{arccosh}(cx)}{b} \right) \right)$$

$bc$

↓ 3782

$$i \left( i \sinh \left( \frac{a}{b} \right) \operatorname{Chi} \left( \frac{a+b \operatorname{arccosh}(cx)}{b} \right) - i \cosh \left( \frac{a}{b} \right) \operatorname{Shi} \left( \frac{a+b \operatorname{arccosh}(cx)}{b} \right) \right)$$

$bc$

input `Int[(a + b*ArcCosh[c*x])^(-1),x]`

output `(I*(I*CoshIntegral[(a + b*ArcCosh[c*x])/b]*Sinh[a/b] - I*Cosh[a/b]*SinhIntegral[(a + b*ArcCosh[c*x])/b]))/(b*c)`

## Definitions of rubi rules used

- rule 25  $\text{Int}[-(\text{Fx}_), \text{x\_Symbol}] \rightarrow \text{Simp}[\text{Identity}[-1] \text{ Int}[\text{Fx}, \text{x}], \text{x}]$
- rule 26  $\text{Int}[(\text{Complex}[0, \text{a}_])*(\text{Fx}_), \text{x\_Symbol}] \rightarrow \text{Simp}[(\text{Complex}[\text{Identity}[0], \text{a}]) \text{ Int}[\text{Fx}, \text{x}], \text{x}] /; \text{FreeQ}[\text{a}, \text{x}] \ \&\& \ \text{EqQ}[\text{a}^2, 1]$
- rule 3042  $\text{Int}[\text{u}_, \text{x\_Symbol}] \rightarrow \text{Int}[\text{DeactivateTrig}[\text{u}, \text{x}], \text{x}] /; \text{FunctionOfTrigOfLinearQ}[\text{u}, \text{x}]$
- rule 3779  $\text{Int}[\sin[(\text{e}_.) + (\text{Complex}[0, \text{fz}_])*(\text{f}_.)*(x_)]/((\text{c}_.) + (\text{d}_.)*(x_)), \text{x\_Symbol}] \rightarrow \text{Simp}[\text{I}*(\text{SinhIntegral}[\text{c}*f*(\text{fz}/\text{d}) + \text{f}*fz*x]/\text{d}), \text{x}] /; \text{FreeQ}[\{c, d, e, f, \text{fz}\}, \text{x}] \ \&\& \ \text{EqQ}[\text{d}*e - \text{c}*f*fz*\text{I}, 0]$
- rule 3782  $\text{Int}[\sin[(\text{e}_.) + (\text{Complex}[0, \text{fz}_])*(\text{f}_.)*(x_)]/((\text{c}_.) + (\text{d}_.)*(x_)), \text{x\_Symbol}] \rightarrow \text{Simp}[\text{CoshIntegral}[\text{c}*f*(\text{fz}/\text{d}) + \text{f}*fz*x]/\text{d}, \text{x}] /; \text{FreeQ}[\{c, d, e, f, \text{fz}\}, \text{x}] \ \&\& \ \text{EqQ}[\text{d}*(e - \text{Pi}/2) - \text{c}*f*fz*\text{I}, 0]$
- rule 3784  $\text{Int}[\sin[(\text{e}_.) + (\text{f}_.)*(x_)]/((\text{c}_.) + (\text{d}_.)*(x_)), \text{x\_Symbol}] \rightarrow \text{Simp}[\text{Cos}[(\text{d}*e - \text{c}*f)/\text{d}] \text{ Int}[\text{Sin}[\text{c}*(\text{f}/\text{d}) + \text{f}*x]/(\text{c} + \text{d}*x), \text{x}], \text{x}] + \text{Simp}[\text{Sin}[(\text{d}*e - \text{c}*f)/\text{d}] \text{ Int}[\text{Cos}[\text{c}*(\text{f}/\text{d}) + \text{f}*x]/(\text{c} + \text{d}*x), \text{x}], \text{x}] /; \text{FreeQ}[\{c, d, e, f\}, \text{x}] \ \&\& \ \text{NeQ}[\text{d}*e - \text{c}*f, 0]$
- rule 6296  $\text{Int}[(\text{a}_.) + \text{ArcCosh}[(\text{c}_.)*(x_)]*(\text{b}_.)^{\text{n}_}, \text{x\_Symbol}] \rightarrow \text{Simp}[1/(\text{b}*c) \text{ Subst}[\text{Int}[\text{x}^{\text{n}}*\text{Sinh}[-\text{a}/\text{b} + \text{x}/\text{b}], \text{x}], \text{x}, \text{a} + \text{b}*\text{ArcCosh}[\text{c}*x]], \text{x}] /; \text{FreeQ}[\{a, b, c, \text{n}\}, \text{x}]$

**Maple [A] (verified)**

Time = 0.07 (sec) , antiderivative size = 56, normalized size of antiderivative = 1.04

method	result	size
derivativedivides	$\frac{\frac{e^{\frac{a}{b}} \expIntegral_1(\operatorname{arccosh}(cx) + \frac{a}{b})}{2b} - \frac{e^{-\frac{a}{b}} \expIntegral_1(-\operatorname{arccosh}(cx) - \frac{a}{b})}{2b}}{c}$	56
default	$\frac{\frac{e^{\frac{a}{b}} \expIntegral_1(\operatorname{arccosh}(cx) + \frac{a}{b})}{2b} - \frac{e^{-\frac{a}{b}} \expIntegral_1(-\operatorname{arccosh}(cx) - \frac{a}{b})}{2b}}{c}$	56

input `int(1/(a+b*arccosh(c*x)),x,method=_RETURNVERBOSE)`

output `1/c*(1/2/b*exp(a/b)*Ei(1,arccosh(c*x)+a/b)-1/2/b*exp(-a/b)*Ei(1,-arccosh(c*x)-a/b))`

**Fricas [F]**

$$\int \frac{1}{a + b \operatorname{arccosh}(cx)} dx = \int \frac{1}{b \operatorname{arccosh}(cx) + a} dx$$

input `integrate(1/(a+b*arccosh(c*x)),x, algorithm="fricas")`

output `integral(1/(b*arccosh(c*x) + a), x)`

**Sympy [F]**

$$\int \frac{1}{a + b \operatorname{arccosh}(cx)} dx = \int \frac{1}{a + b \operatorname{acosh}(cx)} dx$$

input `integrate(1/(a+b*acosh(c*x)),x)`

output `Integral(1/(a + b*acosh(c*x)), x)`

**Maxima [F]**

$$\int \frac{1}{a + b \operatorname{arccosh}(cx)} dx = \int \frac{1}{b \operatorname{arcosh}(cx) + a} dx$$

input `integrate(1/(a+b*arccosh(c*x)),x, algorithm="maxima")`

output `integrate(1/(b*arccosh(c*x) + a), x)`

**Giac [F]**

$$\int \frac{1}{a + b \operatorname{arccosh}(cx)} dx = \int \frac{1}{b \operatorname{arcosh}(cx) + a} dx$$

input `integrate(1/(a+b*arccosh(c*x)),x, algorithm="giac")`

output `integrate(1/(b*arccosh(c*x) + a), x)`

**Mupad [F(-1)]**

Timed out.

$$\int \frac{1}{a + b \operatorname{arccosh}(cx)} dx = \int \frac{1}{a + b \operatorname{acosh}(cx)} dx$$

input `int(1/(a + b*acosh(c*x)),x)`

output `int(1/(a + b*acosh(c*x)), x)`



**Reduce [F]**

$$\int \frac{1}{a + b \operatorname{arccosh}(cx)} dx = \int \frac{1}{\operatorname{acosh}(cx) b + a} dx$$

input `int(1/(a+b*acosh(c*x)),x)`

output `int(1/(acosh(c*x)*b + a),x)`

### 3.6 $\int \frac{1}{(a+b\text{arccosh}(cx))^2} dx$

Optimal result	65
Mathematica [A] (warning: unable to verify)	65
Rubi [A] (verified)	66
Maple [A] (verified)	69
Fricas [F]	69
Sympy [F]	69
Maxima [F]	70
Giac [F]	70
Mupad [F(-1)]	71
Reduce [F]	71

#### Optimal result

Integrand size = 10, antiderivative size = 90

$$\int \frac{1}{(a + \text{barccosh}(cx))^2} dx = -\frac{\sqrt{-1 + cx}\sqrt{1 + cx}}{bc(a + \text{barccosh}(cx))} + \frac{\cosh\left(\frac{a}{b}\right) \text{Chi}\left(\frac{a + \text{barccosh}(cx)}{b}\right)}{b^2c} - \frac{\sinh\left(\frac{a}{b}\right) \text{Shi}\left(\frac{a + \text{barccosh}(cx)}{b}\right)}{b^2c}$$

output

$-(c*x-1)^{(1/2)}*(c*x+1)^{(1/2)}/b/c/(a+b*\text{arccosh}(c*x))+\cosh(a/b)*\text{Chi}((a+b*\text{arccosh}(c*x))/b)/b^2/c-\sinh(a/b)*\text{Shi}((a+b*\text{arccosh}(c*x))/b)/b^2/c$

#### Mathematica [A] (warning: unable to verify)

Time = 0.30 (sec) , antiderivative size = 80, normalized size of antiderivative = 0.89

$$\int \frac{1}{(a + \text{barccosh}(cx))^2} dx = \frac{-\frac{b\sqrt{\frac{-1+cx}{1+cx}}(1+cx)}{a+b\text{arccosh}(cx)} + \cosh\left(\frac{a}{b}\right) \text{Chi}\left(\frac{a}{b} + \text{arccosh}(cx)\right) - \sinh\left(\frac{a}{b}\right) \text{Shi}\left(\frac{a}{b} + \text{arccosh}(cx)\right)}{b^2c}$$

input

`Integrate[(a + b*ArcCosh[c*x])^(-2), x]`

output

$$\frac{(-((b\sqrt{-1+cx}/(1+cx))*(1+cx))/(a+b\text{ArcCosh}[cx])) + \text{Cosh}[a/b]*\text{CoshIntegral}[a/b + \text{ArcCosh}[cx]] - \text{Sinh}[a/b]*\text{SinhIntegral}[a/b + \text{ArcCosh}[cx]])/(b^2c)}$$
**Rubi [A] (verified)**

Time = 1.16 (sec) , antiderivative size = 86, normalized size of antiderivative = 0.96, number of steps used = 10, number of rules used = 9,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.900$ , Rules used = {6295, 6368, 3042, 3784, 26, 3042, 26, 3779, 3782}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\begin{aligned} & \int \frac{1}{(a + \text{barccosh}(cx))^2} dx \\ & \quad \downarrow 6295 \\ & \frac{c \int \frac{x}{\sqrt{cx-1}\sqrt{cx+1}(a+\text{barccosh}(cx))} dx}{b} - \frac{\sqrt{cx-1}\sqrt{cx+1}}{bc(a + \text{barccosh}(cx))} \\ & \quad \downarrow 6368 \\ & \frac{\int \frac{\cosh\left(\frac{a}{b} - \frac{a+\text{barccosh}(cx)}{b}\right)}{a+\text{barccosh}(cx)} d(a + \text{barccosh}(cx))}{b^2c} - \frac{\sqrt{cx-1}\sqrt{cx+1}}{bc(a + \text{barccosh}(cx))} \\ & \quad \downarrow 3042 \\ & -\frac{\sqrt{cx-1}\sqrt{cx+1}}{bc(a + \text{barccosh}(cx))} + \frac{\int \frac{\sin\left(\frac{ia}{b} - \frac{i(a+\text{barccosh}(cx))}{b} + \frac{\pi}{2}\right)}{a+\text{barccosh}(cx)} d(a + \text{barccosh}(cx))}{b^2c} \\ & \quad \downarrow 3784 \\ & -\frac{\sqrt{cx-1}\sqrt{cx+1}}{bc(a + \text{barccosh}(cx))} + \\ & \frac{\cosh\left(\frac{a}{b}\right) \int \frac{\cosh\left(\frac{a+\text{barccosh}(cx)}{b}\right)}{a+\text{barccosh}(cx)} d(a + \text{barccosh}(cx)) - i \sinh\left(\frac{a}{b}\right) \int \frac{i \sinh\left(\frac{a+\text{barccosh}(cx)}{b}\right)}{a+\text{barccosh}(cx)} d(a + \text{barccosh}(cx))}{b^2c} \\ & \quad \downarrow 26 \end{aligned}$$

$$\begin{aligned}
& \frac{\cosh\left(\frac{a}{b}\right) \int \frac{\cosh\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{a+b\operatorname{arccosh}(cx)} d(a+b\operatorname{arccosh}(cx)) - \sinh\left(\frac{a}{b}\right) \int \frac{\sinh\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{a+b\operatorname{arccosh}(cx)} d(a+b\operatorname{arccosh}(cx))}{b^2c} \\
& \quad - \frac{\sqrt{cx-1}\sqrt{cx+1}}{bc(a+b\operatorname{arccosh}(cx))} \\
& \quad \downarrow 3042 \\
& \quad - \frac{\sqrt{cx-1}\sqrt{cx+1}}{bc(a+b\operatorname{arccosh}(cx))} + \\
& \frac{\cosh\left(\frac{a}{b}\right) \int \frac{\sin\left(\frac{i(a+b\operatorname{arccosh}(cx))}{b} + \frac{\pi}{2}\right)}{a+b\operatorname{arccosh}(cx)} d(a+b\operatorname{arccosh}(cx)) - \sinh\left(\frac{a}{b}\right) \int \frac{i \sin\left(\frac{i(a+b\operatorname{arccosh}(cx))}{b}\right)}{a+b\operatorname{arccosh}(cx)} d(a+b\operatorname{arccosh}(cx))}{b^2c} \\
& \quad \downarrow 26 \\
& \quad - \frac{\sqrt{cx-1}\sqrt{cx+1}}{bc(a+b\operatorname{arccosh}(cx))} + \\
& \frac{i \sinh\left(\frac{a}{b}\right) \int \frac{\sin\left(\frac{i(a+b\operatorname{arccosh}(cx))}{b}\right)}{a+b\operatorname{arccosh}(cx)} d(a+b\operatorname{arccosh}(cx)) + \cosh\left(\frac{a}{b}\right) \int \frac{\sin\left(\frac{i(a+b\operatorname{arccosh}(cx))}{b} + \frac{\pi}{2}\right)}{a+b\operatorname{arccosh}(cx)} d(a+b\operatorname{arccosh}(cx))}{b^2c} \\
& \quad \downarrow 3779 \\
& \quad - \frac{\sqrt{cx-1}\sqrt{cx+1}}{bc(a+b\operatorname{arccosh}(cx))} + \\
& \quad - \sinh\left(\frac{a}{b}\right) \operatorname{Shi}\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right) + \cosh\left(\frac{a}{b}\right) \int \frac{\sin\left(\frac{i(a+b\operatorname{arccosh}(cx))}{b} + \frac{\pi}{2}\right)}{a+b\operatorname{arccosh}(cx)} d(a+b\operatorname{arccosh}(cx))}{b^2c} \\
& \quad \downarrow 3782 \\
& \frac{\cosh\left(\frac{a}{b}\right) \operatorname{Chi}\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right) - \sinh\left(\frac{a}{b}\right) \operatorname{Shi}\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{b^2c} - \frac{\sqrt{cx-1}\sqrt{cx+1}}{bc(a+b\operatorname{arccosh}(cx))}
\end{aligned}$$

input `Int[(a + b*ArcCosh[c*x])^(-2),x]`

output `-((Sqrt[-1 + c*x]*Sqrt[1 + c*x])/(b*c*(a + b*ArcCosh[c*x]))) + (Cosh[a/b]*CoshIntegral[(a + b*ArcCosh[c*x])/b] - Sinh[a/b]*SinhIntegral[(a + b*ArcCosh[c*x])/b])/(b^2*c)`

## Definitions of rubi rules used

- rule 26  $\text{Int}[(\text{Complex}[0, a_])*(F_{x_}), x\_Symbol] \rightarrow \text{Simp}[(\text{Complex}[\text{Identity}[0], a]) \text{Int}[F_{x_}, x], x] /; \text{FreeQ}[a, x] \ \&\& \ \text{EqQ}[a^2, 1]$
- rule 3042  $\text{Int}[u_, x\_Symbol] \rightarrow \text{Int}[\text{DeactivateTrig}[u, x], x] /; \text{FunctionOfTrigOfLinearQ}[u, x]$
- rule 3779  $\text{Int}[\sin[(e_.) + (\text{Complex}[0, fz_])*(f_.)*(x_)]/((c_.) + (d_.)*(x_)), x\_Symbol] \rightarrow \text{Simp}[I*(\text{SinhIntegral}[c*f*(fz/d) + f*fz*x]/d), x] /; \text{FreeQ}[\{c, d, e, f, fz\}, x] \ \&\& \ \text{EqQ}[d*e - c*f*fz*I, 0]$
- rule 3782  $\text{Int}[\sin[(e_.) + (\text{Complex}[0, fz_])*(f_.)*(x_)]/((c_.) + (d_.)*(x_)), x\_Symbol] \rightarrow \text{Simp}[\text{CoshIntegral}[c*f*(fz/d) + f*fz*x]/d, x] /; \text{FreeQ}[\{c, d, e, f, fz\}, x] \ \&\& \ \text{EqQ}[d*(e - \text{Pi}/2) - c*f*fz*I, 0]$
- rule 3784  $\text{Int}[\sin[(e_.) + (f_.)*(x_)]/((c_.) + (d_.)*(x_)), x\_Symbol] \rightarrow \text{Simp}[\text{Cos}[(d*e - c*f)/d] \text{Int}[\text{Sin}[c*(f/d) + f*x]/(c + d*x), x], x] + \text{Simp}[\text{Sin}[(d*e - c*f)/d] \text{Int}[\text{Cos}[c*(f/d) + f*x]/(c + d*x), x], x] /; \text{FreeQ}[\{c, d, e, f\}, x] \ \&\& \ \text{NeQ}[d*e - c*f, 0]$
- rule 6295  $\text{Int}[((a_.) + \text{ArcCosh}[(c_.)*(x_)]*(b_.) )^{(n_)}, x\_Symbol] \rightarrow \text{Simp}[\text{Sqrt}[1 + c*x]*\text{Sqrt}[-1 + c*x]*((a + b*\text{ArcCosh}[c*x])^{(n+1)})/(b*c*(n+1)), x] - \text{Simp}[c/(b*(n+1)) \text{Int}[x*((a + b*\text{ArcCosh}[c*x])^{(n+1)})/(\text{Sqrt}[1 + c*x]*\text{Sqrt}[-1 + c*x]), x], x] /; \text{FreeQ}[\{a, b, c\}, x] \ \&\& \ \text{LtQ}[n, -1]$
- rule 6368  $\text{Int}[((a_.) + \text{ArcCosh}[(c_.)*(x_)]*(b_.) )^{(n_.)}*(x_)^{(m_.)}*((d1_.) + (e1_.)*(x_))^{(p_.)}*((d2_.) + (e2_.)*(x_))^{(p_.)}, x\_Symbol] \rightarrow \text{Simp}[(1/(b*c^{(m+1)}))*\text{Simp}[(d1 + e1*x)^p/(1 + c*x)^p]*\text{Simp}[(d2 + e2*x)^p/(-1 + c*x)^p] \text{Subst}[\text{Int}[x^n*\text{Cosh}[-a/b + x/b]^m*\text{Sinh}[-a/b + x/b]^{(2*p+1)}, x], x, a + b*\text{ArcCosh}[c*x]], x] /; \text{FreeQ}[\{a, b, c, d1, e1, d2, e2, n\}, x] \ \&\& \ \text{EqQ}[e1, c*d1] \ \&\& \ \text{EqQ}[e2, (-c)*d2] \ \&\& \ \text{IGtQ}[p + 3/2, 0] \ \&\& \ \text{IGtQ}[m, 0]$

**Maple [A] (verified)**

Time = 0.06 (sec) , antiderivative size = 125, normalized size of antiderivative = 1.39

method	result	size
derivativedivides	$\frac{-\sqrt{cx-1}\sqrt{cx+1}+cx}{2b(a+b \operatorname{arccosh}(cx))} - \frac{e^{\frac{a}{b}} \operatorname{expIntegral}_1\left(\operatorname{arccosh}(cx)+\frac{a}{b}\right)}{2b^2} - \frac{cx+\sqrt{cx-1}\sqrt{cx+1}}{2b(a+b \operatorname{arccosh}(cx))} - \frac{e^{-\frac{a}{b}} \operatorname{expIntegral}_1\left(-\operatorname{arccosh}(cx)-\frac{a}{b}\right)}{2b^2}$	125
default	$\frac{-\sqrt{cx-1}\sqrt{cx+1}+cx}{2b(a+b \operatorname{arccosh}(cx))} - \frac{e^{\frac{a}{b}} \operatorname{expIntegral}_1\left(\operatorname{arccosh}(cx)+\frac{a}{b}\right)}{2b^2} - \frac{cx+\sqrt{cx-1}\sqrt{cx+1}}{2b(a+b \operatorname{arccosh}(cx))} - \frac{e^{-\frac{a}{b}} \operatorname{expIntegral}_1\left(-\operatorname{arccosh}(cx)-\frac{a}{b}\right)}{2b^2}$	125

input `int(1/(a+b*arccosh(c*x))^2,x,method=_RETURNVERBOSE)`

output `1/c*(1/2*(-(c*x-1)^(1/2)*(c*x+1)^(1/2)+c*x)/b/(a+b*arccosh(c*x))-1/2/b^2*exp(a/b)*Ei(1,arccosh(c*x)+a/b)-1/2/b*(c*x+(c*x-1)^(1/2)*(c*x+1)^(1/2))/(a+b*arccosh(c*x))-1/2/b^2*exp(-a/b)*Ei(1,-arccosh(c*x)-a/b)`

**Fricas [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^2} dx = \int \frac{1}{(b \operatorname{arccosh}(cx) + a)^2} dx$$

input `integrate(1/(a+b*arccosh(c*x))^2,x, algorithm="fricas")`

output `integral(1/(b^2*arccosh(c*x)^2 + 2*a*b*arccosh(c*x) + a^2), x)`

**Sympy [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^2} dx = \int \frac{1}{(a + b \operatorname{acosh}(cx))^2} dx$$

input `integrate(1/(a+b*acosh(c*x))**2,x)`

output `Integral((a + b*acosh(c*x))**(-2), x)`

### Maxima [F]

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^2} dx = \int \frac{1}{(b \operatorname{arcosh}(cx) + a)^2} dx$$

input `integrate(1/(a+b*arccosh(c*x))^2,x, algorithm="maxima")`

output `-(c^3*x^3 + (c^2*x^2 - 1)*sqrt(c*x + 1)*sqrt(c*x - 1) - c*x)/(a*b*c^3*x^2 + sqrt(c*x + 1)*sqrt(c*x - 1)*a*b*c^2*x - a*b*c + (b^2*c^3*x^2 + sqrt(c*x + 1)*sqrt(c*x - 1)*b^2*c^2*x - b^2*c)*log(c*x + sqrt(c*x + 1)*sqrt(c*x - 1))) + integrate((c^4*x^4 - 2*c^2*x^2 + (c^2*x^2 + 1)*(c*x + 1)*(c*x - 1) + (2*c^3*x^3 - c*x)*sqrt(c*x + 1)*sqrt(c*x - 1) + 1)/(a*b*c^4*x^4 + (c*x + 1)*(c*x - 1)*a*b*c^2*x^2 - 2*a*b*c^2*x^2 + 2*(a*b*c^3*x^3 - a*b*c*x)*sqrt(c*x + 1)*sqrt(c*x - 1) + a*b + (b^2*c^4*x^4 + (c*x + 1)*(c*x - 1)*b^2*c^2*x^2 - 2*b^2*c^2*x^2 + 2*(b^2*c^3*x^3 - b^2*c*x)*sqrt(c*x + 1)*sqrt(c*x - 1)) + b^2)*log(c*x + sqrt(c*x + 1)*sqrt(c*x - 1))), x)`

### Giac [F]

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^2} dx = \int \frac{1}{(b \operatorname{arcosh}(cx) + a)^2} dx$$

input `integrate(1/(a+b*arccosh(c*x))^2,x, algorithm="giac")`

output `integrate((b*arccosh(c*x) + a)^(-2), x)`

**Mupad [F(-1)]**

Timed out.

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^2} dx = \int \frac{1}{(a + b \operatorname{acosh}(cx))^2} dx$$

input `int(1/(a + b*acosh(c*x))^2,x)`output `int(1/(a + b*acosh(c*x))^2, x)`**Reduce [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^2} dx = \int \frac{1}{\operatorname{acosh}(cx)^2 b^2 + 2 \operatorname{acosh}(cx) ab + a^2} dx$$

input `int(1/(a+b*acosh(c*x))^2,x)`output `int(1/(acosh(c*x)**2*b**2 + 2*acosh(c*x)*a*b + a**2),x)`



### 3.7 $\int \frac{1}{(a+b\operatorname{arccosh}(cx))^3} dx$

Optimal result	72
Mathematica [A] (verified)	73
Rubi [C] (verified)	73
Maple [A] (verified)	77
Fricas [F]	78
Sympy [F]	78
Maxima [F]	79
Giac [F]	79
Mupad [F(-1)]	80
Reduce [F]	80

#### Optimal result

Integrand size = 10, antiderivative size = 115

$$\int \frac{1}{(a+b\operatorname{arccosh}(cx))^3} dx = -\frac{\sqrt{-1+cx}\sqrt{1+cx}}{2bc(a+b\operatorname{arccosh}(cx))^2} - \frac{x}{2b^2(a+b\operatorname{arccosh}(cx))} - \frac{\operatorname{Chi}\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right) \sinh\left(\frac{a}{b}\right)}{2b^3c} + \frac{\cosh\left(\frac{a}{b}\right) \operatorname{Shi}\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{2b^3c}$$

output

```
-1/2*(c*x-1)^(1/2)*(c*x+1)^(1/2)/b/c/(a+b*arccosh(c*x))^2-1/2*x/b^2/(a+b*arccosh(c*x))-1/2*Chi((a+b*arccosh(c*x))/b)*sinh(a/b)/b^3/c+1/2*cosh(a/b)*Shi((a+b*arccosh(c*x))/b)/b^3/c
```

**Mathematica [A] (verified)**

Time = 0.25 (sec) , antiderivative size = 93, normalized size of antiderivative = 0.81

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^3} dx = \frac{\frac{b(acx + b\sqrt{-1+cx}\sqrt{1+cx} + bcx \operatorname{arccosh}(cx))}{(a + b \operatorname{arccosh}(cx))^2} + \operatorname{Chi}\left(\frac{a}{b} + \operatorname{arccosh}(cx)\right) \sinh\left(\frac{a}{b}\right) - \cosh\left(\frac{a}{b}\right) \operatorname{Shi}\left(\frac{a}{b} + \operatorname{arccosh}(cx)\right)}{2b^3c}$$

input

```
Integrate[(a + b*ArcCosh[c*x])^(-3), x]
```

output

```
-1/2*((b*(a*c*x + b*Sqrt[-1 + c*x]*Sqrt[1 + c*x] + b*c*x*ArcCosh[c*x]))/(a + b*ArcCosh[c*x])^2 + CoshIntegral[a/b + ArcCosh[c*x]]*Sinh[a/b] - Cosh[a/b]*SinhIntegral[a/b + ArcCosh[c*x]])/(b^3*c)
```

**Rubi [C] (verified)**

Result contains complex when optimal does not.

Time = 1.53 (sec) , antiderivative size = 124, normalized size of antiderivative = 1.08, number of steps used = 13, number of rules used = 12,  $\frac{\text{number of rules}}{\text{integrand size}} = 1.200$ , Rules used = {6295, 6366, 6296, 25, 3042, 26, 3784, 26, 3042, 26, 3779, 3782}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\begin{aligned} & \int \frac{1}{(a + b \operatorname{arccosh}(cx))^3} dx \\ & \quad \downarrow \text{6295} \\ & \frac{c \int \frac{x}{\sqrt{cx-1}\sqrt{cx+1}(a+b \operatorname{arccosh}(cx))^2} dx}{2b} - \frac{\sqrt{cx-1}\sqrt{cx+1}}{2bc(a + b \operatorname{arccosh}(cx))^2} \\ & \quad \downarrow \text{6366} \\ & \frac{c \left( \frac{\int \frac{1}{a+b \operatorname{arccosh}(cx)} dx}{bc} - \frac{x}{bc(a+b \operatorname{arccosh}(cx))} \right)}{2b} - \frac{\sqrt{cx-1}\sqrt{cx+1}}{2bc(a + b \operatorname{arccosh}(cx))^2} \end{aligned}$$

$$\begin{array}{c}
 \downarrow \text{6296} \\
 c \left( \frac{\int -\frac{\sinh\left(\frac{a}{b} - \frac{a+b\operatorname{arccosh}(cx)}{b}\right) d(a+b\operatorname{arccosh}(cx))}{a+b\operatorname{arccosh}(cx)} - \frac{x}{bc(a+b\operatorname{arccosh}(cx))}}{b^2 c^2} \right) \\
 \hline
 2b \qquad \qquad \qquad - \frac{\sqrt{cx-1}\sqrt{cx+1}}{2bc(a+b\operatorname{arccosh}(cx))^2}
 \end{array}$$

$$\begin{array}{c}
 \downarrow \text{25} \\
 c \left( -\frac{\int \frac{\sinh\left(\frac{a}{b} - \frac{a+b\operatorname{arccosh}(cx)}{b}\right) d(a+b\operatorname{arccosh}(cx))}{a+b\operatorname{arccosh}(cx)} - \frac{x}{bc(a+b\operatorname{arccosh}(cx))}}{b^2 c^2} \right) \\
 \hline
 2b \qquad \qquad \qquad - \frac{\sqrt{cx-1}\sqrt{cx+1}}{2bc(a+b\operatorname{arccosh}(cx))^2}
 \end{array}$$

$$\begin{array}{c}
 \downarrow \text{3042} \\
 -\frac{\sqrt{cx-1}\sqrt{cx+1}}{2bc(a+b\operatorname{arccosh}(cx))^2} + \\
 c \left( -\frac{x}{bc(a+b\operatorname{arccosh}(cx))} - \frac{\int -\frac{i \sin\left(\frac{ia}{b} - \frac{i(a+b\operatorname{arccosh}(cx))}{b}\right) d(a+b\operatorname{arccosh}(cx))}{a+b\operatorname{arccosh}(cx)} - \frac{x}{bc(a+b\operatorname{arccosh}(cx))}}{b^2 c^2} \right) \\
 \hline
 2b
 \end{array}$$

$$\begin{array}{c}
 \downarrow \text{26} \\
 -\frac{\sqrt{cx-1}\sqrt{cx+1}}{2bc(a+b\operatorname{arccosh}(cx))^2} + \\
 c \left( -\frac{x}{bc(a+b\operatorname{arccosh}(cx))} + \frac{\int \frac{i \sin\left(\frac{ia}{b} - \frac{i(a+b\operatorname{arccosh}(cx))}{b}\right) d(a+b\operatorname{arccosh}(cx))}{a+b\operatorname{arccosh}(cx)} - \frac{x}{bc(a+b\operatorname{arccosh}(cx))}}{b^2 c^2} \right) \\
 \hline
 2b
 \end{array}$$

$$\begin{array}{c}
 \downarrow \text{3784} \\
 -\frac{\sqrt{cx-1}\sqrt{cx+1}}{2bc(a+b\operatorname{arccosh}(cx))^2} + \\
 c \left( -\frac{x}{bc(a+b\operatorname{arccosh}(cx))} + \frac{i \left( i \sinh\left(\frac{a}{b}\right) \int \frac{\cosh\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right) d(a+b\operatorname{arccosh}(cx)) + \cosh\left(\frac{a}{b}\right) \int -\frac{i \sinh\left(\frac{a+b\operatorname{arccosh}(cx)}{b}\right) d(a+b\operatorname{arccosh}(cx))}{a+b\operatorname{arccosh}(cx)} \right)}{b^2 c^2} \right) \\
 \hline
 2b
 \end{array}$$

$$\begin{array}{c}
 \downarrow 26 \\
 -\frac{\sqrt{cx-1}\sqrt{cx+1}}{2bc(a+\operatorname{barccosh}(cx))^2} + \\
 c \left( -\frac{x}{bc(a+\operatorname{barccosh}(cx))} + \frac{i \left( i \sinh\left(\frac{a}{b}\right) \int \frac{\cosh\left(\frac{a+\operatorname{barccosh}(cx)}{b}\right)}{a+\operatorname{barccosh}(cx)} d(a+\operatorname{barccosh}(cx)) - i \cosh\left(\frac{a}{b}\right) \int \frac{\sinh\left(\frac{a+\operatorname{barccosh}(cx)}{b}\right)}{a+\operatorname{barccosh}(cx)} d(a+\operatorname{barccosh}(cx)) \right)}{b^2 c^2} \right)
 \end{array}$$

2b

$$\begin{array}{c}
 \downarrow 3042 \\
 -\frac{\sqrt{cx-1}\sqrt{cx+1}}{2bc(a+\operatorname{barccosh}(cx))^2} + \\
 c \left( -\frac{x}{bc(a+\operatorname{barccosh}(cx))} + \frac{i \left( i \sinh\left(\frac{a}{b}\right) \int \frac{\sin\left(\frac{i(a+\operatorname{barccosh}(cx))}{b} + \frac{\pi}{2}\right)}{a+\operatorname{barccosh}(cx)} d(a+\operatorname{barccosh}(cx)) - i \cosh\left(\frac{a}{b}\right) \int \frac{i \sin\left(\frac{i(a+\operatorname{barccosh}(cx))}{b}\right)}{a+\operatorname{barccosh}(cx)} d(a+\operatorname{barccosh}(cx)) \right)}{b^2 c^2} \right)
 \end{array}$$

2b

$$\begin{array}{c}
 \downarrow 26 \\
 -\frac{\sqrt{cx-1}\sqrt{cx+1}}{2bc(a+\operatorname{barccosh}(cx))^2} + \\
 c \left( -\frac{x}{bc(a+\operatorname{barccosh}(cx))} + \frac{i \left( i \sinh\left(\frac{a}{b}\right) \int \frac{\sin\left(\frac{i(a+\operatorname{barccosh}(cx))}{b} + \frac{\pi}{2}\right)}{a+\operatorname{barccosh}(cx)} d(a+\operatorname{barccosh}(cx)) - \cosh\left(\frac{a}{b}\right) \int \frac{\sin\left(\frac{i(a+\operatorname{barccosh}(cx))}{b}\right)}{a+\operatorname{barccosh}(cx)} d(a+\operatorname{barccosh}(cx)) \right)}{b^2 c^2} \right)
 \end{array}$$

2b

$$\begin{array}{c}
 \downarrow 3779 \\
 -\frac{\sqrt{cx-1}\sqrt{cx+1}}{2bc(a+\operatorname{barccosh}(cx))^2} + \\
 c \left( -\frac{x}{bc(a+\operatorname{barccosh}(cx))} + \frac{i \left( i \sinh\left(\frac{a}{b}\right) \int \frac{\sin\left(\frac{i(a+\operatorname{barccosh}(cx))}{b} + \frac{\pi}{2}\right)}{a+\operatorname{barccosh}(cx)} d(a+\operatorname{barccosh}(cx)) - i \cosh\left(\frac{a}{b}\right) \operatorname{Shi}\left(\frac{a+\operatorname{barccosh}(cx)}{b}\right) \right)}{b^2 c^2} \right)
 \end{array}$$

2b

$$\begin{array}{c}
 \downarrow \text{3782} \\
 -\frac{\sqrt{cx-1}\sqrt{cx+1}}{2bc(a+\operatorname{arccosh}(cx))^2} + \\
 c \left( -\frac{x}{bc(a+\operatorname{arccosh}(cx))} + \frac{i \left( i \sinh\left(\frac{a}{b}\right) \operatorname{Chi}\left(\frac{a+\operatorname{arccosh}(cx)}{b}\right) - i \cosh\left(\frac{a}{b}\right) \operatorname{Shi}\left(\frac{a+\operatorname{arccosh}(cx)}{b}\right) \right)}{b^2 c^2} \right) \\
 \hline
 2b
 \end{array}$$

input `Int[(a + b*ArcCosh[c*x])^(-3),x]`

output `-1/2*(Sqrt[-1 + c*x]*Sqrt[1 + c*x])/(b*c*(a + b*ArcCosh[c*x])^2) + (c*(-x/(b*c*(a + b*ArcCosh[c*x]))) + (I*(I*CoshIntegral[(a + b*ArcCosh[c*x])/b]*Sinh[a/b] - I*Cosh[a/b]*SinhIntegral[(a + b*ArcCosh[c*x])/b]))/(b^2*c^2)))/(2*b)`

### Defintions of rubi rules used

rule 25 `Int[-(Fx_), x_Symbol] := Simp[Identity[-1] Int[Fx, x], x]`

rule 26 `Int[(Complex[0, a_]*(Fx_), x_Symbol] := Simp[(Complex[Identity[0], a]) Int[Fx, x], x] /; FreeQ[a, x] && EqQ[a^2, 1]`

rule 3042 `Int[u_, x_Symbol] := Int[DeactivateTrig[u, x], x] /; FunctionOfTrigOfLinearQ[u, x]`

rule 3779 `Int[sin[(e_.) + (Complex[0, fz_])*(f_.)*(x_)]/((c_.) + (d_.)*(x_)), x_Symbol] := Simp[I*(SinhIntegral[c*f*(fz/d) + f*fz*x]/d), x] /; FreeQ[{c, d, e, f, fz}, x] && EqQ[d*e - c*f*fz*I, 0]`

rule 3782 `Int[sin[(e_.) + (Complex[0, fz_])*(f_.)*(x_)]/((c_.) + (d_.)*(x_)), x_Symbol] := Simp[CoshIntegral[c*f*(fz/d) + f*fz*x]/d, x] /; FreeQ[{c, d, e, f, fz}, x] && EqQ[d*(e - Pi/2) - c*f*fz*I, 0]`

```
rule 3784 Int[sin[(e_.) + (f_.)*(x_)]/((c_.) + (d_.)*(x_)), x_Symbol] := Simp[Cos[(d*
e - c*f)/d] Int[Sin[c*(f/d) + f*x]/(c + d*x), x], x] + Simp[Sin[(d*e - c*
f)/d] Int[Cos[c*(f/d) + f*x]/(c + d*x), x], x] /; FreeQ[{c, d, e, f}, x]
&& NeQ[d*e - c*f, 0]
```

```
rule 6295 Int[((a_.) + ArcCosh[(c_.)*(x_)]*(b_.))^n, x_Symbol] := Simp[Sqrt[1 + c*
x]*Sqrt[-1 + c*x]*((a + b*ArcCosh[c*x])^(n + 1)/(b*c*(n + 1))), x] - Simp[c
/(b*(n + 1)) Int[x*((a + b*ArcCosh[c*x])^(n + 1)/(Sqrt[1 + c*x]*Sqrt[-1 +
c*x])), x], x] /; FreeQ[{a, b, c}, x] && LtQ[n, -1]
```

```
rule 6296 Int[((a_.) + ArcCosh[(c_.)*(x_)]*(b_.))^n, x_Symbol] := Simp[1/(b*c) S
ubst[Int[x^n*Sinh[-a/b + x/b], x], x, a + b*ArcCosh[c*x]], x] /; FreeQ[{a,
b, c, n}, x]
```

```
rule 6366 Int[(((a_.) + ArcCosh[(c_.)*(x_)]*(b_.))^n)*((f_.)*(x_))^(m_.)/(Sqrt[(d1
_) + (e1_.)*(x_)]*Sqrt[(d2_) + (e2_.)*(x_)]), x_Symbol] := Simp[(f*x)^m*((a
+ b*ArcCosh[c*x])^(n + 1)/(b*c*(n + 1)))*Simp[Sqrt[1 + c*x]/Sqrt[d1 + e1*x
]]*Simp[Sqrt[-1 + c*x]/Sqrt[d2 + e2*x]], x] - Simp[f*(m/(b*c*(n + 1)))*Simp
[Sqrt[1 + c*x]/Sqrt[d1 + e1*x]]*Simp[Sqrt[-1 + c*x]/Sqrt[d2 + e2*x]] Int[
(f*x)^(m - 1)*(a + b*ArcCosh[c*x])^(n + 1), x], x] /; FreeQ[{a, b, c, d1, e
1, d2, e2, f, m}, x] && EqQ[e1, c*d1] && EqQ[e2, (-c)*d2] && LtQ[n, -1]
```

Maple [A] (verified)

Time = 0.07 (sec) , antiderivative size = 184, normalized size of antiderivative = 1.60

method	result
derivativedivides	$\frac{-\frac{(-\sqrt{cx-1}\sqrt{cx+1}+cx)(b \operatorname{arccosh}(cx)+a-b)}{4b^2(\operatorname{arccosh}(cx)^2b^2+2 \operatorname{arccosh}(cx)ab+a^2)} + \frac{e^{\frac{a}{b}} \operatorname{expIntegral}_1(\operatorname{arccosh}(cx)+\frac{a}{b})}{4b^3} - \frac{cx+\sqrt{cx-1}\sqrt{cx+1}}{4b(a+b \operatorname{arccosh}(cx))^2} - \frac{cx+\sqrt{cx-1}\sqrt{cx+1}}{4b^2(a+b \operatorname{arccosh}(cx))}}{c}$
default	$\frac{-\frac{(-\sqrt{cx-1}\sqrt{cx+1}+cx)(b \operatorname{arccosh}(cx)+a-b)}{4b^2(\operatorname{arccosh}(cx)^2b^2+2 \operatorname{arccosh}(cx)ab+a^2)} + \frac{e^{\frac{a}{b}} \operatorname{expIntegral}_1(\operatorname{arccosh}(cx)+\frac{a}{b})}{4b^3} - \frac{cx+\sqrt{cx-1}\sqrt{cx+1}}{4b(a+b \operatorname{arccosh}(cx))^2} - \frac{cx+\sqrt{cx-1}\sqrt{cx+1}}{4b^2(a+b \operatorname{arccosh}(cx))}}{c}$

```
input int(1/(a+b*arccosh(c*x))^3,x,method=_RETURNVERBOSE)
```

output

```
1/c*(-1/4*(-(c*x-1)^(1/2)*(c*x+1)^(1/2)+c*x)*(b*arccosh(c*x)+a-b)/b^2/(arc
cosh(c*x)^2*b^2+2*arccosh(c*x)*a*b+a^2)+1/4/b^3*exp(a/b)*Ei(1,arccosh(c*x)
+a/b)-1/4/b*(c*x+(c*x-1)^(1/2)*(c*x+1)^(1/2))/(a+b*arccosh(c*x))^2-1/4/b^2
*(c*x+(c*x-1)^(1/2)*(c*x+1)^(1/2))/(a+b*arccosh(c*x))-1/4/b^3*exp(-a/b)*Ei
(1,-arccosh(c*x)-a/b))
```

**Fricas [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^3} dx = \int \frac{1}{(b \operatorname{arcosh}(cx) + a)^3} dx$$

input

```
integrate(1/(a+b*arccosh(c*x))^3,x, algorithm="fricas")
```

output

```
integral(1/(b^3*arccosh(c*x)^3 + 3*a*b^2*arccosh(c*x)^2 + 3*a^2*b*arccosh(
c*x) + a^3), x)
```

**Sympy [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^3} dx = \int \frac{1}{(a + b \operatorname{acosh}(cx))^3} dx$$

input

```
integrate(1/(a+b*acosh(c*x))**3,x)
```

output

```
Integral((a + b*acosh(c*x))**(-3), x)
```

**Maxima [F]**

$$\int \frac{1}{(a + \operatorname{arccosh}(cx))^3} dx = \int \frac{1}{(b \operatorname{arccosh}(cx) + a)^3} dx$$

input `integrate(1/(a+b*arccosh(c*x))^3,x, algorithm="maxima")`

output

```
-1/2*((a*c^7 + b*c^7)*x^7 - 3*(a*c^5 + b*c^5)*x^5 - (b*c^2*x^2 - (a*c^4 +
b*c^4)*x^4 + a)*(c*x + 1)^(3/2)*(c*x - 1)^(3/2) + 3*(a*c^3 + b*c^3)*x^3 +
(3*(a*c^5 + b*c^5)*x^5 - (3*a*c^3 + 5*b*c^3)*x^3 + 2*b*c*x)*(c*x + 1)*(c*x
- 1) + (3*(a*c^6 + b*c^6)*x^6 - (6*a*c^4 + 7*b*c^4)*x^4 + (4*a*c^2 + 5*b*
c^2)*x^2 - a - b)*sqrt(c*x + 1)*sqrt(c*x - 1) - (a*c + b*c)*x + (b*c^7*x^7
- 3*b*c^5*x^5 + 3*b*c^3*x^3 + (b*c^4*x^4 - b)*(c*x + 1)^(3/2)*(c*x - 1)^(
3/2) + 3*(b*c^5*x^5 - b*c^3*x^3)*(c*x + 1)*(c*x - 1) - b*c*x + (3*b*c^6*x^
6 - 6*b*c^4*x^4 + 4*b*c^2*x^2 - b)*sqrt(c*x + 1)*sqrt(c*x - 1))*log(c*x +
sqrt(c*x + 1)*sqrt(c*x - 1)))/(a^2*b^2*c^7*x^6 + (c*x + 1)^(3/2)*(c*x - 1)
^(3/2)*a^2*b^2*c^4*x^3 - 3*a^2*b^2*c^5*x^4 + 3*a^2*b^2*c^3*x^2 - a^2*b^2*c
+ 3*(a^2*b^2*c^5*x^4 - a^2*b^2*c^3*x^2)*(c*x + 1)*(c*x - 1) + (b^4*c^7*x^
6 + (c*x + 1)^(3/2)*(c*x - 1)^(3/2)*b^4*c^4*x^3 - 3*b^4*c^5*x^4 + 3*b^4*c^
3*x^2 - b^4*c + 3*(b^4*c^5*x^4 - b^4*c^3*x^2)*(c*x + 1)*(c*x - 1) + 3*(b^4
*c^6*x^5 - 2*b^4*c^4*x^3 + b^4*c^2*x)*sqrt(c*x + 1)*sqrt(c*x - 1))*log(c*x
+ sqrt(c*x + 1)*sqrt(c*x - 1))^2 + 3*(a^2*b^2*c^6*x^5 - 2*a^2*b^2*c^4*x^3
+ a^2*b^2*c^2*x)*sqrt(c*x + 1)*sqrt(c*x - 1) + 2*(a*b^3*c^7*x^6 + (c*x +
1)^(3/2)*(c*x - 1)^(3/2)*a*b^3*c^4*x^3 - 3*a*b^3*c^5*x^4 + 3*a*b^3*c^3*x^2
- a*b^3*c + 3*(a*b^3*c^5*x^4 - a*b^3*c^3*x^2)*(c*x + 1)*(c*x - 1) + 3*(a*
b^3*c^6*x^5 - 2*a*b^3*c^4*x^3 + a*b^3*c^2*x)*sqrt(c*x + 1)*sqrt(c*x - 1))*
log(c*x + sqrt(c*x + 1)*sqrt(c*x - 1))) + integrate(1/2*(c^8*x^8 - 4*c^...
```

**Giac [F]**

$$\int \frac{1}{(a + \operatorname{arccosh}(cx))^3} dx = \int \frac{1}{(b \operatorname{arccosh}(cx) + a)^3} dx$$

input `integrate(1/(a+b*arccosh(c*x))^3,x, algorithm="giac")`



output `integrate((b*arccosh(c*x) + a)^(-3), x)`

### Mupad [F(-1)]

Timed out.

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^3} dx = \int \frac{1}{(a + b \operatorname{acosh}(cx))^3} dx$$

input `int(1/(a + b*acosh(c*x))^3,x)`

output `int(1/(a + b*acosh(c*x))^3, x)`

### Reduce [F]

$$\begin{aligned} & \int \frac{1}{(a + b \operatorname{arccosh}(cx))^3} dx \\ &= \int \frac{1}{\operatorname{acosh}(cx)^3 b^3 + 3 \operatorname{acosh}(cx)^2 a b^2 + 3 \operatorname{acosh}(cx) a^2 b + a^3} dx \end{aligned}$$

input `int(1/(a+b*acosh(c*x))^3,x)`

output `int(1/(acosh(c*x)**3*b**3 + 3*acosh(c*x)**2*a*b**2 + 3*acosh(c*x)*a**2*b + a**3),x)`

### 3.8 $\int (a + b \operatorname{arccosh}(cx))^{5/2} dx$

Optimal result	81
Mathematica [B] (warning: unable to verify)	82
Rubi [A] (verified)	82
Maple [F]	86
Fricas [F(-2)]	87
Sympy [F(-1)]	87
Maxima [F]	87
Giac [F(-2)]	88
Mupad [F(-1)]	88
Reduce [F]	88

#### Optimal result

Integrand size = 12, antiderivative size = 160

$$\int (a + b \operatorname{arccosh}(cx))^{5/2} dx = \frac{15}{4} b^2 x \sqrt{a + b \operatorname{arccosh}(cx)} - \frac{5b\sqrt{-1 + cx}\sqrt{1 + cx}(a + b \operatorname{arccosh}(cx))^{3/2}}{2c} + x(a + b \operatorname{arccosh}(cx))^{5/2} - \frac{15b^{5/2} e^{a/b} \sqrt{\pi} \operatorname{erf}\left(\frac{\sqrt{a + b \operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{16c} - \frac{15b^{5/2} e^{-\frac{a}{b}} \sqrt{\pi} \operatorname{erfi}\left(\frac{\sqrt{a + b \operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{16c}$$

output

```
15/4*b^2*x*(a+b*arccosh(c*x))^(1/2)-5/2*b*(c*x-1)^(1/2)*(c*x+1)^(1/2)*(a+b
*arccosh(c*x))^(3/2)/c+x*(a+b*arccosh(c*x))^(5/2)-15/16*b^(5/2)*exp(a/b)*P
i^(1/2)*erf((a+b*arccosh(c*x))^(1/2)/b^(1/2))/c-15/16*b^(5/2)*Pi^(1/2)*erf
i((a+b*arccosh(c*x))^(1/2)/b^(1/2))/c/exp(a/b)
```

**Mathematica [B] (warning: unable to verify)**

Leaf count is larger than twice the leaf count of optimal. 452 vs.  $2(160) = 320$ .

Time = 1.56 (sec) , antiderivative size = 452, normalized size of antiderivative = 2.82

$$\int (a$$

$$+ b \sqrt{a + b \operatorname{arccosh}(cx)} \left( 2 \sqrt{\frac{-1+cx}{1+cx}} (1+cx)(a - 5b \operatorname{arccosh}(cx)) + bcx(15 + 4 \operatorname{arccosh}(cx)) \right)^{5/2} dx =$$

input `Integrate[(a + b*ArcCosh[c*x])^(5/2), x]`

output

```
(4*b*Sqrt[a + b*ArcCosh[c*x]]*(2*Sqrt[(-1 + c*x)/(1 + c*x)]*(1 + c*x)*(a -
5*b*ArcCosh[c*x]) + b*c*x*(15 + 4*ArcCosh[c*x]^2)) + (8*a^2*Sqrt[a + b*Ar
cCosh[c*x]]*((E^((2*a)/b)*Gamma[3/2, a/b + ArcCosh[c*x]])/Sqrt[a/b + ArcCo
sh[c*x]] + Gamma[3/2, -(a + b*ArcCosh[c*x])/b])/Sqrt[-(a + b*ArcCosh[c*x
])/b]))/E^(a/b) - Sqrt[b]*(4*a^2 + 12*a*b + 15*b^2)*Sqrt[Pi]*Erfi[Sqrt[a
+ b*ArcCosh[c*x]]/Sqrt[b]]*(Cosh[a/b] - Sinh[a/b]) - Sqrt[b]*(4*a^2 - 12*a
*b + 15*b^2)*Sqrt[Pi]*Erf[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]]*(Cosh[a/b] + S
inh[a/b]) + 4*a*b*(-12*Sqrt[(-1 + c*x)/(1 + c*x)]*(1 + c*x)*Sqrt[a + b*Arc
Cosh[c*x]] + 8*c*x*ArcCosh[c*x]*Sqrt[a + b*ArcCosh[c*x]] + ((2*a + 3*b)*Sqr
t[Pi]*Erfi[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]]*(Cosh[a/b] - Sinh[a/b]))/Sqr
t[b] + ((2*a - 3*b)*Sqrt[Pi]*Erf[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]]*(Cosh[a
/b] + Sinh[a/b]))/Sqrt[b]))/(16*c)
```

**Rubi [A] (verified)**

Time = 1.41 (sec) , antiderivative size = 168, normalized size of antiderivative = 1.05, number of steps used = 11, number of rules used = 10,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.833$ , Rules used = {6294, 6330, 6294, 6368, 3042, 3788, 26, 2611, 2633, 2634}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\int (a + \operatorname{barccosh}(cx))^{5/2} dx$$

↓ 6294

$$x(a + \operatorname{barccosh}(cx))^{5/2} - \frac{5}{2}bc \int \frac{x(a + \operatorname{barccosh}(cx))^{3/2}}{\sqrt{cx-1}\sqrt{cx+1}} dx$$

↓ 6330

$$\frac{5}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{5/2} - \sqrt{cx-1}\sqrt{cx+1}(a + \operatorname{barccosh}(cx))^{3/2}}{c^2} - \frac{3b \int \sqrt{a + \operatorname{barccosh}(cx)} dx}{2c} \right)$$

↓ 6294

$$\frac{5}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{5/2} - \sqrt{cx-1}\sqrt{cx+1}(a + \operatorname{barccosh}(cx))^{3/2}}{c^2} - \frac{3b \left( x\sqrt{a + \operatorname{barccosh}(cx)} - \frac{1}{2}bc \int \frac{x}{\sqrt{cx-1}\sqrt{cx+1}\sqrt{a + \operatorname{barccosh}(cx)}} dx \right)}{2c} \right)$$

↓ 6368

$$\frac{5}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{5/2} - \sqrt{cx-1}\sqrt{cx+1}(a + \operatorname{barccosh}(cx))^{3/2}}{c^2} - \frac{3b \left( x\sqrt{a + \operatorname{barccosh}(cx)} - \frac{\int \frac{\cosh\left(\frac{a}{b} - \frac{a + \operatorname{barccosh}(cx)}{b}\right)}{\sqrt{a + \operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx))}{2c} \right)}{2c} \right)$$

↓ 3042

$$\frac{5}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{5/2} - \sqrt{cx-1}\sqrt{cx+1}(a + \operatorname{barccosh}(cx))^{3/2}}{c^2} - \frac{3b \left( x\sqrt{a + \operatorname{barccosh}(cx)} - \frac{\int \frac{\sin\left(\frac{ia}{b} - \frac{i(a + \operatorname{barccosh}(cx))}{b} + \frac{\pi}{2}\right)}{\sqrt{a + \operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx))}{2c} \right)}{2c} \right)$$

$$\begin{aligned} & \downarrow \text{3788} \\ & \frac{5}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{5/2} - \frac{3b \left( x\sqrt{a + \operatorname{barccosh}(cx)} - \frac{\frac{1}{2}i \int -\frac{ie^{-\operatorname{arccosh}(cx)}}{\sqrt{a + \operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx))}{2c} \right)}{c^2}}{2c} \right) \end{aligned}$$

$$\begin{aligned} & \downarrow \text{26} \\ & \frac{5}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{5/2} - \frac{3b \left( x\sqrt{a + \operatorname{barccosh}(cx)} - \frac{\frac{1}{2} \int \frac{e^{-\operatorname{arccosh}(cx)}}{\sqrt{a + \operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx))}{2c} \right)}{c^2}}{2c} \right) \end{aligned}$$

$$\begin{aligned} & \downarrow \text{2611} \\ & \frac{5}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{5/2} - \frac{3b \left( x\sqrt{a + \operatorname{barccosh}(cx)} - \frac{\int e^{\frac{a}{b} - \frac{a + \operatorname{barccosh}(cx)}{b}} d\sqrt{a + \operatorname{barccosh}(cx)} + \frac{1}{2c}}{c^2}}{2c} \right)}{c^2}}{2c} \right) \end{aligned}$$

$$\begin{aligned} & \downarrow \text{2633} \\ & \frac{5}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{5/2} - \frac{3b \left( x\sqrt{a + \operatorname{barccosh}(cx)} - \frac{\int e^{\frac{a}{b} - \frac{a + \operatorname{barccosh}(cx)}{b}} d\sqrt{a + \operatorname{barccosh}(cx)} + \frac{1}{2c}}{c^2}}{2c} \right)}{c^2}}{2c} \right) \end{aligned}$$

$$\downarrow \text{2634}$$

$$\frac{5}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{5/2} - \sqrt{cx-1}\sqrt{cx+1}(a + \operatorname{barccosh}(cx))^{3/2}}{c^2} - \frac{3b \left( x\sqrt{a + \operatorname{barccosh}(cx)} - \frac{\frac{1}{2}\sqrt{\pi}\sqrt{b}e^{a/b}\operatorname{erf}\left(\frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}}\right) + \frac{1}{2}\sqrt{\pi}}{2c} \right)}{2c} \right)$$

input `Int[(a + b*ArcCosh[c*x])^(5/2),x]`

output `x*(a + b*ArcCosh[c*x])^(5/2) - (5*b*c*((Sqrt[-1 + c*x]*Sqrt[1 + c*x]*(a + b*ArcCosh[c*x])^(3/2))/c^2 - (3*b*(x*Sqrt[a + b*ArcCosh[c*x]] - ((Sqrt[b]*E^(a/b)*Sqrt[Pi]*Erf[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]])/2 + (Sqrt[b]*Sqrt[Pi]*Erfi[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]])/(2*E^(a/b)))/(2*c)))/(2*c)))/2`

### Defintions of rubi rules used

rule 26 `Int[(Complex[0, a_]*(Fx_), x_Symbol] := Simp[(Complex[Identity[0], a]) Int[Fx, x], x] /; FreeQ[a, x] && EqQ[a^2, 1]`

rule 2611 `Int[(F_)^((g_.)*(e_.) + (f_.)*(x_)))/Sqrt[(c_.) + (d_.)*(x_)], x_Symbol] := Simp[2/d Subst[Int[F^(g*(e - c*(f/d)) + f*g*(x^2/d)), x], x, Sqrt[c + d*x]], x] /; FreeQ[{F, c, d, e, f, g}, x] && !TrueQ[$UseGamma]`

rule 2633 `Int[(F_)^((a_.) + (b_.)*((c_.) + (d_.)*(x_)^2), x_Symbol] := Simp[F^a*Sqrt[Pi]*(Erfi[(c + d*x)*Rt[b*Log[F], 2]]/(2*d*Rt[b*Log[F], 2])), x] /; FreeQ[{F, a, b, c, d}, x] && PosQ[b]`

rule 2634 `Int[(F_)^((a_.) + (b_.)*((c_.) + (d_.)*(x_)^2), x_Symbol] := Simp[F^a*Sqrt[Pi]*(Erf[(c + d*x)*Rt[(-b)*Log[F], 2]]/(2*d*Rt[(-b)*Log[F], 2])), x] /; FreeQ[{F, a, b, c, d}, x] && NegQ[b]`

rule 3042 `Int[u_, x_Symbol] := Int[DeactivateTrig[u, x], x] /; FunctionOfTrigOfLinearQ[u, x]`

rule 3788 `Int[((c_.) + (d_.)*(x_)^(m_.)*sin[(e_.) + Pi*(k_.) + (f_.)*(x_)], x_Symbol] := Simp[I/2 Int[(c + d*x)^m/(E^(I*k*Pi)*E^(I*(e + f*x))), x], x] - Simp[I/2 Int[(c + d*x)^m*E^(I*k*Pi)*E^(I*(e + f*x)), x], x] /; FreeQ[{c, d, e, f, m}, x] && IntegerQ[2*k]`

rule 6294 `Int[((a_.) + ArcCosh[(c_.)*(x_)]*(b_.))^(n_.), x_Symbol] := Simp[x*(a + b*ArcCosh[c*x])^n, x] - Simp[b*c*n Int[x*(a + b*ArcCosh[c*x])^(n - 1)/(Sqrt[1 + c*x]*Sqrt[-1 + c*x]), x], x] /; FreeQ[{a, b, c}, x] && GtQ[n, 0]`

rule 6330 `Int[((a_.) + ArcCosh[(c_.)*(x_)]*(b_.))^(n_.)*(x_)*((d1_) + (e1_.)*(x_)^(p_) * ((d2_) + (e2_.)*(x_)^(p_)), x_Symbol] := Simp[(d1 + e1*x)^(p + 1)*(d2 + e2*x)^(p + 1)*((a + b*ArcCosh[c*x])^n/(2*e1*e2*(p + 1))), x] - Simp[b*(n/(2*c*(p + 1)))*Simp[(d1 + e1*x)^p/(1 + c*x)^p]*Simp[(d2 + e2*x)^p/(-1 + c*x)^p] Int[(1 + c*x)^(p + 1/2)*(-1 + c*x)^(p + 1/2)*(a + b*ArcCosh[c*x])^(n - 1), x], x] /; FreeQ[{a, b, c, d1, e1, d2, e2, p}, x] && EqQ[e1, c*d1] && EqQ[e2, (-c)*d2] && GtQ[n, 0] && NeQ[p, -1]`

rule 6368 `Int[((a_.) + ArcCosh[(c_.)*(x_)]*(b_.))^(n_.)*(x_)^(m_.)*((d1_) + (e1_.)*(x_)^(p_.)*((d2_) + (e2_.)*(x_)^(p_)), x_Symbol] := Simp[(1/(b*c^(m + 1)))*Simp[(d1 + e1*x)^p/(1 + c*x)^p]*Simp[(d2 + e2*x)^p/(-1 + c*x)^p] Subst[Int[x^n*Cosh[-a/b + x/b]^m*Sinh[-a/b + x/b]^(2*p + 1), x], x, a + b*ArcCosh[c*x]], x] /; FreeQ[{a, b, c, d1, e1, d2, e2, n}, x] && EqQ[e1, c*d1] && EqQ[e2, (-c)*d2] && IGtQ[p + 3/2, 0] && IGtQ[m, 0]`

## Maple [F]

$$\int (a + b \operatorname{arccosh}(cx))^{\frac{5}{2}} dx$$

input `int((a+b*arccosh(c*x))^(5/2),x)`

output `int((a+b*arccosh(c*x))^(5/2),x)`

### Fricas [F(-2)]

Exception generated.

$$\int (a + b \operatorname{arccosh}(cx))^{5/2} dx = \text{Exception raised: TypeError}$$

input `integrate((a+b*arccosh(c*x))^(5/2),x, algorithm="fricas")`

output `Exception raised: TypeError >> Error detected within library code: integrate: implementation incomplete (constant residues)`

### Sympy [F(-1)]

Timed out.

$$\int (a + b \operatorname{arccosh}(cx))^{5/2} dx = \text{Timed out}$$

input `integrate((a+b*acosh(c*x))**(5/2),x)`

output `Timed out`

### Maxima [F]

$$\int (a + b \operatorname{arccosh}(cx))^{5/2} dx = \int (b \operatorname{arccosh}(cx) + a)^{5/2} dx$$

input `integrate((a+b*arccosh(c*x))^(5/2),x, algorithm="maxima")`

output `integrate((b*arccosh(c*x) + a)^(5/2), x)`



**Giac [F(-2)]**

Exception generated.

$$\int (a + \operatorname{barccosh}(cx))^{5/2} dx = \text{Exception raised: RuntimeError}$$

input `integrate((a+b*arccosh(c*x))^(5/2),x, algorithm="giac")`

output `Exception raised: RuntimeError >> an error occurred running a Giac command :INPUT:sage2OUTPUT:sym2poly/r2sym(const gen & e,const index_m & i,const ve cteur & l) Error: Bad Argument Value`

**Mupad [F(-1)]**

Timed out.

$$\int (a + \operatorname{barccosh}(cx))^{5/2} dx = \int (a + b \operatorname{acosh}(cx))^{5/2} dx$$

input `int((a + b*acosh(c*x))^(5/2),x)`

output `int((a + b*acosh(c*x))^(5/2), x)`

**Reduce [F]**

$$\begin{aligned} \int (a + \operatorname{barccosh}(cx))^{5/2} dx &= \left( \int \sqrt{\operatorname{acosh}(cx) b + a} dx \right) a^2 \\ &+ 2 \left( \int \sqrt{\operatorname{acosh}(cx) b + a} \operatorname{acosh}(cx) dx \right) ab \\ &+ \left( \int \sqrt{\operatorname{acosh}(cx) b + a} \operatorname{acosh}(cx)^2 dx \right) b^2 \end{aligned}$$

input `int((a+b*acosh(c*x))^(5/2),x)`

output

```
int(sqrt(acosh(c*x)*b + a),x)*a**2 + 2*int(sqrt(acosh(c*x)*b + a)*acosh(c*  
x),x)*a*b + int(sqrt(acosh(c*x)*b + a)*acosh(c*x)**2,x)*b**2
```

### 3.9 $\int (a + b \operatorname{arccosh}(cx))^{3/2} dx$

Optimal result	90
Mathematica [A] (warning: unable to verify)	91
Rubi [C] (verified)	91
Maple [F]	95
Fricas [F(-2)]	95
Sympy [F]	96
Maxima [F]	96
Giac [F]	96
Mupad [F(-1)]	97
Reduce [F]	97

#### Optimal result

Integrand size = 12, antiderivative size = 140

$$\int (a + b \operatorname{arccosh}(cx))^{3/2} dx = -\frac{3b\sqrt{-1+cx}\sqrt{1+cx}\sqrt{a+b\operatorname{arccosh}(cx)}}{2c}$$

$$+ x(a + b \operatorname{arccosh}(cx))^{3/2} - \frac{3b^{3/2}e^{a/b}\sqrt{\pi}\operatorname{erf}\left(\frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{8c}$$

$$+ \frac{3b^{3/2}e^{-\frac{a}{b}}\sqrt{\pi}\operatorname{erfi}\left(\frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{8c}$$

output

```
-3/2*b*(c*x-1)^(1/2)*(c*x+1)^(1/2)*(a+b*arccosh(c*x))^(1/2)/c+x*(a+b*arccosh(c*x))^(3/2)-3/8*b^(3/2)*exp(a/b)*Pi^(1/2)*erf((a+b*arccosh(c*x))^(1/2)/b^(1/2))/c+3/8*b^(3/2)*Pi^(1/2)*erfi((a+b*arccosh(c*x))^(1/2)/b^(1/2))/c*exp(a/b)
```

### Mathematica [A] (warning: unable to verify)

Time = 0.19 (sec) , antiderivative size = 269, normalized size of antiderivative = 1.92

$$\int (a + b \operatorname{arccosh}(cx))^{3/2} dx = \frac{ae^{-\frac{a}{b}} \sqrt{a + b \operatorname{arccosh}(cx)} \left( \frac{e^{\frac{2a}{b}} \Gamma\left(\frac{3}{2}, \frac{a}{b} + \operatorname{arccosh}(cx)\right)}{\sqrt{\frac{a}{b} + \operatorname{arccosh}(cx)}} + \frac{\Gamma\left(\frac{3}{2}, -\frac{a + b \operatorname{arccosh}(cx)}{b}\right)}{\sqrt{-\frac{a + b \operatorname{arccosh}(cx)}{b}}}\right)}{2c} + \frac{b \left( -12 \sqrt{\frac{-1 + cx}{1 + cx}} (1 + cx) \sqrt{a + b \operatorname{arccosh}(cx)} + 8cx \operatorname{arccosh}(cx) \sqrt{a + b \operatorname{arccosh}(cx)} + \frac{(2a + 3b) \sqrt{\pi} \operatorname{erfi}\left(\frac{\sqrt{a + b \operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{8c} \right)}{8c}$$

input `Integrate[(a + b*ArcCosh[c*x])^(3/2), x]`

output `(a*Sqrt[a + b*ArcCosh[c*x]]*(E^((2*a)/b)*Gamma[3/2, a/b + ArcCosh[c*x]])/Sqrt[a/b + ArcCosh[c*x]] + Gamma[3/2, -(a + b*ArcCosh[c*x])/b])/Sqrt[-((a + b*ArcCosh[c*x])/b)))/(2*c*E^(a/b)) + (b*(-12*Sqrt[(-1 + c*x)/(1 + c*x)]*(1 + c*x)*Sqrt[a + b*ArcCosh[c*x]] + 8*c*x*ArcCosh[c*x]*Sqrt[a + b*ArcCosh[c*x]] + ((2*a + 3*b)*Sqrt[Pi]*Erfi[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]]*(Cosh[a/b] - Sinh[a/b])/Sqrt[b] + ((2*a - 3*b)*Sqrt[Pi]*Erf[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]]*(Cosh[a/b] + Sinh[a/b])/Sqrt[b])))/(8*c)`

### Rubi [C] (verified)

Result contains complex when optimal does not.

Time = 0.91 (sec) , antiderivative size = 151, normalized size of antiderivative = 1.08, number of steps used = 11, number of rules used = 10,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.833$ , Rules used = {6294, 6330, 6296, 25, 3042, 26, 3789, 2611, 2633, 2634}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\int (a + b \operatorname{arccosh}(cx))^{3/2} dx$$

$$\begin{aligned}
& \downarrow 6294 \\
& x(a + \operatorname{barccosh}(cx))^{3/2} - \frac{3}{2}bc \int \frac{x\sqrt{a + \operatorname{barccosh}(cx)}}{\sqrt{cx-1}\sqrt{cx+1}} dx \\
& \downarrow 6330 \\
& x(a + \operatorname{barccosh}(cx))^{3/2} - \frac{3}{2}bc \left( \frac{\sqrt{cx-1}\sqrt{cx+1}\sqrt{a + \operatorname{barccosh}(cx)}}{c^2} - \frac{b \int \frac{1}{\sqrt{a + \operatorname{barccosh}(cx)}} dx}{2c} \right) \\
& \downarrow 6296 \\
& \frac{3}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{3/2} - \int \frac{\sinh\left(\frac{a}{b} - \frac{a + b\operatorname{barccosh}(cx)}{b}\right)}{\sqrt{a + \operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx))}{c^2} - \frac{\int \frac{\sinh\left(\frac{a}{b} - \frac{a + b\operatorname{barccosh}(cx)}{b}\right)}{\sqrt{a + \operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx))}{2c^2} \right) \\
& \downarrow 25 \\
& \frac{3}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{3/2} - \int \frac{\sinh\left(\frac{a}{b} - \frac{a + b\operatorname{barccosh}(cx)}{b}\right)}{\sqrt{a + \operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx))}{2c^2} + \frac{\sqrt{cx-1}\sqrt{cx+1}\sqrt{a + \operatorname{barccosh}(cx)}}{c^2} \right) \\
& \downarrow 3042 \\
& \frac{3}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{3/2} - \int \frac{i \sin\left(\frac{ia}{b} - \frac{i(a + b\operatorname{barccosh}(cx))}{b}\right)}{\sqrt{a + \operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx))}{2c^2} + \frac{\sqrt{cx-1}\sqrt{cx+1}\sqrt{a + \operatorname{barccosh}(cx)}}{c^2} \right) \\
& \downarrow 26 \\
& \frac{3}{2}bc \left( \frac{x(a + \operatorname{barccosh}(cx))^{3/2} - i \int \frac{\sin\left(\frac{ia}{b} - \frac{i(a + b\operatorname{barccosh}(cx))}{b}\right)}{\sqrt{a + \operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx))}{2c^2} + \frac{\sqrt{cx-1}\sqrt{cx+1}\sqrt{a + \operatorname{barccosh}(cx)}}{c^2} \right)
\end{aligned}$$

$$\begin{aligned} & \downarrow \text{3789} \\ & \frac{x(a + \operatorname{barccosh}(cx))^{3/2} - \frac{3}{2}bc \left( \frac{\sqrt{cx-1}\sqrt{cx+1}\sqrt{a + \operatorname{barccosh}(cx)}}{c^2} - \frac{i \left( \frac{1}{2}i \int \frac{e^{-\operatorname{arccosh}(cx)}}{\sqrt{a+\operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx)) - \frac{1}{2}i \int \frac{e^{\operatorname{arccosh}(cx)}}{\sqrt{a+\operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx)) \right)}{2c^2} \right)}{2c^2} \end{aligned}$$

$$\begin{aligned} & \downarrow \text{2611} \\ & \frac{x(a + \operatorname{barccosh}(cx))^{3/2} - \frac{3}{2}bc \left( \frac{\sqrt{cx-1}\sqrt{cx+1}\sqrt{a + \operatorname{barccosh}(cx)}}{c^2} - \frac{i \left( i \int e^{\frac{a}{b} - \frac{a+\operatorname{barccosh}(cx)}{b}} d\sqrt{a + \operatorname{barccosh}(cx)} - i \int e^{\frac{a+\operatorname{barccosh}(cx)}{b} - \frac{a}{b}} d\sqrt{a + \operatorname{barccosh}(cx)} \right)}{2c^2} \right)}{2c^2} \end{aligned}$$

$$\begin{aligned} & \downarrow \text{2633} \\ & \frac{x(a + \operatorname{barccosh}(cx))^{3/2} - \frac{3}{2}bc \left( \frac{\sqrt{cx-1}\sqrt{cx+1}\sqrt{a + \operatorname{barccosh}(cx)}}{c^2} - \frac{i \left( i \int e^{\frac{a}{b} - \frac{a+\operatorname{barccosh}(cx)}{b}} d\sqrt{a + \operatorname{barccosh}(cx)} - \frac{1}{2}i\sqrt{\pi}\sqrt{b}e^{-\frac{a}{b}} \operatorname{erfi} \left( \frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}} \right) \right)}{2c^2} \right)}{2c^2} \end{aligned}$$

$$\begin{aligned} & \downarrow \text{2634} \\ & \frac{x(a + \operatorname{barccosh}(cx))^{3/2} - \frac{3}{2}bc \left( \frac{\sqrt{cx-1}\sqrt{cx+1}\sqrt{a + \operatorname{barccosh}(cx)}}{c^2} - \frac{i \left( \frac{1}{2}i\sqrt{\pi}\sqrt{b}e^{a/b} \operatorname{erf} \left( \frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}} \right) - \frac{1}{2}i\sqrt{\pi}\sqrt{b}e^{-\frac{a}{b}} \operatorname{erfi} \left( \frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}} \right) \right)}{2c^2} \right)}{2c^2} \end{aligned}$$

input `Int[(a + b*ArcCosh[c*x])^(3/2),x]`

output `x*(a + b*ArcCosh[c*x])^(3/2) - (3*b*c*((Sqrt[-1 + c*x]*Sqrt[1 + c*x]*Sqrt[a + b*ArcCosh[c*x]])/c^2 - ((I/2)*((I/2)*Sqrt[b]*E^(a/b)*Sqrt[Pi]*Erf[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]] - ((I/2)*Sqrt[b]*Sqrt[Pi]*Erfi[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]])/E^(a/b)))/c^2))/2`

## Definitions of rubi rules used

- rule 25  $\text{Int}[-(\text{Fx}_), \text{x\_Symbol}] \rightarrow \text{Simp}[\text{Identity}[-1] \text{ Int}[\text{Fx}, \text{x}], \text{x}]$
- rule 26  $\text{Int}[(\text{Complex}[0, \text{a}_])*(\text{Fx}_), \text{x\_Symbol}] \rightarrow \text{Simp}[(\text{Complex}[\text{Identity}[0], \text{a}]) \text{ Int}[\text{Fx}, \text{x}], \text{x}] /; \text{FreeQ}[\text{a}, \text{x}] \ \&\& \ \text{EqQ}[\text{a}^2, 1]$
- rule 2611  $\text{Int}[(\text{F}_)^{((\text{g}_)*((\text{e}_.) + (\text{f}_.)*(x_)))}/\text{Sqrt}[(\text{c}_.) + (\text{d}_.)*(x_)], \text{x\_Symbol}] \rightarrow \text{Simp}[2/\text{d} \text{ Subst}[\text{Int}[\text{F}^{(\text{g}*(\text{e} - \text{c}*(\text{f}/\text{d}) + \text{f}*g*(x^2/\text{d}))}, \text{x}], \text{x}, \text{Sqrt}[\text{c} + \text{d}*x]], \text{x}] /; \text{FreeQ}[\{\text{F}, \text{c}, \text{d}, \text{e}, \text{f}, \text{g}\}, \text{x}] \ \&\& \ \text{!TrueQ}[\$UseGamma]$
- rule 2633  $\text{Int}[(\text{F}_)^{((\text{a}_.) + (\text{b}_.)*((\text{c}_.) + (\text{d}_.)*(x_))^{2})}, \text{x\_Symbol}] \rightarrow \text{Simp}[\text{F}^{\text{a}}*\text{Sqrt}[\text{Pi}]*(\text{Erfi}[(\text{c} + \text{d}*x)*\text{Rt}[\text{b}*\text{Log}[\text{F}], 2]]/(2*\text{d}*\text{Rt}[\text{b}*\text{Log}[\text{F}], 2])), \text{x}] /; \text{FreeQ}[\{\text{F}, \text{a}, \text{b}, \text{c}, \text{d}\}, \text{x}] \ \&\& \ \text{PosQ}[\text{b}]$
- rule 2634  $\text{Int}[(\text{F}_)^{((\text{a}_.) + (\text{b}_.)*((\text{c}_.) + (\text{d}_.)*(x_))^{2})}, \text{x\_Symbol}] \rightarrow \text{Simp}[\text{F}^{\text{a}}*\text{Sqrt}[\text{Pi}]*(\text{Erf}[(\text{c} + \text{d}*x)*\text{Rt}[(\text{-b})*\text{Log}[\text{F}], 2]]/(2*\text{d}*\text{Rt}[(\text{-b})*\text{Log}[\text{F}], 2])), \text{x}] /; \text{FreeQ}[\{\text{F}, \text{a}, \text{b}, \text{c}, \text{d}\}, \text{x}] \ \&\& \ \text{NegQ}[\text{b}]$
- rule 3042  $\text{Int}[\text{u}_, \text{x\_Symbol}] \rightarrow \text{Int}[\text{DeactivateTrig}[\text{u}, \text{x}], \text{x}] /; \text{FunctionOfTrigOfLinearQ}[\text{u}, \text{x}]$
- rule 3789  $\text{Int}[(\text{c}_.) + (\text{d}_.)*(x_))^{(\text{m}_.)}*\sin[(\text{e}_.) + (\text{f}_.)*(x_)], \text{x\_Symbol}] \rightarrow \text{Simp}[\text{I}/2 \text{ Int}[(\text{c} + \text{d}*x)^{\text{m}}/\text{E}^{\text{I}*(\text{e} + \text{f}*x)}, \text{x}], \text{x}] - \text{Simp}[\text{I}/2 \text{ Int}[(\text{c} + \text{d}*x)^{\text{m}}*\text{E}^{\text{I}*(\text{e} + \text{f}*x)}, \text{x}], \text{x}] /; \text{FreeQ}[\{\text{c}, \text{d}, \text{e}, \text{f}, \text{m}\}, \text{x}]$
- rule 6294  $\text{Int}[(\text{a}_.) + \text{ArcCosh}[(\text{c}_.)*(x_)]*(\text{b}_.)^{(\text{n}_.)}, \text{x\_Symbol}] \rightarrow \text{Simp}[\text{x}*(\text{a} + \text{b}*\text{ArcCosh}[\text{c}*x])^{\text{n}}, \text{x}] - \text{Simp}[\text{b}*c*\text{n} \text{ Int}[\text{x}*((\text{a} + \text{b}*\text{ArcCosh}[\text{c}*x])^{(\text{n} - 1)})/(\text{Sqrt}[1 + \text{c}*x]*\text{Sqrt}[-1 + \text{c}*x])], \text{x}], \text{x}] /; \text{FreeQ}[\{\text{a}, \text{b}, \text{c}\}, \text{x}] \ \&\& \ \text{GtQ}[\text{n}, 0]$

rule 6296 `Int[((a_.) + ArcCosh[(c_.)*(x_)]*(b_.))^(n_), x_Symbol] := Simp[1/(b*c) S  
ubst[Int[x^n*Sinh[-a/b + x/b], x], x, a + b*ArcCosh[c*x]], x] /; FreeQ[{a,  
b, c, n}, x]`

rule 6330 `Int[((a_.) + ArcCosh[(c_.)*(x_)]*(b_.))^(n_.)*(x_)*((d1_) + (e1_.)*(x_))^(p  
_)*((d2_) + (e2_.)*(x_))^(p_), x_Symbol] := Simp[(d1 + e1*x)^(p + 1)*(d2 +  
e2*x)^(p + 1)*((a + b*ArcCosh[c*x])^n/(2*e1*e2*(p + 1))), x] - Simp[b*(n/(2  
*c*(p + 1)))*Simp[(d1 + e1*x)^p/(1 + c*x)^p]*Simp[(d2 + e2*x)^p/(-1 + c*x)^  
p] Int[(1 + c*x)^(p + 1/2)*(-1 + c*x)^(p + 1/2)*(a + b*ArcCosh[c*x])^(n -  
1), x], x] /; FreeQ[{a, b, c, d1, e1, d2, e2, p}, x] && EqQ[e1, c*d1] && E  
qQ[e2, (-c)*d2] && GtQ[n, 0] && NeQ[p, -1]`

## Maple [F]

$$\int (a + b \operatorname{arccosh}(cx))^{\frac{3}{2}} dx$$

input `int((a+b*arccosh(c*x))^(3/2),x)`

output `int((a+b*arccosh(c*x))^(3/2),x)`

## Fricas [F(-2)]

Exception generated.

$$\int (a + b \operatorname{arccosh}(cx))^{3/2} dx = \text{Exception raised: TypeError}$$

input `integrate((a+b*arccosh(c*x))^(3/2),x, algorithm="fricas")`

output `Exception raised: TypeError >> Error detected within library code: inte  
grate: implementation incomplete (constant residues)`



**Sympy [F]**

$$\int (a + b \operatorname{arccosh}(cx))^{3/2} dx = \int (a + b \operatorname{acosh}(cx))^{\frac{3}{2}} dx$$

input `integrate((a+b*acosh(c*x))**(3/2),x)`

output `Integral((a + b*acosh(c*x))**(3/2), x)`

**Maxima [F]**

$$\int (a + b \operatorname{arccosh}(cx))^{3/2} dx = \int (b \operatorname{arcosh}(cx) + a)^{\frac{3}{2}} dx$$

input `integrate((a+b*arccosh(c*x))^(3/2),x, algorithm="maxima")`

output `integrate((b*arccosh(c*x) + a)^(3/2), x)`

**Giac [F]**

$$\int (a + b \operatorname{arccosh}(cx))^{3/2} dx = \int (b \operatorname{arcosh}(cx) + a)^{\frac{3}{2}} dx$$

input `integrate((a+b*arccosh(c*x))^(3/2),x, algorithm="giac")`

output `integrate((b*arccosh(c*x) + a)^(3/2), x)`

**Mupad [F(-1)]**

Timed out.

$$\int (a + b \operatorname{arccosh}(cx))^{3/2} dx = \int (a + b \operatorname{acosh}(cx))^{3/2} dx$$

input `int((a + b*acosh(c*x))^(3/2), x)`output `int((a + b*acosh(c*x))^(3/2), x)`**Reduce [F]**

$$\int (a + b \operatorname{arccosh}(cx))^{3/2} dx = \left( \int \sqrt{a \operatorname{cosh}(cx) b + a} dx \right) a$$

$$+ \left( \int \sqrt{a \operatorname{cosh}(cx) b + a} \operatorname{acosh}(cx) dx \right) b$$

input `int((a+b*acosh(c*x))^(3/2), x)`output `int(sqrt(acosh(c*x)*b + a), x)*a + int(sqrt(acosh(c*x)*b + a)*acosh(c*x), x)*b`

### 3.10 $\int \sqrt{a + b \operatorname{arccosh}(cx)} dx$

Optimal result	98
Mathematica [A] (verified)	98
Rubi [A] (verified)	99
Maple [F]	102
Fricas [F(-2)]	102
Sympy [F]	102
Maxima [F]	103
Giac [F]	103
Mupad [F(-1)]	103
Reduce [F]	104

#### Optimal result

Integrand size = 12, antiderivative size = 102

$$\int \sqrt{a + b \operatorname{arccosh}(cx)} dx = x \sqrt{a + b \operatorname{arccosh}(cx)} - \frac{\sqrt{b} e^{a/b} \sqrt{\pi} \operatorname{erf}\left(\frac{\sqrt{a + b \operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{4c} - \frac{\sqrt{b} e^{-\frac{a}{b}} \sqrt{\pi} \operatorname{erfi}\left(\frac{\sqrt{a + b \operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{4c}$$

output

```
x*(a+b*arccosh(c*x))^(1/2)-1/4*b^(1/2)*exp(a/b)*Pi^(1/2)*erf((a+b*arccosh(c*x))^(1/2)/b^(1/2))/c-1/4*b^(1/2)*Pi^(1/2)*erfi((a+b*arccosh(c*x))^(1/2)/b^(1/2))/c/exp(a/b)
```

#### Mathematica [A] (verified)

Time = 0.13 (sec) , antiderivative size = 100, normalized size of antiderivative = 0.98

$$\int \sqrt{a + b \operatorname{arccosh}(cx)} dx = \frac{e^{-\frac{a}{b}} \sqrt{a + b \operatorname{arccosh}(cx)} \left( \frac{e^{\frac{2a}{b}} \Gamma\left(\frac{3}{2}, \frac{a}{b} + \operatorname{arccosh}(cx)\right)}{\sqrt{\frac{a}{b} + \operatorname{arccosh}(cx)}} + \frac{\Gamma\left(\frac{3}{2}, -\frac{a + b \operatorname{arccosh}(cx)}{b}\right)}{\sqrt{-\frac{a + b \operatorname{arccosh}(cx)}{b}}} \right)}{2c}$$

input `Integrate[Sqrt[a + b*ArcCosh[c*x]], x]`

output `(Sqrt[a + b*ArcCosh[c*x]]*((E^((2*a)/b)*Gamma[3/2, a/b + ArcCosh[c*x]])/Sqrt[a/b + ArcCosh[c*x]] + Gamma[3/2, -((a + b*ArcCosh[c*x])/b)]/Sqrt[-((a + b*ArcCosh[c*x])/b)]))/(2*c*E^(a/b))`

### Rubi [A] (verified)

Time = 0.76 (sec) , antiderivative size = 104, normalized size of antiderivative = 1.02, number of steps used = 9, number of rules used = 8,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.667$ , Rules used = {6294, 6368, 3042, 3788, 26, 2611, 2633, 2634}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\begin{aligned}
 & \int \sqrt{a + b \operatorname{arccosh}(cx)} dx \\
 & \quad \downarrow 6294 \\
 & x \sqrt{a + b \operatorname{arccosh}(cx)} - \frac{1}{2} bc \int \frac{x}{\sqrt{cx - 1} \sqrt{cx + 1} \sqrt{a + b \operatorname{arccosh}(cx)}} dx \\
 & \quad \downarrow 6368 \\
 & x \sqrt{a + b \operatorname{arccosh}(cx)} - \frac{\int \frac{\cosh\left(\frac{a}{b} - \frac{a + b \operatorname{arccosh}(cx)}{b}\right)}{\sqrt{a + b \operatorname{arccosh}(cx)}} d(a + b \operatorname{arccosh}(cx))}{2c} \\
 & \quad \downarrow 3042 \\
 & x \sqrt{a + b \operatorname{arccosh}(cx)} - \frac{\int \frac{\sin\left(\frac{ia}{b} - \frac{i(a + b \operatorname{arccosh}(cx))}{b} + \frac{\pi}{2}\right)}{\sqrt{a + b \operatorname{arccosh}(cx)}} d(a + b \operatorname{arccosh}(cx))}{2c} \\
 & \quad \downarrow 3788 \\
 & \frac{x \sqrt{a + b \operatorname{arccosh}(cx)} - \frac{1}{2} i \int -\frac{ie^{-\operatorname{arccosh}(cx)}}{\sqrt{a + b \operatorname{arccosh}(cx)}} d(a + b \operatorname{arccosh}(cx)) - \frac{1}{2} i \int \frac{ie^{\operatorname{arccosh}(cx)}}{\sqrt{a + b \operatorname{arccosh}(cx)}} d(a + b \operatorname{arccosh}(cx))}{2c} \\
 & \quad \downarrow 26
 \end{aligned}$$

$$\begin{aligned}
& \frac{\frac{1}{2} \int \frac{e^{-\operatorname{arccosh}(cx)}}{\sqrt{a+\operatorname{barccosh}(cx)}} d(a+\operatorname{barccosh}(cx)) + \frac{1}{2} \int \frac{e^{\operatorname{arccosh}(cx)}}{\sqrt{a+\operatorname{barccosh}(cx)}} d(a+\operatorname{barccosh}(cx))}{2c} \\
& \quad \downarrow \text{2611} \\
& \frac{\int e^{\frac{a}{b} - \frac{a+\operatorname{barccosh}(cx)}{b}} d\sqrt{a+\operatorname{barccosh}(cx)} + \int e^{\frac{a+\operatorname{barccosh}(cx)}{b} - \frac{a}{b}} d\sqrt{a+\operatorname{barccosh}(cx)}}{2c} \\
& \quad \downarrow \text{2633} \\
& \frac{\int e^{\frac{a}{b} - \frac{a+\operatorname{barccosh}(cx)}{b}} d\sqrt{a+\operatorname{barccosh}(cx)} + \frac{1}{2} \sqrt{\pi} \sqrt{b} e^{-\frac{a}{b}} \operatorname{erfi}\left(\frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}}\right)}{2c} \\
& \quad \downarrow \text{2634} \\
& \frac{\frac{1}{2} \sqrt{\pi} \sqrt{b} e^{a/b} \operatorname{erf}\left(\frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}}\right) + \frac{1}{2} \sqrt{\pi} \sqrt{b} e^{-\frac{a}{b}} \operatorname{erfi}\left(\frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}}\right)}{2c}
\end{aligned}$$

input `Int[Sqrt[a + b*ArcCosh[c*x]], x]`

output `x*Sqrt[a + b*ArcCosh[c*x]] - ((Sqrt[b]*E^(a/b)*Sqrt[Pi]*Erf[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]])/2 + (Sqrt[b]*Sqrt[Pi]*Erfi[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]])/(2*E^(a/b))/(2*c)`

### Defintions of rubi rules used

rule 26 `Int[(Complex[0, a_]*(F_x_), x_Symbol] := Simp[(Complex[Identity[0], a]) Int[F_x, x], x] /; FreeQ[a, x] && EqQ[a^2, 1]`

rule 2611 `Int[(F_)^((g_.)*(e_.) + (f_.)*(x_))]/Sqrt[(c_.) + (d_.)*(x_)], x_Symbol] :> Simp[2/d Subst[Int[F^(g*(e - c*(f/d)) + f*g*(x^2/d)), x], x, Sqrt[c + d*x]], x] /; FreeQ[{F, c, d, e, f, g}, x] && !TrueQ[$UseGamma]`

rule 2633  $\text{Int}[(F\_)^{(a\_)} + (b\_)*((c\_)+ (d\_)*(x\_))^2], x\_Symbol] \rightarrow \text{Simp}[F^a*\text{Sqrt}[\text{Pi}*(\text{Erfi}[(c + d*x)*\text{Rt}[b*\text{Log}[F], 2]]/(2*d*\text{Rt}[b*\text{Log}[F], 2]))], x] /; \text{FreeQ}\{F, a, b, c, d\}, x] \ \&\& \ \text{PosQ}[b]$

rule 2634  $\text{Int}[(F\_)^{(a\_)} + (b\_)*((c\_)+ (d\_)*(x\_))^2], x\_Symbol] \rightarrow \text{Simp}[F^a*\text{Sqrt}[\text{Pi}*(\text{Erf}[(c + d*x)*\text{Rt}[(-b)*\text{Log}[F], 2]]/(2*d*\text{Rt}[(-b)*\text{Log}[F], 2]))], x] /; \text{FreeQ}\{F, a, b, c, d\}, x] \ \&\& \ \text{NegQ}[b]$

rule 3042  $\text{Int}[u_, x\_Symbol] \rightarrow \text{Int}[\text{DeactivateTrig}[u, x], x] /; \text{FunctionOfTrigOfLinearQ}[u, x]$

rule 3788  $\text{Int}[(c\_)+ (d\_)*(x\_))^{(m\_)}*\sin[(e\_)+ \text{Pi}*(k\_)+ (f\_)*(x\_)], x\_Symbol] \rightarrow \text{Simp}[I/2 \ \text{Int}[(c + d*x)^m/(E^{(I*k*Pi)}*E^{(I*(e + f*x))}), x], x] - \text{Simp}[I/2 \ \text{Int}[(c + d*x)^m*E^{(I*k*Pi)}*E^{(I*(e + f*x))}, x], x] /; \text{FreeQ}\{c, d, e, f, m\}, x] \ \&\& \ \text{IntegerQ}[2*k]$

rule 6294  $\text{Int}[(a\_)+ \text{ArcCosh}[(c\_)*(x\_)]*(b\_))^{(n\_)}, x\_Symbol] \rightarrow \text{Simp}[x*(a + b*\text{ArcCosh}[c*x])^n, x] - \text{Simp}[b*c*n \ \text{Int}[x*((a + b*\text{ArcCosh}[c*x])^{(n - 1)})/(\text{Sqrt}[1 + c*x]*\text{Sqrt}[-1 + c*x])], x], x] /; \text{FreeQ}\{a, b, c\}, x] \ \&\& \ \text{GtQ}[n, 0]$

rule 6368  $\text{Int}[(a\_)+ \text{ArcCosh}[(c\_)*(x\_)]*(b\_))^{(n\_)}*(x\_)^{(m\_)}*((d1\_)+ (e1\_)*(x\_))^{(p\_)}*((d2\_)+ (e2\_)*(x\_))^{(p\_)}, x\_Symbol] \rightarrow \text{Simp}[(1/(b*c^{(m + 1)}))*\text{Simp}[(d1 + e1*x)^p/(1 + c*x)^p]*\text{Simp}[(d2 + e2*x)^p/(-1 + c*x)^p] \ \text{Subst}[\text{Int}[x^n*\text{Cosh}[-a/b + x/b]^m*\text{Sinh}[-a/b + x/b]^{(2*p + 1)}, x], x, a + b*\text{ArcCosh}[c*x]], x] /; \text{FreeQ}\{a, b, c, d1, e1, d2, e2, n\}, x] \ \&\& \ \text{EqQ}[e1, c*d1] \ \&\& \ \text{EqQ}[e2, (-c)*d2] \ \&\& \ \text{IGtQ}[p + 3/2, 0] \ \&\& \ \text{IGtQ}[m, 0]$

**Maple [F]**

$$\int \sqrt{a + b \operatorname{arccosh}(cx)} dx$$

input `int((a+b*arccosh(c*x))^(1/2),x)`

output `int((a+b*arccosh(c*x))^(1/2),x)`

**Fricas [F(-2)]**

Exception generated.

$$\int \sqrt{a + b \operatorname{arccosh}(cx)} dx = \text{Exception raised: TypeError}$$

input `integrate((a+b*arccosh(c*x))^(1/2),x, algorithm="fricas")`

output `Exception raised: TypeError >> Error detected within library code: integrate: implementation incomplete (constant residues)`

**Sympy [F]**

$$\int \sqrt{a + b \operatorname{arccosh}(cx)} dx = \int \sqrt{a + b \operatorname{acosh}(cx)} dx$$

input `integrate((a+b*acosh(c*x))**(1/2),x)`

output `Integral(sqrt(a + b*acosh(c*x)), x)`

**Maxima [F]**

$$\int \sqrt{a + b \operatorname{arccosh}(cx)} dx = \int \sqrt{b \operatorname{arcosh}(cx) + a} dx$$

input `integrate((a+b*arccosh(c*x))^(1/2),x, algorithm="maxima")`

output `integrate(sqrt(b*arccosh(c*x) + a), x)`

**Giac [F]**

$$\int \sqrt{a + b \operatorname{arccosh}(cx)} dx = \int \sqrt{b \operatorname{arcosh}(cx) + a} dx$$

input `integrate((a+b*arccosh(c*x))^(1/2),x, algorithm="giac")`

output `integrate(sqrt(b*arccosh(c*x) + a), x)`

**Mupad [F(-1)]**

Timed out.

$$\int \sqrt{a + b \operatorname{arccosh}(cx)} dx = \int \sqrt{a + b \operatorname{acosh}(cx)} dx$$

input `int((a + b*acosh(c*x))^(1/2),x)`

output `int((a + b*acosh(c*x))^(1/2), x)`



**Reduce [F]**

$$\int \sqrt{a + b \operatorname{arccosh}(cx)} dx = \int \sqrt{a \operatorname{cosh}(cx) + b} dx$$

input `int((a+b*acosh(c*x))^(1/2),x)`

output `int(sqrt(acosh(c*x)*b + a),x)`

**3.11** 
$$\int \frac{1}{\sqrt{a+b\operatorname{arccosh}(cx)}} dx$$

Optimal result	105
Mathematica [A] (verified)	105
Rubi [C] (verified)	106
Maple [F]	109
Fricas [F(-2)]	109
Sympy [F]	109
Maxima [F]	110
Giac [F]	110
Mupad [F(-1)]	110
Reduce [F]	111

**Optimal result**

Integrand size = 12, antiderivative size = 88

$$\int \frac{1}{\sqrt{a+b\operatorname{arccosh}(cx)}} dx = -\frac{e^{a/b}\sqrt{\pi}\operatorname{erf}\left(\frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{2\sqrt{bc}} + \frac{e^{-\frac{a}{b}}\sqrt{\pi}\operatorname{erfi}\left(\frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{2\sqrt{bc}}$$

output `-1/2*exp(a/b)*Pi^(1/2)*erf((a+b*arccosh(c*x))^(1/2)/b^(1/2))/b^(1/2)/c+1/2*Pi^(1/2)*erfi((a+b*arccosh(c*x))^(1/2)/b^(1/2))/b^(1/2)/c/exp(a/b)`

**Mathematica [A] (verified)**

Time = 0.10 (sec) , antiderivative size = 100, normalized size of antiderivative = 1.14

$$\int \frac{1}{\sqrt{a+b\operatorname{arccosh}(cx)}} dx = \frac{e^{-\frac{a}{b}} \left( e^{\frac{2a}{b}} \sqrt{\frac{a}{b} + \operatorname{arccosh}(cx)} \Gamma\left(\frac{1}{2}, \frac{a}{b} + \operatorname{arccosh}(cx)\right) + \sqrt{-\frac{a+b\operatorname{arccosh}(cx)}{b}} \Gamma\left(\frac{1}{2}, -\frac{a+b\operatorname{arccosh}(cx)}{b}\right) \right)}{2c\sqrt{a+b\operatorname{arccosh}(cx)}}$$

input `Integrate[1/Sqrt[a + b*ArcCosh[c*x]], x]`

output `(E^((2*a)/b)*Sqrt[a/b + ArcCosh[c*x]]*Gamma[1/2, a/b + ArcCosh[c*x]] + Sqrt[-((a + b*ArcCosh[c*x])/b)]*Gamma[1/2, -((a + b*ArcCosh[c*x])/b)])/(2*c*E^(a/b)*Sqrt[a + b*ArcCosh[c*x]])`

## Rubi [C] (verified)

Result contains complex when optimal does not.

Time = 0.42 (sec) , antiderivative size = 96, normalized size of antiderivative = 1.09, number of steps used = 9, number of rules used = 8,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.667$ , Rules used = {6296, 25, 3042, 26, 3789, 2611, 2633, 2634}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\begin{array}{c}
 \int \frac{1}{\sqrt{a + b \operatorname{arccosh}(cx)}} dx \\
 \downarrow 6296 \\
 \int \frac{\sinh\left(\frac{a}{b} - \frac{a + b \operatorname{arccosh}(cx)}{b}\right)}{\sqrt{a + b \operatorname{arccosh}(cx)}} d(a + b \operatorname{arccosh}(cx)) \\
 \hline bc \\
 \downarrow 25 \\
 \int \frac{\sinh\left(\frac{a}{b} - \frac{a + b \operatorname{arccosh}(cx)}{b}\right)}{\sqrt{a + b \operatorname{arccosh}(cx)}} d(a + b \operatorname{arccosh}(cx)) \\
 \hline bc \\
 \downarrow 3042 \\
 \int \frac{i \sin\left(\frac{ia}{b} - \frac{i(a + b \operatorname{arccosh}(cx))}{b}\right)}{\sqrt{a + b \operatorname{arccosh}(cx)}} d(a + b \operatorname{arccosh}(cx)) \\
 \hline bc \\
 \downarrow 26
 \end{array}$$

$$\begin{array}{c}
 \frac{i \int \frac{\sin\left(\frac{ia}{b} - \frac{i(a+b\operatorname{arccosh}(cx))}{b}\right)}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx))}{bc} \\
 \downarrow \text{3789} \\
 \frac{i \left( \frac{1}{2} i \int \frac{e^{-\operatorname{arccosh}(cx)}}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx)) - \frac{1}{2} i \int \frac{e^{\operatorname{arccosh}(cx)}}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx)) \right)}{bc} \\
 \downarrow \text{2611} \\
 \frac{i \left( i \int e^{\frac{a}{b} - \frac{a+b\operatorname{arccosh}(cx)}{b}} d\sqrt{a+b\operatorname{arccosh}(cx)} - i \int e^{\frac{a+b\operatorname{arccosh}(cx)}{b} - \frac{a}{b}} d\sqrt{a+b\operatorname{arccosh}(cx)} \right)}{bc} \\
 \downarrow \text{2633} \\
 \frac{i \left( i \int e^{\frac{a}{b} - \frac{a+b\operatorname{arccosh}(cx)}{b}} d\sqrt{a+b\operatorname{arccosh}(cx)} - \frac{1}{2} i \sqrt{\pi} \sqrt{b} e^{-\frac{a}{b}} \operatorname{erfi} \left( \frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}} \right) \right)}{bc} \\
 \downarrow \text{2634} \\
 \frac{i \left( \frac{1}{2} i \sqrt{\pi} \sqrt{b} e^{a/b} \operatorname{erf} \left( \frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}} \right) - \frac{1}{2} i \sqrt{\pi} \sqrt{b} e^{-\frac{a}{b}} \operatorname{erfi} \left( \frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}} \right) \right)}{bc}
 \end{array}$$

input `Int[1/Sqrt[a + b*ArcCosh[c*x]], x]`

output `(I*((I/2)*Sqrt[b]*E^(a/b)*Sqrt[Pi]*Erf[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]] - ((I/2)*Sqrt[b]*Sqrt[Pi]*Erfi[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]])/E^(a/b)))/(b*c)`

## Definitions of rubi rules used

- rule 25 `Int[-(Fx_), x_Symbol] := Simp[Identity[-1] Int[Fx, x], x]`
- rule 26 `Int[(Complex[0, a_])*(Fx_), x_Symbol] := Simp[(Complex[Identity[0], a]) Int[Fx, x], x] /; FreeQ[a, x] && EqQ[a^2, 1]`
- rule 2611 `Int[(F_)^((g_)*((e_) + (f_)*(x_)))/Sqrt[(c_) + (d_)*(x_)], x_Symbol] :> Simp[2/d Subst[Int[F^(g*(e - c*(f/d)) + f*g*(x^2/d)), x], x, Sqrt[c + d*x]], x] /; FreeQ[{F, c, d, e, f, g}, x] && !TrueQ[$UseGamma]`
- rule 2633 `Int[(F_)^((a_) + (b_)*((c_) + (d_)*(x_)^2), x_Symbol] := Simp[F^a*Sqrt[Pi]*(Erfi[(c + d*x)*Rt[b*Log[F], 2]]/(2*d*Rt[b*Log[F], 2])), x] /; FreeQ[{F, a, b, c, d}, x] && PosQ[b]`
- rule 2634 `Int[(F_)^((a_) + (b_)*((c_) + (d_)*(x_)^2), x_Symbol] := Simp[F^a*Sqrt[Pi]*(Erf[(c + d*x)*Rt[(-b)*Log[F], 2]]/(2*d*Rt[(-b)*Log[F], 2])), x] /; FreeQ[{F, a, b, c, d}, x] && NegQ[b]`
- rule 3042 `Int[u_, x_Symbol] := Int[DeactivateTrig[u, x], x] /; FunctionOfTrigOfLinearQ[u, x]`
- rule 3789 `Int[((c_) + (d_)*(x_))^(m_)*sin[(e_) + (f_)*(x_)], x_Symbol] := Simp[I/2 Int[(c + d*x)^m/E^(I*(e + f*x)), x], x] - Simp[I/2 Int[(c + d*x)^m*E^(I*(e + f*x)), x], x] /; FreeQ[{c, d, e, f, m}, x]`
- rule 6296 `Int[((a_) + ArcCosh[(c_)*(x_)])*(b_)^(n_), x_Symbol] := Simp[1/(b*c) Subst[Int[x^n*Sinh[-a/b + x/b], x], x, a + b*ArcCosh[c*x]], x] /; FreeQ[{a, b, c, n}, x]`

**Maple [F]**

$$\int \frac{1}{\sqrt{a + b \operatorname{arccosh}(cx)}} dx$$

input `int(1/(a+b*arccosh(c*x))^(1/2),x)`

output `int(1/(a+b*arccosh(c*x))^(1/2),x)`

**Fricas [F(-2)]**

Exception generated.

$$\int \frac{1}{\sqrt{a + b \operatorname{arccosh}(cx)}} dx = \text{Exception raised: TypeError}$$

input `integrate(1/(a+b*arccosh(c*x))^(1/2),x, algorithm="fricas")`

output `Exception raised: TypeError >> Error detected within library code: integrate: implementation incomplete (constant residues)`

**Sympy [F]**

$$\int \frac{1}{\sqrt{a + b \operatorname{arccosh}(cx)}} dx = \int \frac{1}{\sqrt{a + b \operatorname{acosh}(cx)}} dx$$

input `integrate(1/(a+b*acosh(c*x))**(1/2),x)`

output `Integral(1/sqrt(a + b*acosh(c*x)), x)`

**Maxima [F]**

$$\int \frac{1}{\sqrt{a + b \operatorname{arccosh}(cx)}} dx = \int \frac{1}{\sqrt{b \operatorname{arcosh}(cx) + a}} dx$$

input `integrate(1/(a+b*arccosh(c*x))^(1/2),x, algorithm="maxima")`

output `integrate(1/sqrt(b*arccosh(c*x) + a), x)`

**Giac [F]**

$$\int \frac{1}{\sqrt{a + b \operatorname{arccosh}(cx)}} dx = \int \frac{1}{\sqrt{b \operatorname{arcosh}(cx) + a}} dx$$

input `integrate(1/(a+b*arccosh(c*x))^(1/2),x, algorithm="giac")`

output `integrate(1/sqrt(b*arccosh(c*x) + a), x)`

**Mupad [F(-1)]**

Timed out.

$$\int \frac{1}{\sqrt{a + b \operatorname{arccosh}(cx)}} dx = \int \frac{1}{\sqrt{a + b \operatorname{acosh}(cx)}} dx$$

input `int(1/(a + b*acosh(c*x))^(1/2),x)`

output `int(1/(a + b*acosh(c*x))^(1/2), x)`

**Reduce [F]**

$$\int \frac{1}{\sqrt{a + b \operatorname{arccosh}(cx)}} dx = \int \frac{\sqrt{a \operatorname{cosh}(cx) b + a}}{a \operatorname{cosh}(cx) b + a} dx$$

input `int(1/(a+b*acosh(c*x))^(1/2),x)`

output `int(sqrt(acosh(c*x)*b + a)/(acosh(c*x)*b + a),x)`



### 3.12 $\int \frac{1}{(a+b\operatorname{arccosh}(cx))^{3/2}} dx$

Optimal result	112
Mathematica [A] (warning: unable to verify)	112
Rubi [A] (verified)	113
Maple [F]	116
Fricas [F(-2)]	116
Sympy [F]	117
Maxima [F]	117
Giac [F]	117
Mupad [F(-1)]	118
Reduce [F]	118

#### Optimal result

Integrand size = 12, antiderivative size = 120

$$\int \frac{1}{(a + b\operatorname{arccosh}(cx))^{3/2}} dx = -\frac{2\sqrt{-1 + cx}\sqrt{1 + cx}}{bc\sqrt{a + b\operatorname{arccosh}(cx)}} + \frac{e^{a/b}\sqrt{\pi}\operatorname{erf}\left(\frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{b^{3/2}c} + \frac{e^{-\frac{a}{b}}\sqrt{\pi}\operatorname{erfi}\left(\frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{b^{3/2}c}$$

output

```
-2*(c*x-1)^(1/2)*(c*x+1)^(1/2)/b/c/(a+b*arccosh(c*x))^(1/2)+exp(a/b)*Pi^(1/2)*erf((a+b*arccosh(c*x))^(1/2)/b^(1/2))/b^(3/2)/c+Pi^(1/2)*erfi((a+b*arccosh(c*x))^(1/2)/b^(1/2))/b^(3/2)/c/exp(a/b)
```

#### Mathematica [A] (warning: unable to verify)

Time = 0.19 (sec) , antiderivative size = 132, normalized size of antiderivative = 1.10

$$\int \frac{1}{(a + b\operatorname{arccosh}(cx))^{3/2}} dx = \frac{e^{-\frac{a}{b}} \left( -2e^{a/b} \sqrt{\frac{-1+cx}{1+cx}} (1 + cx) - e^{\frac{2a}{b}} \sqrt{\frac{a}{b} + \operatorname{arccosh}(cx)} \Gamma\left(\frac{1}{2}, \frac{a}{b} + \operatorname{arccosh}(cx)\right) \right)}{bc\sqrt{a + b\operatorname{arccosh}(cx)}}$$

input

```
Integrate[(a + b*ArcCosh[c*x])^(-3/2), x]
```

output

```
(-2*E^(a/b)*Sqrt[(-1 + c*x)/(1 + c*x)]*(1 + c*x) - E^((2*a)/b)*Sqrt[a/b + ArcCosh[c*x]]*Gamma[1/2, a/b + ArcCosh[c*x]] + Sqrt[-((a + b*ArcCosh[c*x])/b)]*Gamma[1/2, -((a + b*ArcCosh[c*x])/b)]/(b*c*E^(a/b)*Sqrt[a + b*ArcCosh[c*x]])
```

**Rubi [A] (verified)**

Time = 0.83 (sec) , antiderivative size = 129, normalized size of antiderivative = 1.08, number of steps used = 9, number of rules used = 8,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.667$ , Rules used = {6295, 6368, 3042, 3788, 26, 2611, 2633, 2634}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\begin{aligned}
 & \int \frac{1}{(a + b \operatorname{arccosh}(cx))^{3/2}} dx \\
 & \quad \downarrow \text{6295} \\
 & \frac{2c \int \frac{x}{\sqrt{cx-1}\sqrt{cx+1}\sqrt{a+b\operatorname{arccosh}(cx)}} dx}{b} - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{bc\sqrt{a+b\operatorname{arccosh}(cx)}} \\
 & \quad \downarrow \text{6368} \\
 & \frac{2 \int \frac{\cosh\left(\frac{a}{b} - \frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx))}{b^2c} - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{bc\sqrt{a+b\operatorname{arccosh}(cx)}} \\
 & \quad \downarrow \text{3042} \\
 & -\frac{2\sqrt{cx-1}\sqrt{cx+1}}{bc\sqrt{a+b\operatorname{arccosh}(cx)}} + \frac{2 \int \frac{\sin\left(\frac{ia}{b} - \frac{i(a+b\operatorname{arccosh}(cx))}{b} + \frac{\pi}{2}\right)}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx))}{b^2c} \\
 & \quad \downarrow \text{3788} \\
 & -\frac{2\sqrt{cx-1}\sqrt{cx+1}}{bc\sqrt{a+b\operatorname{arccosh}(cx)}} + \\
 & \frac{2 \left( \frac{1}{2}i \int -\frac{ie^{-\operatorname{arccosh}(cx)}}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx)) - \frac{1}{2}i \int \frac{ie^{\operatorname{arccosh}(cx)}}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx)) \right)}{b^2c}
 \end{aligned}$$

$$\begin{aligned}
& \downarrow 26 \\
& \frac{2 \left( \frac{1}{2} \int \frac{e^{-\operatorname{arccosh}(cx)}}{\sqrt{a+\operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx)) + \frac{1}{2} \int \frac{e^{\operatorname{arccosh}(cx)}}{\sqrt{a+\operatorname{barccosh}(cx)}} d(a + \operatorname{barccosh}(cx)) \right)}{\frac{b^2 c}{2\sqrt{cx-1}\sqrt{cx+1}} bc\sqrt{a + \operatorname{barccosh}(cx)}} \\
& \downarrow 2611 \\
& \frac{2 \left( \int e^{\frac{a}{b} - \frac{a+\operatorname{barccosh}(cx)}{b}} d\sqrt{a + \operatorname{barccosh}(cx)} + \int e^{\frac{a+\operatorname{barccosh}(cx)}{b} - \frac{a}{b}} d\sqrt{a + \operatorname{barccosh}(cx)} \right)}{\frac{b^2 c}{2\sqrt{cx-1}\sqrt{cx+1}} bc\sqrt{a + \operatorname{barccosh}(cx)}} \\
& \downarrow 2633 \\
& \frac{2 \left( \int e^{\frac{a}{b} - \frac{a+\operatorname{barccosh}(cx)}{b}} d\sqrt{a + \operatorname{barccosh}(cx)} + \frac{1}{2} \sqrt{\pi} \sqrt{b} e^{-\frac{a}{b}} \operatorname{erfi} \left( \frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}} \right) \right)}{\frac{b^2 c}{2\sqrt{cx-1}\sqrt{cx+1}} bc\sqrt{a + \operatorname{barccosh}(cx)}} \\
& \downarrow 2634 \\
& \frac{2 \left( \frac{1}{2} \sqrt{\pi} \sqrt{b} e^{a/b} \operatorname{erf} \left( \frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}} \right) + \frac{1}{2} \sqrt{\pi} \sqrt{b} e^{-\frac{a}{b}} \operatorname{erfi} \left( \frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}} \right) \right)}{\frac{b^2 c}{2\sqrt{cx-1}\sqrt{cx+1}} bc\sqrt{a + \operatorname{barccosh}(cx)}}
\end{aligned}$$

input `Int[(a + b*ArcCosh[c*x])^(-3/2),x]`

output `(-2*Sqrt[-1 + c*x]*Sqrt[1 + c*x])/(b*c*Sqrt[a + b*ArcCosh[c*x]]) + (2*((Sqrt[b]*E^(a/b)*Sqrt[Pi]*Erf[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]])/2 + (Sqrt[b]*Sqrt[Pi]*Erfi[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]])/(2*E^(a/b))))/(b^2*c)`

## Definitions of rubi rules used

- rule 26 `Int[(Complex[0, a_])*(Fx_), x_Symbol] := Simp[(Complex[Identity[0], a]) Int[Fx, x], x] /; FreeQ[a, x] && EqQ[a^2, 1]`
- rule 2611 `Int[(F_)^((g_)*((e_) + (f_)*(x_)))/Sqrt[(c_) + (d_)*(x_)], x_Symbol] :> Simp[2/d Subst[Int[F^(g*(e - c*(f/d)) + f*g*(x^2/d)), x], x, Sqrt[c + d*x]], x] /; FreeQ[{F, c, d, e, f, g}, x] && !TrueQ[$UseGamma]`
- rule 2633 `Int[(F_)^((a_) + (b_)*((c_) + (d_)*(x_))^2), x_Symbol] := Simp[F^a*Sqrt[Pi]*(Erfi[(c + d*x)*Rt[b*Log[F], 2]]/(2*d*Rt[b*Log[F], 2])), x] /; FreeQ[{F, a, b, c, d}, x] && PosQ[b]`
- rule 2634 `Int[(F_)^((a_) + (b_)*((c_) + (d_)*(x_))^2), x_Symbol] := Simp[F^a*Sqrt[Pi]*(Erf[(c + d*x)*Rt[(-b)*Log[F], 2]]/(2*d*Rt[(-b)*Log[F], 2])), x] /; FreeQ[{F, a, b, c, d}, x] && NegQ[b]`
- rule 3042 `Int[u_, x_Symbol] := Int[DeactivateTrig[u, x], x] /; FunctionOfTrigOfLinearQ[u, x]`
- rule 3788 `Int[((c_) + (d_)*(x_))^(m_)*sin[(e_) + Pi*(k_) + (f_)*(x_)], x_Symbol] := Simp[I/2 Int[(c + d*x)^m/(E^(I*k*Pi)*E^(I*(e + f*x))), x], x] - Simp[I/2 Int[(c + d*x)^m*E^(I*k*Pi)*E^(I*(e + f*x)), x], x] /; FreeQ[{c, d, e, f, m}, x] && IntegerQ[2*k]`
- rule 6295 `Int[((a_) + ArcCosh[(c_)*(x_)])*(b_)^(n_), x_Symbol] := Simp[Sqrt[1 + c*x]*Sqrt[-1 + c*x]*((a + b*ArcCosh[c*x])^(n + 1)/(b*c*(n + 1))), x] - Simp[c/(b*(n + 1)) Int[x*((a + b*ArcCosh[c*x])^(n + 1)/(Sqrt[1 + c*x]*Sqrt[-1 + c*x])), x], x] /; FreeQ[{a, b, c}, x] && LtQ[n, -1]`

rule 6368

```
Int[((a_.) + ArcCosh[(c_.)*(x_.)]*(b_.))^(n_.)*(x_)^(m_.)*((d1_) + (e1_.)*(x_)^(p_.))*((d2_) + (e2_.)*(x_.))^(p_.), x_Symbol] := Simp[(1/(b*c^(m + 1)))*Simp[(d1 + e1*x)^p/(1 + c*x)^p]*Simp[(d2 + e2*x)^p/(-1 + c*x)^p] Subst[Int[x^n*Cosh[-a/b + x/b]^m*Sinh[-a/b + x/b]^(2*p + 1), x], x, a + b*ArcCosh[c*x]], x] /; FreeQ[{a, b, c, d1, e1, d2, e2, n}, x] && EqQ[e1, c*d1] && EqQ[e2, (-c)*d2] && IGtQ[p + 3/2, 0] && IGtQ[m, 0]
```

**Maple [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{\frac{3}{2}}} dx$$

input `int(1/(a+b*arccosh(c*x))^(3/2),x)`

output `int(1/(a+b*arccosh(c*x))^(3/2),x)`

**Fricas [F(-2)]**

Exception generated.

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{3/2}} dx = \text{Exception raised: TypeError}$$

input `integrate(1/(a+b*arccosh(c*x))^(3/2),x, algorithm="fricas")`

output `Exception raised: TypeError >> Error detected within library code: integrate: implementation incomplete (constant residues)`

**Sympy [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{3/2}} dx = \int \frac{1}{(a + b \operatorname{acosh}(cx))^{\frac{3}{2}}} dx$$

input `integrate(1/(a+b*acosh(c*x))**(3/2), x)`

output `Integral((a + b*acosh(c*x))**(-3/2), x)`

**Maxima [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{3/2}} dx = \int \frac{1}{(b \operatorname{arcosh}(cx) + a)^{\frac{3}{2}}} dx$$

input `integrate(1/(a+b*arccosh(c*x))^(3/2), x, algorithm="maxima")`

output `integrate((b*arccosh(c*x) + a)^(-3/2), x)`

**Giac [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{3/2}} dx = \int \frac{1}{(b \operatorname{arcosh}(cx) + a)^{\frac{3}{2}}} dx$$

input `integrate(1/(a+b*arccosh(c*x))^(3/2), x, algorithm="giac")`

output `integrate((b*arccosh(c*x) + a)^(-3/2), x)`

**Mupad [F(-1)]**

Timed out.

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{3/2}} dx = \int \frac{1}{(a + b \operatorname{acosh}(cx))^{3/2}} dx$$

input `int(1/(a + b*acosh(c*x))^(3/2), x)`output `int(1/(a + b*acosh(c*x))^(3/2), x)`**Reduce [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{3/2}} dx = \text{Too large to display}$$

input `int(1/(a+b*acosh(c*x))^(3/2), x)`

output

```
(2*sqrt(c*x + 1)*sqrt(c*x - 1)*sqrt(acosh(c*x)*b + a)*acosh(c*x) - acosh(c
*x)*int((sqrt(acosh(c*x)*b + a)*acosh(c*x)*x**2)/(acosh(c*x)**2*b**2*c**2*
x**2 - acosh(c*x)**2*b**2 + 2*acosh(c*x)*a*b*c**2*x**2 - 2*acosh(c*x)*a*b
+ a**2*c**2*x**2 - a**2),x)*b**2*c**3 + acosh(c*x)*int((sqrt(acosh(c*x)*b
+ a)*acosh(c*x))/(acosh(c*x)**2*b**2*c**2*x**2 - acosh(c*x)**2*b**2 + 2*ac
osh(c*x)*a*b*c**2*x**2 - 2*acosh(c*x)*a*b + a**2*c**2*x**2 - a**2),x)*b**2
*c - 2*acosh(c*x)*int((sqrt(c*x + 1)*sqrt(c*x - 1)*sqrt(acosh(c*x)*b + a)*
acosh(c*x)*x)/(acosh(c*x)**2*b**2*c**2*x**2 - acosh(c*x)**2*b**2 + 2*acosh
(c*x)*a*b*c**2*x**2 - 2*acosh(c*x)*a*b + a**2*c**2*x**2 - a**2),x)*a*b*c**
2 - 2*acosh(c*x)*int((sqrt(c*x + 1)*sqrt(c*x - 1)*sqrt(acosh(c*x)*b + a)*a
cosh(c*x)**2*x)/(acosh(c*x)**2*b**2*c**2*x**2 - acosh(c*x)**2*b**2 + 2*aco
sh(c*x)*a*b*c**2*x**2 - 2*acosh(c*x)*a*b + a**2*c**2*x**2 - a**2),x)*b**2*
c**2 - int((sqrt(acosh(c*x)*b + a)*acosh(c*x)*x**2)/(acosh(c*x)**2*b**2*c*
**2*x**2 - acosh(c*x)**2*b**2 + 2*acosh(c*x)*a*b*c**2*x**2 - 2*acosh(c*x)*a
*b + a**2*c**2*x**2 - a**2),x)*a*b*c**3 + int((sqrt(acosh(c*x)*b + a)*acos
h(c*x))/(acosh(c*x)**2*b**2*c**2*x**2 - acosh(c*x)**2*b**2 + 2*acosh(c*x)*
a*b*c**2*x**2 - 2*acosh(c*x)*a*b + a**2*c**2*x**2 - a**2),x)*a*b*c - 2*int
((sqrt(c*x + 1)*sqrt(c*x - 1)*sqrt(acosh(c*x)*b + a)*acosh(c*x)*x)/(acosh(
c*x)**2*b**2*c**2*x**2 - acosh(c*x)**2*b**2 + 2*acosh(c*x)*a*b*c**2*x**2 -
2*acosh(c*x)*a*b + a**2*c**2*x**2 - a**2),x)*a**2*c**2 - 2*int((sqrt(c...
```



### 3.13 $\int \frac{1}{(a+b\operatorname{arccosh}(cx))^{5/2}} dx$

Optimal result	120
Mathematica [A] (warning: unable to verify)	120
Rubi [C] (verified)	121
Maple [F]	125
Fricas [F(-2)]	125
Sympy [F]	125
Maxima [F]	126
Giac [F]	126
Mupad [F(-1)]	126
Reduce [F]	127

#### Optimal result

Integrand size = 12, antiderivative size = 148

$$\int \frac{1}{(a + b\operatorname{arccosh}(cx))^{5/2}} dx = -\frac{2\sqrt{-1 + cx}\sqrt{1 + cx}}{3bc(a + b\operatorname{arccosh}(cx))^{3/2}} - \frac{4x}{3b^2\sqrt{a + b\operatorname{arccosh}(cx)}} - \frac{2e^{a/b}\sqrt{\pi}\operatorname{erf}\left(\frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{3b^{5/2}c} + \frac{2e^{-\frac{a}{b}}\sqrt{\pi}\operatorname{erfi}\left(\frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{3b^{5/2}c}$$

output

```
-2/3*(c*x-1)^(1/2)*(c*x+1)^(1/2)/b/c/(a+b*arccosh(c*x))^(3/2)-4/3*x/b^2/(a+b*arccosh(c*x))^(1/2)-2/3*exp(a/b)*Pi^(1/2)*erf((a+b*arccosh(c*x))^(1/2)/b^(1/2))/b^(5/2)/c+2/3*Pi^(1/2)*erfi((a+b*arccosh(c*x))^(1/2)/b^(1/2))/b^(5/2)/c/exp(a/b)
```

#### Mathematica [A] (warning: unable to verify)

Time = 0.72 (sec) , antiderivative size = 192, normalized size of antiderivative = 1.30

$$\int \frac{1}{(a + b\operatorname{arccosh}(cx))^{5/2}} dx = \frac{e^{-\frac{a+b\operatorname{arccosh}(cx)}{b}} \left( 2e^{\frac{2a}{b} + \operatorname{arccosh}(cx)} \sqrt{\frac{a}{b} + \operatorname{arccosh}(cx)} (a + b\operatorname{arccosh}(cx)) \Gamma\left(\frac{1}{2}, \frac{a}{b} + \operatorname{arccosh}(cx)\right) \right)}{\dots}$$

input

```
Integrate[(a + b*ArcCosh[c*x])^(-5/2), x]
```

output

```
(2*E^((2*a)/b + ArcCosh[c*x])*Sqrt[a/b + ArcCosh[c*x]]*(a + b*ArcCosh[c*x])
)*Gamma[1/2, a/b + ArcCosh[c*x]] - 2*(E^(a/b)*(b*E^ArcCosh[c*x]*Sqrt[(-1 +
c*x)/(1 + c*x)]*(1 + c*x) + (1 + E^(2*ArcCosh[c*x]))*(a + b*ArcCosh[c*x])
) + b*E^ArcCosh[c*x]*(-((a + b*ArcCosh[c*x])/b))^(3/2)*Gamma[1/2, -((a + b
*ArcCosh[c*x])/b)])/(3*b^2*c*E^((a + b*ArcCosh[c*x])/b)*(a + b*ArcCosh[c*
x])^(3/2))
```

### Rubi [C] (verified)

Result contains complex when optimal does not.

Time = 1.35 (sec) , antiderivative size = 167, normalized size of antiderivative = 1.13, number of steps used = 11, number of rules used = 10,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.833$ , Rules used = {6295, 6366, 6296, 25, 3042, 26, 3789, 2611, 2633, 2634}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\begin{aligned}
 & \int \frac{1}{(a + \operatorname{barccosh}(cx))^{5/2}} dx \\
 & \quad \downarrow \text{6295} \\
 & \frac{2c \int \frac{x}{\sqrt{cx-1}\sqrt{cx+1}(a+\operatorname{barccosh}(cx))^{3/2}} dx}{3b} - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{3bc(a + \operatorname{barccosh}(cx))^{3/2}} \\
 & \quad \downarrow \text{6366} \\
 & \frac{2c \left( \frac{2 \int \frac{1}{\sqrt{a+b\operatorname{arccosh}(cx)}} dx}{bc} - \frac{2x}{bc\sqrt{a+b\operatorname{arccosh}(cx)}} \right)}{3b} - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{3bc(a + \operatorname{barccosh}(cx))^{3/2}} \\
 & \quad \downarrow \text{6296} \\
 & \frac{2c \left( \frac{2 \int -\frac{\sinh\left(\frac{a}{b} - \frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+\operatorname{barccosh}(cx))}{b^2c^2} - \frac{2x}{bc\sqrt{a+b\operatorname{arccosh}(cx)}} \right)}{3b} - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{3bc(a + \operatorname{barccosh}(cx))^{3/2}} \\
 & \quad \downarrow \text{25}
 \end{aligned}$$

$$\begin{array}{c}
 2c \left( -\frac{2 \int \frac{\sinh\left(\frac{a}{b} - \frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx))}{b^2 c^2} - \frac{2x}{bc\sqrt{a+b\operatorname{arccosh}(cx)}} \right) \\
 \hline
 \frac{3b}{2\sqrt{cx-1}\sqrt{cx+1}} \\
 \frac{3bc(a+b\operatorname{arccosh}(cx))^{3/2}}{3042} \\
 \downarrow \\
 -\frac{2\sqrt{cx-1}\sqrt{cx+1}}{3bc(a+b\operatorname{arccosh}(cx))^{3/2}} + \\
 2c \left( -\frac{2x}{bc\sqrt{a+b\operatorname{arccosh}(cx)}} - \frac{2 \int \frac{i \sin\left(\frac{ia}{b} - \frac{i(a+b\operatorname{arccosh}(cx))}{b}\right)}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx))}{b^2 c^2} \right) \\
 \hline
 \frac{3b}{26} \\
 \downarrow \\
 -\frac{2\sqrt{cx-1}\sqrt{cx+1}}{3bc(a+b\operatorname{arccosh}(cx))^{3/2}} + \\
 2c \left( -\frac{2x}{bc\sqrt{a+b\operatorname{arccosh}(cx)}} + \frac{2i \int \frac{\sin\left(\frac{ia}{b} - \frac{i(a+b\operatorname{arccosh}(cx))}{b}\right)}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx))}{b^2 c^2} \right) \\
 \hline
 \frac{3b}{3789} \\
 \downarrow \\
 -\frac{2\sqrt{cx-1}\sqrt{cx+1}}{3bc(a+b\operatorname{arccosh}(cx))^{3/2}} + \\
 2c \left( -\frac{2x}{bc\sqrt{a+b\operatorname{arccosh}(cx)}} + \frac{2i \left( \frac{1}{2} \int \frac{e^{-\operatorname{arccosh}(cx)}}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx)) - \frac{1}{2} \int \frac{e^{\operatorname{arccosh}(cx)}}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx)) \right)}{b^2 c^2} \right) \\
 \hline
 \frac{3b}{2611} \\
 \downarrow \\
 -\frac{2\sqrt{cx-1}\sqrt{cx+1}}{3bc(a+b\operatorname{arccosh}(cx))^{3/2}} + \\
 2c \left( -\frac{2x}{bc\sqrt{a+b\operatorname{arccosh}(cx)}} + \frac{2i \left( i \int e^{\frac{a}{b} - \frac{a+b\operatorname{arccosh}(cx)}{b}} d\sqrt{a+b\operatorname{arccosh}(cx)} - i \int e^{\frac{a+b\operatorname{arccosh}(cx)}{b} - \frac{a}{b}} d\sqrt{a+b\operatorname{arccosh}(cx)} \right)}{b^2 c^2} \right) \\
 \hline
 \frac{3b}{
 \end{array}$$

$$\begin{array}{c}
\downarrow 2633 \\
\frac{2\sqrt{cx-1}\sqrt{cx+1}}{3bc(a+\operatorname{barccosh}(cx))^{3/2}} + \\
2c \left( -\frac{2x}{bc\sqrt{a+\operatorname{barccosh}(cx)}} + \frac{2i \left( i \int e^{\frac{a}{b} - \frac{a+\operatorname{barccosh}(cx)}{b}} d\sqrt{a+\operatorname{barccosh}(cx)} - \frac{1}{2} i \sqrt{\pi} \sqrt{b} e^{-\frac{a}{b}} \operatorname{erfi} \left( \frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}} \right) \right)}{b^2 c^2} \right) \\
\hline
3b \\
\downarrow 2634 \\
\frac{2\sqrt{cx-1}\sqrt{cx+1}}{3bc(a+\operatorname{barccosh}(cx))^{3/2}} + \\
2c \left( -\frac{2x}{bc\sqrt{a+\operatorname{barccosh}(cx)}} + \frac{2i \left( \frac{1}{2} i \sqrt{\pi} \sqrt{b} e^{a/b} \operatorname{erf} \left( \frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}} \right) - \frac{1}{2} i \sqrt{\pi} \sqrt{b} e^{-\frac{a}{b}} \operatorname{erfi} \left( \frac{\sqrt{a+\operatorname{barccosh}(cx)}}{\sqrt{b}} \right) \right)}{b^2 c^2} \right) \\
\hline
3b
\end{array}$$

input `Int[(a + b*ArcCosh[c*x])^(-5/2), x]`

output `(-2*Sqrt[-1 + c*x]*Sqrt[1 + c*x])/(3*b*c*(a + b*ArcCosh[c*x])^(3/2)) + (2*c*((-2*x)/(b*c*Sqrt[a + b*ArcCosh[c*x]]) + ((2*I)*((I/2)*Sqrt[b]*E^(a/b)*Sqrt[Pi]*Erf[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]] - ((I/2)*Sqrt[b]*Sqrt[Pi]*Erfi[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]])/E^(a/b)))/(b^2*c^2))/(3*b)`

### Defintions of rubi rules used

rule 25 `Int[-(Fx_), x_Symbol] :> Simp[Identity[-1] Int[Fx, x], x]`

rule 26 `Int[(Complex[0, a_])*(Fx_), x_Symbol] :> Simp[(Complex[Identity[0], a]) Int[Fx, x], x] /; FreeQ[a, x] && EqQ[a^2, 1]`

rule 2611 `Int[(F_)^((g_)*((e_)+(f_)*(x_)))/Sqrt[(c_)+(d_)*(x_)], x_Symbol] :> Simp[2/d Subst[Int[F^(g*(e - c*(f/d)) + f*g*(x^2/d)), x], x, Sqrt[c + d*x]], x] /; FreeQ[{F, c, d, e, f, g}, x] && !TrueQ[$UseGamma]`

rule 2633  $\text{Int}[(F_)^((a_.) + (b_.)*((c_.) + (d_.)*(x_))^{2}), x\_Symbol] \rightarrow \text{Simp}[F^a*\text{Sqrt}[\text{Pi}]*(\text{Erfi}[(c + d*x)*\text{Rt}[b*\text{Log}[F], 2]]/(2*d*\text{Rt}[b*\text{Log}[F], 2])), x] /;$   $\text{FreeQ}\{F, a, b, c, d\}, x\} \ \&\& \ \text{PosQ}[b]$

rule 2634  $\text{Int}[(F_)^((a_.) + (b_.)*((c_.) + (d_.)*(x_))^{2}), x\_Symbol] \rightarrow \text{Simp}[F^a*\text{Sqrt}[\text{Pi}]*(\text{Erf}[(c + d*x)*\text{Rt}[(-b)*\text{Log}[F], 2]]/(2*d*\text{Rt}[(-b)*\text{Log}[F], 2])), x] /;$   $\text{FreeQ}\{F, a, b, c, d\}, x\} \ \&\& \ \text{NegQ}[b]$

rule 3042  $\text{Int}[u_, x\_Symbol] \rightarrow \text{Int}[\text{DeactivateTrig}[u, x], x] /;$   $\text{FunctionOfTrigOfLinearQ}[u, x]$

rule 3789  $\text{Int}[((c_.) + (d_.)*(x_))^{(m_.)*\sin[(e_.) + (f_.)*(x_)]}, x\_Symbol] \rightarrow \text{Simp}[I/2 \ \text{Int}[(c + d*x)^m/E^{(I*(e + f*x))}, x], x] - \text{Simp}[I/2 \ \text{Int}[(c + d*x)^m*E^{(I*(e + f*x))}, x], x] /;$   $\text{FreeQ}\{c, d, e, f, m\}, x\}$

rule 6295  $\text{Int}[((a_.) + \text{ArcCosh}[(c_.)*(x_)]*(b_.))^{(n_)}, x\_Symbol] \rightarrow \text{Simp}[\text{Sqrt}[1 + c*x]*\text{Sqrt}[-1 + c*x]*((a + b*\text{ArcCosh}[c*x])^{(n + 1)/(b*c*(n + 1))}), x] - \text{Simp}[c/(b*(n + 1)) \ \text{Int}[x*((a + b*\text{ArcCosh}[c*x])^{(n + 1)/(b*c*(n + 1))}/(\text{Sqrt}[1 + c*x]*\text{Sqrt}[-1 + c*x])), x], x] /;$   $\text{FreeQ}\{a, b, c\}, x\} \ \&\& \ \text{LtQ}[n, -1]$

rule 6296  $\text{Int}[((a_.) + \text{ArcCosh}[(c_.)*(x_)]*(b_.))^{(n_)}, x\_Symbol] \rightarrow \text{Simp}[1/(b*c) \ \text{Subst}[\text{Int}[x^n*\text{Sinh}[-a/b + x/b], x], x, a + b*\text{ArcCosh}[c*x]], x] /;$   $\text{FreeQ}\{a, b, c, n\}, x\}$

rule 6366  $\text{Int}[(((a_.) + \text{ArcCosh}[(c_.)*(x_)]*(b_.))^{(n_)}*((f_.)*(x_))^{(m_.)})/(\text{Sqrt}[(d1_) + (e1_.)*(x_)]*\text{Sqrt}[(d2_) + (e2_.)*(x_)]), x\_Symbol] \rightarrow \text{Simp}[(f*x)^m*((a + b*\text{ArcCosh}[c*x])^{(n + 1)/(b*c*(n + 1))})*\text{Simp}[\text{Sqrt}[1 + c*x]/\text{Sqrt}[d1 + e1*x]]*\text{Simp}[\text{Sqrt}[-1 + c*x]/\text{Sqrt}[d2 + e2*x]], x] - \text{Simp}[f*(m/(b*c*(n + 1)))*\text{Simp}[\text{Sqrt}[1 + c*x]/\text{Sqrt}[d1 + e1*x]]*\text{Simp}[\text{Sqrt}[-1 + c*x]/\text{Sqrt}[d2 + e2*x]] \ \text{Int}[(f*x)^{(m - 1)}*(a + b*\text{ArcCosh}[c*x])^{(n + 1)}, x], x] /;$   $\text{FreeQ}\{a, b, c, d1, e1, d2, e2, f, m\}, x\} \ \&\& \ \text{EqQ}[e1, c*d1] \ \&\& \ \text{EqQ}[e2, (-c)*d2] \ \&\& \ \text{LtQ}[n, -1]$

**Maple [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{\frac{5}{2}}} dx$$

input `int(1/(a+b*arccosh(c*x))^(5/2),x)`

output `int(1/(a+b*arccosh(c*x))^(5/2),x)`

**Fricas [F(-2)]**

Exception generated.

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{5/2}} dx = \text{Exception raised: TypeError}$$

input `integrate(1/(a+b*arccosh(c*x))^(5/2),x, algorithm="fricas")`

output `Exception raised: TypeError >> Error detected within library code: integrate: implementation incomplete (constant residues)`

**Sympy [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{5/2}} dx = \int \frac{1}{(a + b \operatorname{acosh}(cx))^{\frac{5}{2}}} dx$$

input `integrate(1/(a+b*acosh(c*x))**(5/2),x)`

output `Integral((a + b*acosh(c*x))**(-5/2), x)`

**Maxima [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{5/2}} dx = \int \frac{1}{(b \operatorname{arcosh}(cx) + a)^{5/2}} dx$$

input `integrate(1/(a+b*arccosh(c*x))^(5/2),x, algorithm="maxima")`

output `integrate((b*arccosh(c*x) + a)^(-5/2), x)`

**Giac [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{5/2}} dx = \int \frac{1}{(b \operatorname{arcosh}(cx) + a)^{5/2}} dx$$

input `integrate(1/(a+b*arccosh(c*x))^(5/2),x, algorithm="giac")`

output `integrate((b*arccosh(c*x) + a)^(-5/2), x)`

**Mupad [F(-1)]**

Timed out.

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{5/2}} dx = \int \frac{1}{(a + b \operatorname{acosh}(cx))^{5/2}} dx$$

input `int(1/(a + b*acosh(c*x))^(5/2),x)`

output `int(1/(a + b*acosh(c*x))^(5/2), x)`

## Reduce [F]

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{5/2}} dx = \text{too large to display}$$

input `int(1/(a+b*acosh(c*x))^(5/2),x)`

output

```
(acosh(c*x)**2*int((sqrt(acosh(c*x)*b + a)*acosh(c*x)*x**2)/(acosh(c*x)**3
*b**3*c**2*x**2 - acosh(c*x)**3*b**3 + 3*acosh(c*x)**2*a*b**2*c**2*x**2 -
3*acosh(c*x)**2*a*b**2 + 3*acosh(c*x)*a**2*b*c**2*x**2 - 3*acosh(c*x)*a**2
*b + a**3*c**2*x**2 - a**3),x)*b**3*c**3 - acosh(c*x)**2*int((sqrt(acosh(c
*x)*b + a)*acosh(c*x))/(acosh(c*x)**3*b**3*c**2*x**2 - acosh(c*x)**3*b**3
+ 3*acosh(c*x)**2*a*b**2*c**2*x**2 - 3*acosh(c*x)**2*a*b**2 + 3*acosh(c*x)
*a**2*b*c**2*x**2 - 3*acosh(c*x)*a**2*b + a**3*c**2*x**2 - a**3),x)*b**3*c
- 2*acosh(c*x)**2*int((sqrt(c*x + 1)*sqrt(c*x - 1)*sqrt(acosh(c*x)*b + a)
*acosh(c*x)*x)/(acosh(c*x)**3*b**3*c**2*x**2 - acosh(c*x)**3*b**3 + 3*acos
h(c*x)**2*a*b**2*c**2*x**2 - 3*acosh(c*x)**2*a*b**2 + 3*acosh(c*x)*a**2*b*
c**2*x**2 - 3*acosh(c*x)*a**2*b + a**3*c**2*x**2 - a**3),x)*a*b**2*c**2 -
2*acosh(c*x)**2*int((sqrt(c*x + 1)*sqrt(c*x - 1)*sqrt(acosh(c*x)*b + a)*ac
osh(c*x)**2*x)/(acosh(c*x)**3*b**3*c**2*x**2 - acosh(c*x)**3*b**3 + 3*acos
h(c*x)**2*a*b**2*c**2*x**2 - 3*acosh(c*x)**2*a*b**2 + 3*acosh(c*x)*a**2*b*
c**2*x**2 - 3*acosh(c*x)*a**2*b + a**3*c**2*x**2 - a**3),x)*b**3*c**2 + 2*
sqrt(c*x + 1)*sqrt(c*x - 1)*sqrt(acosh(c*x)*b + a)*acosh(c*x) + 2*acosh(c*
x)*int((sqrt(acosh(c*x)*b + a)*acosh(c*x)*x**2)/(acosh(c*x)**3*b**3*c**2*x
**2 - acosh(c*x)**3*b**3 + 3*acosh(c*x)**2*a*b**2*c**2*x**2 - 3*acosh(c*x)
**2*a*b**2 + 3*acosh(c*x)*a**2*b*c**2*x**2 - 3*acosh(c*x)*a**2*b + a**3*c*
**2*x**2 - a**3),x)*a*b**2*c**3 - 2*acosh(c*x)*int((sqrt(acosh(c*x)*b + ...
```



### 3.14 $\int \frac{1}{(a+b\operatorname{arccosh}(cx))^{7/2}} dx$

Optimal result	128
Mathematica [A] (warning: unable to verify)	129
Rubi [A] (verified)	129
Maple [F]	134
Fricas [F(-2)]	134
Sympy [F(-1)]	135
Maxima [F]	135
Giac [F]	135
Mupad [F(-1)]	136
Reduce [F]	136

#### Optimal result

Integrand size = 12, antiderivative size = 188

$$\int \frac{1}{(a+b\operatorname{arccosh}(cx))^{7/2}} dx = -\frac{2\sqrt{-1+cx}\sqrt{1+cx}}{5bc(a+b\operatorname{arccosh}(cx))^{5/2}} - \frac{4x}{15b^2(a+b\operatorname{arccosh}(cx))^{3/2}} - \frac{8\sqrt{-1+cx}\sqrt{1+cx}}{15b^3c\sqrt{a+b\operatorname{arccosh}(cx)}} + \frac{4e^{a/b}\sqrt{\pi}\operatorname{erf}\left(\frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{15b^{7/2}c} + \frac{4e^{-\frac{a}{b}}\sqrt{\pi}\operatorname{erfi}\left(\frac{\sqrt{a+b\operatorname{arccosh}(cx)}}{\sqrt{b}}\right)}{15b^{7/2}c}$$

output

```
-2/5*(c*x-1)^(1/2)*(c*x+1)^(1/2)/b/c/(a+b*arccosh(c*x))^(5/2)-4/15*x/b^2/(a+b*arccosh(c*x))^(3/2)-8/15*(c*x-1)^(1/2)*(c*x+1)^(1/2)/b^3/c/(a+b*arccosh(c*x))^(1/2)+4/15*exp(a/b)*Pi^(1/2)*erf((a+b*arccosh(c*x))^(1/2)/b^(1/2))/b^(7/2)/c+4/15*Pi^(1/2)*erfi((a+b*arccosh(c*x))^(1/2)/b^(1/2))/b^(7/2)/c/exp(a/b)
```

**Mathematica [A] (warning: unable to verify)**

Time = 0.51 (sec) , antiderivative size = 214, normalized size of antiderivative = 1.14

$$\int \frac{1}{(a + \operatorname{barccosh}(cx))^{7/2}} dx = \frac{-6\sqrt{\frac{-1+cx}{1+cx}}(1+cx) - \frac{2e^{-\operatorname{arccosh}(cx)}(a+\operatorname{barccosh}(cx))(-2a+b-2\operatorname{barccosh}(cx)+2e^{\frac{a}{b}+a}}}{(a+b\operatorname{barccosh}(cx))^{5/2}}}{(a+b\operatorname{barccosh}(cx))^{5/2}}$$

input `Integrate[(a + b*ArcCosh[c*x])^(-7/2), x]`output

```
(-6*Sqrt[(-1 + c*x)/(1 + c*x)]*(1 + c*x) - (2*(a + b*ArcCosh[c*x])*(-2*a +
b - 2*b*ArcCosh[c*x] + 2*E^(a/b + ArcCosh[c*x])*Sqrt[a/b + ArcCosh[c*x]]*
(a + b*ArcCosh[c*x])*Gamma[1/2, a/b + ArcCosh[c*x]]))/(b^2*E^ArcCosh[c*x])
- (2*(a + b*ArcCosh[c*x])*(E^(a/b + ArcCosh[c*x])*(2*a + b + 2*b*ArcCosh[
c*x]) + 2*b*(-((a + b*ArcCosh[c*x])/b))^(3/2)*Gamma[1/2, -((a + b*ArcCosh[
c*x])/b)]))/(b^2*E^(a/b))/(15*b*c*(a + b*ArcCosh[c*x])^(5/2))
```

**Rubi [A] (verified)**Time = 2.12 (sec) , antiderivative size = 212, normalized size of antiderivative = 1.13, number of steps used = 11, number of rules used = 10,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.833$ , Rules used = {6295, 6366, 6295, 6368, 3042, 3788, 26, 2611, 2633, 2634}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\int \frac{1}{(a + \operatorname{barccosh}(cx))^{7/2}} dx$$

$$\downarrow 6295$$

$$\frac{2c \int \frac{x}{\sqrt{cx-1}\sqrt{cx+1}(a+\operatorname{barccosh}(cx))^{5/2}} dx}{5b} - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{5bc(a + \operatorname{barccosh}(cx))^{5/2}}$$

$$\downarrow 6366$$

$$\frac{2c \left( \frac{2 \int \frac{1}{(a+\operatorname{barccosh}(cx))^{3/2}} dx}{3bc} - \frac{2x}{3bc(a+\operatorname{barccosh}(cx))^{3/2}} \right)}{5b} - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{5bc(a + \operatorname{barccosh}(cx))^{5/2}}$$

$$\begin{array}{c}
 \downarrow \text{6295} \\
 2c \left( \frac{2 \int \frac{\frac{x}{\sqrt{cx-1}\sqrt{cx+1}\sqrt{a+b\operatorname{arccosh}(cx)}}{b} dx}{3bc} - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{bc\sqrt{a+b\operatorname{arccosh}(cx)}}}{3bc(a+b\operatorname{arccosh}(cx))^{3/2}} \right) \\
 \hline
 \frac{5b}{2\sqrt{cx-1}\sqrt{cx+1}} \\
 \frac{5bc(a+b\operatorname{arccosh}(cx))^{5/2}}{\phantom{2\sqrt{cx-1}\sqrt{cx+1}}} \\
 \downarrow \text{6368} \\
 2c \left( \frac{2 \int \frac{\frac{\cosh\left(\frac{a}{b} - \frac{a+b\operatorname{arccosh}(cx)}{b}\right)}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx))}{b^2c} - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{bc\sqrt{a+b\operatorname{arccosh}(cx)}}}{3bc} \right) - \frac{2x}{3bc(a+b\operatorname{arccosh}(cx))^{3/2}} \\
 \hline
 \frac{5b}{2\sqrt{cx-1}\sqrt{cx+1}} \\
 \frac{5bc(a+b\operatorname{arccosh}(cx))^{5/2}}{\phantom{2\sqrt{cx-1}\sqrt{cx+1}}} \\
 \downarrow \text{3042} \\
 -\frac{2\sqrt{cx-1}\sqrt{cx+1}}{5bc(a+b\operatorname{arccosh}(cx))^{5/2}} + \\
 2c \left( -\frac{2x}{3bc(a+b\operatorname{arccosh}(cx))^{3/2}} + \frac{2 \int \frac{\frac{\sin\left(\frac{ia}{b} - \frac{i(a+b\operatorname{arccosh}(cx))}{b} + \frac{\pi}{2}\right)}{\sqrt{a+b\operatorname{arccosh}(cx)}} d(a+b\operatorname{arccosh}(cx))}{b^2c} - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{bc\sqrt{a+b\operatorname{arccosh}(cx)}}}{3bc} \right) \\
 \hline
 \frac{5b}{\phantom{2\sqrt{cx-1}\sqrt{cx+1}}} \\
 \downarrow \text{3788}
 \end{array}$$

$$2c \left( -\frac{2x}{3bc(a+\operatorname{barccosh}(cx))^{3/2}} + \frac{2\sqrt{cx-1}\sqrt{cx+1}}{5bc(a+\operatorname{barccosh}(cx))^{5/2}} + \frac{2 \left( -\frac{2\sqrt{cx-1}\sqrt{cx+1}}{bc\sqrt{a+\operatorname{barccosh}(cx)}} + \frac{\frac{1}{2} \int \frac{e^{-\operatorname{arccosh}(cx)}}{\sqrt{a+\operatorname{barccosh}(cx)}} d(a+\operatorname{barccosh}(cx)) - \frac{1}{2} \int \frac{e^{\operatorname{arccosh}(cx)}}{\sqrt{a+\operatorname{barccosh}(cx)}} d(a+\operatorname{barccosh}(cx))}{b^2c} \right)}{3bc} \right)$$

5b

↓ 26

$$2c \left( \frac{2 \left( \frac{\frac{1}{2} \int \frac{e^{-\operatorname{arccosh}(cx)}}{\sqrt{a+\operatorname{barccosh}(cx)}} d(a+\operatorname{barccosh}(cx)) + \frac{1}{2} \int \frac{e^{\operatorname{arccosh}(cx)}}{\sqrt{a+\operatorname{barccosh}(cx)}} d(a+\operatorname{barccosh}(cx))}{b^2c} - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{bc\sqrt{a+\operatorname{barccosh}(cx)}} \right)}{3bc} - \frac{2x}{3bc(a+\operatorname{barccosh}(cx))^{3/2}} \right)$$

5b

$$\frac{2\sqrt{cx-1}\sqrt{cx+1}}{5bc(a+\operatorname{barccosh}(cx))^{5/2}}$$

↓ 2611

$$2c \left( \frac{2 \left( \frac{\int \frac{e^{\frac{a}{b} - \frac{a+\operatorname{barccosh}(cx)}{b}} d\sqrt{a+\operatorname{barccosh}(cx)} + \int \frac{e^{\frac{a+\operatorname{barccosh}(cx)}{b} - \frac{a}{b}} d\sqrt{a+\operatorname{barccosh}(cx)}}{b^2c} - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{bc\sqrt{a+\operatorname{barccosh}(cx)}} \right)}{3bc} - \frac{2x}{3bc(a+\operatorname{barccosh}(cx))^{3/2}} \right)$$

5b

$$\frac{2\sqrt{cx-1}\sqrt{cx+1}}{5bc(a+\operatorname{barccosh}(cx))^{5/2}}$$

↓ 2633

$$\begin{aligned}
 & 2c \left( \frac{2 \left( \frac{e^{\frac{a}{b} - \frac{a + \operatorname{arccosh}(cx)}{b}} d\sqrt{a + b \operatorname{arccosh}(cx)} + \frac{1}{2} \sqrt{\pi} \sqrt{b} e^{-\frac{a}{b}} \operatorname{erfi} \left( \frac{\sqrt{a + b \operatorname{arccosh}(cx)}}{\sqrt{b}} \right)}{b^2 c} \right) - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{bc\sqrt{a + b \operatorname{arccosh}(cx)}}}{3bc} \right) - \frac{2x}{3bc(a + b \operatorname{arccosh}(cx))} \\
 & \frac{2\sqrt{cx-1}\sqrt{cx+1}}{5bc(a + b \operatorname{arccosh}(cx))^{5/2}} \quad 5b \\
 & \quad \downarrow \text{2634} \\
 & 2c \left( \frac{2 \left( \frac{\frac{1}{2} \sqrt{\pi} \sqrt{b} e^{a/b} \operatorname{erf} \left( \frac{\sqrt{a + b \operatorname{arccosh}(cx)}}{\sqrt{b}} \right) + \frac{1}{2} \sqrt{\pi} \sqrt{b} e^{-\frac{a}{b}} \operatorname{erfi} \left( \frac{\sqrt{a + b \operatorname{arccosh}(cx)}}{\sqrt{b}} \right)}{b^2 c} \right) - \frac{2\sqrt{cx-1}\sqrt{cx+1}}{bc\sqrt{a + b \operatorname{arccosh}(cx)}}}{3bc} \right) - \frac{2x}{3bc(a + b \operatorname{arccosh}(cx))} \\
 & \frac{2\sqrt{cx-1}\sqrt{cx+1}}{5bc(a + b \operatorname{arccosh}(cx))^{5/2}} \quad 5b
 \end{aligned}$$

input `Int[(a + b*ArcCosh[c*x])^(-7/2), x]`

output `(-2*Sqrt[-1 + c*x]*Sqrt[1 + c*x])/(5*b*c*(a + b*ArcCosh[c*x])^(5/2)) + (2*c*((-2*x)/(3*b*c*(a + b*ArcCosh[c*x])^(3/2)) + (2*((-2*Sqrt[-1 + c*x]*Sqrt[1 + c*x])/(b*c*Sqrt[a + b*ArcCosh[c*x]]) + (2*((Sqrt[b]*E^(a/b)*Sqrt[Pi]*Erf[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]])/2 + (Sqrt[b]*Sqrt[Pi]*Erfi[Sqrt[a + b*ArcCosh[c*x]]/Sqrt[b]])/(2*E^(a/b))))/(b^2*c)))/(3*b*c))/(5*b)`

**Defintions of rubi rules used**

rule 26 `Int[(Complex[0, a_])*(Fx_), x_Symbol] := Simp[(Complex[Identity[0], a]) Int[Fx, x], x] /; FreeQ[a, x] && EqQ[a^2, 1]`

rule 2611  $\text{Int}[(F_)^((g_.) * ((e_.) + (f_.) * (x_))) / \text{Sqrt}[(c_.) + (d_.) * (x_)], x\_Symbol] :> \text{Simp}[2/d \text{ Subst}[\text{Int}[F^{(g*(e - c*(f/d)) + f*g*(x^2/d))}, x], x, \text{Sqrt}[c + d*x]], x] /; \text{FreeQ}[\{F, c, d, e, f, g\}, x] \&\& !\text{TrueQ}[\$UseGamma]$

rule 2633  $\text{Int}[(F_)^((a_.) + (b_.) * ((c_.) + (d_.) * (x_))^{2}), x\_Symbol] :> \text{Simp}[F^a * \text{Sqrt}[\text{Pi}] * (\text{Erfi}[(c + d*x) * \text{Rt}[b * \text{Log}[F], 2]] / (2 * d * \text{Rt}[b * \text{Log}[F], 2])), x] /; \text{FreeQ}[\{F, a, b, c, d\}, x] \&\& \text{PosQ}[b]$

rule 2634  $\text{Int}[(F_)^((a_.) + (b_.) * ((c_.) + (d_.) * (x_))^{2}), x\_Symbol] :> \text{Simp}[F^a * \text{Sqrt}[\text{Pi}] * (\text{Erf}[(c + d*x) * \text{Rt}[(-b) * \text{Log}[F], 2]] / (2 * d * \text{Rt}[(-b) * \text{Log}[F], 2])), x] /; \text{FreeQ}[\{F, a, b, c, d\}, x] \&\& \text{NegQ}[b]$

rule 3042  $\text{Int}[u_, x\_Symbol] :> \text{Int}[\text{DeactivateTrig}[u, x], x] /; \text{FunctionOfTrigOfLinearQ}[u, x]$

rule 3788  $\text{Int}[((c_.) + (d_.) * (x_))^{(m_.)} * \sin[(e_.) + \text{Pi} * (k_.) + (f_.) * (x_)], x\_Symbol] :> \text{Simp}[I/2 \text{ Int}[(c + d*x)^m / (E^{(I*k*Pi)} * E^{(I*(e + f*x))}), x], x] - \text{Simp}[I/2 \text{ Int}[(c + d*x)^m * E^{(I*k*Pi)} * E^{(I*(e + f*x))}, x], x] /; \text{FreeQ}[\{c, d, e, f, m\}, x] \&\& \text{IntegerQ}[2*k]$

rule 6295  $\text{Int}(((a_.) + \text{ArcCosh}[(c_.) * (x_)] * (b_.) )^{(n_.)}, x\_Symbol] :> \text{Simp}[\text{Sqrt}[1 + c*x] * \text{Sqrt}[-1 + c*x] * ((a + b * \text{ArcCosh}[c*x])^{(n + 1)} / (b * c * (n + 1))), x] - \text{Simp}[c / (b * (n + 1)) \text{ Int}[x * ((a + b * \text{ArcCosh}[c*x])^{(n + 1)} / (\text{Sqrt}[1 + c*x] * \text{Sqrt}[-1 + c*x])), x], x] /; \text{FreeQ}[\{a, b, c\}, x] \&\& \text{LtQ}[n, -1]$

rule 6366  $\text{Int}(((a_.) + \text{ArcCosh}[(c_.) * (x_)] * (b_.) )^{(n_.)} * ((f_.) * (x_))^{(m_.)} / (\text{Sqrt}[(d1_.) + (e1_.) * (x_)] * \text{Sqrt}[(d2_.) + (e2_.) * (x_)]), x\_Symbol] :> \text{Simp}[(f*x)^m * ((a + b * \text{ArcCosh}[c*x])^{(n + 1)} / (b * c * (n + 1))) * \text{Simp}[\text{Sqrt}[1 + c*x] / \text{Sqrt}[d1 + e1*x]], x] * \text{Simp}[\text{Sqrt}[-1 + c*x] / \text{Sqrt}[d2 + e2*x]], x] - \text{Simp}[f * (m / (b * c * (n + 1))) * \text{Simp}[\text{Sqrt}[1 + c*x] / \text{Sqrt}[d1 + e1*x]] * \text{Simp}[\text{Sqrt}[-1 + c*x] / \text{Sqrt}[d2 + e2*x]] \text{ Int}[(f*x)^{(m - 1)} * (a + b * \text{ArcCosh}[c*x])^{(n + 1)}, x], x] /; \text{FreeQ}[\{a, b, c, d1, e1, d2, e2, f, m\}, x] \&\& \text{EqQ}[e1, c*d1] \&\& \text{EqQ}[e2, (-c)*d2] \&\& \text{LtQ}[n, -1]$

rule 6368

```
Int[((a_.) + ArcCosh[(c_.)*(x_.)]*(b_.))^(n_.)*(x_)^(m_.)*((d1_) + (e1_.)*(x_))^(p_.)*((d2_) + (e2_.)*(x_.))^(p_.), x_Symbol] := Simp[(1/(b*c^(m + 1)))*Simp[(d1 + e1*x)^p/(1 + c*x)^p]*Simp[(d2 + e2*x)^p/(-1 + c*x)^p] Subst[Int[x^n*Cosh[-a/b + x/b]^m*Sinh[-a/b + x/b]^(2*p + 1), x], x, a + b*ArcCosh[c*x]], x] /; FreeQ[{a, b, c, d1, e1, d2, e2, n}, x] && EqQ[e1, c*d1] && EqQ[e2, (-c)*d2] && IGtQ[p + 3/2, 0] && IGtQ[m, 0]
```

**Maple [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{\frac{7}{2}}} dx$$

input `int(1/(a+b*arccosh(c*x))^(7/2),x)`

output `int(1/(a+b*arccosh(c*x))^(7/2),x)`

**Fricas [F(-2)]**

Exception generated.

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{\frac{7}{2}}} dx = \text{Exception raised: TypeError}$$

input `integrate(1/(a+b*arccosh(c*x))^(7/2),x, algorithm="fricas")`

output `Exception raised: TypeError >> Error detected within library code: integrate: implementation incomplete (constant residues)`

**Sympy [F(-1)]**

Timed out.

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{7/2}} dx = \text{Timed out}$$

input `integrate(1/(a+b*acosh(c*x))**(7/2),x)`output `Timed out`**Maxima [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{7/2}} dx = \int \frac{1}{(b \operatorname{arcosh}(cx) + a)^{7/2}} dx$$

input `integrate(1/(a+b*arccosh(c*x))^(7/2),x, algorithm="maxima")`output `integrate((b*arccosh(c*x) + a)^(-7/2), x)`**Giac [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{7/2}} dx = \int \frac{1}{(b \operatorname{arcosh}(cx) + a)^{7/2}} dx$$

input `integrate(1/(a+b*arccosh(c*x))^(7/2),x, algorithm="giac")`output `integrate((b*arccosh(c*x) + a)^(-7/2), x)`



**Mupad [F(-1)]**

Timed out.

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{7/2}} dx = \int \frac{1}{(a + b \operatorname{acosh}(cx))^{7/2}} dx$$

input `int(1/(a + b*acosh(c*x))^(7/2), x)`output `int(1/(a + b*acosh(c*x))^(7/2), x)`**Reduce [F]**

$$\int \frac{1}{(a + b \operatorname{arccosh}(cx))^{7/2}} dx = \text{too large to display}$$

input `int(1/(a+b*acosh(c*x))^(7/2), x)`

output

```

(3*acosh(c*x)**3*int((sqrt(acosh(c*x)*b + a)*acosh(c*x)*x**2)/(acosh(c*x)*
**4*b**4*c**2*x**2 - acosh(c*x)**4*b**4 + 4*acosh(c*x)**3*a*b**3*c**2*x**2
- 4*acosh(c*x)**3*a*b**3 + 6*acosh(c*x)**2*a**2*b**2*c**2*x**2 - 6*acosh(c
*x)**2*a**2*b**2 + 4*acosh(c*x)*a**3*b*c**2*x**2 - 4*acosh(c*x)*a**3*b + a
**4*c**2*x**2 - a**4),x)*b**4*c**3 - 3*acosh(c*x)**3*int((sqrt(acosh(c*x)*
b + a)*acosh(c*x))/(acosh(c*x)**4*b**4*c**2*x**2 - acosh(c*x)**4*b**4 + 4*
acosh(c*x)**3*a*b**3*c**2*x**2 - 4*acosh(c*x)**3*a*b**3 + 6*acosh(c*x)**2*
a**2*b**2*c**2*x**2 - 6*acosh(c*x)**2*a**2*b**2 + 4*acosh(c*x)*a**3*b*c**2
*x**2 - 4*acosh(c*x)*a**3*b + a**4*c**2*x**2 - a**4),x)*b**4*c - 2*acosh(c
*x)**3*int((sqrt(c*x + 1)*sqrt(c*x - 1)*sqrt(acosh(c*x)*b + a)*acosh(c*x)*
x)/(acosh(c*x)**4*b**4*c**2*x**2 - acosh(c*x)**4*b**4 + 4*acosh(c*x)**3*a*
b**3*c**2*x**2 - 4*acosh(c*x)**3*a*b**3 + 6*acosh(c*x)**2*a**2*b**2*c**2*x
**2 - 6*acosh(c*x)**2*a**2*b**2 + 4*acosh(c*x)*a**3*b*c**2*x**2 - 4*acosh(
c*x)*a**3*b + a**4*c**2*x**2 - a**4),x)*a*b**3*c**2 - 2*acosh(c*x)**3*int(
(sqrt(c*x + 1)*sqrt(c*x - 1)*sqrt(acosh(c*x)*b + a)*acosh(c*x)**2*x)/(acos
h(c*x)**4*b**4*c**2*x**2 - acosh(c*x)**4*b**4 + 4*acosh(c*x)**3*a*b**3*c**
2*x**2 - 4*acosh(c*x)**3*a*b**3 + 6*acosh(c*x)**2*a**2*b**2*c**2*x**2 - 6*
acosh(c*x)**2*a**2*b**2 + 4*acosh(c*x)*a**3*b*c**2*x**2 - 4*acosh(c*x)*a**
3*b + a**4*c**2*x**2 - a**4),x)*b**4*c**2 + 9*acosh(c*x)**2*int((sqrt(acos
h(c*x)*b + a)*acosh(c*x)*x**2)/(acosh(c*x)**4*b**4*c**2*x**2 - acosh(c*...

```

### 3.15 $\int (a + b \operatorname{arccosh}(cx))^n dx$

Optimal result	138
Mathematica [A] (verified)	138
Rubi [C] (verified)	139
Maple [F]	141
Fricas [F]	141
Sympy [F]	142
Maxima [F]	142
Giac [F]	142
Mupad [F(-1)]	143
Reduce [F]	143

#### Optimal result

Integrand size = 10, antiderivative size = 116

$$\int (a + b \operatorname{arccosh}(cx))^n dx$$

$$= \frac{e^{-\frac{a}{b}} (a + b \operatorname{arccosh}(cx))^n \left( -\frac{a + b \operatorname{arccosh}(cx)}{b} \right)^{-n} \Gamma\left(1 + n, -\frac{a + b \operatorname{arccosh}(cx)}{b}\right)}{2c} + \frac{e^{a/b} (a + b \operatorname{arccosh}(cx))^n \left( \frac{a + b \operatorname{arccosh}(cx)}{b} \right)^{-n} \Gamma\left(1 + n, \frac{a + b \operatorname{arccosh}(cx)}{b}\right)}{2c}$$

```
output 1/2*(a+b*arccosh(c*x))^n*GAMMA(1+n,-(a+b*arccosh(c*x))/b)/c/exp(a/b)/((-a+b*arccosh(c*x))/b)^n+1/2*exp(a/b)*(a+b*arccosh(c*x))^n*GAMMA(1+n,(a+b*arccosh(c*x))/b)/c/(((a+b*arccosh(c*x))/b)^n)
```

#### Mathematica [A] (verified)

Time = 0.09 (sec) , antiderivative size = 98, normalized size of antiderivative = 0.84

$$\int (a + b \operatorname{arccosh}(cx))^n dx$$

$$= \frac{e^{-\frac{a}{b}} (a + b \operatorname{arccosh}(cx))^n \left( e^{\frac{2a}{b}} \left( \frac{a}{b} + \operatorname{arccosh}(cx) \right)^{-n} \Gamma\left(1 + n, \frac{a}{b} + \operatorname{arccosh}(cx)\right) + \left( -\frac{a + b \operatorname{arccosh}(cx)}{b} \right)^{-n} \Gamma\left(1 + n, -\frac{a + b \operatorname{arccosh}(cx)}{b}\right) \right)}{2c}$$

input `Integrate[(a + b*ArcCosh[c*x])^n,x]`

output  $((a + b*\text{ArcCosh}[c*x])^n*((E^{((2*a)/b)}*\text{Gamma}[1 + n, a/b + \text{ArcCosh}[c*x]])/(a/b + \text{ArcCosh}[c*x])^n + \text{Gamma}[1 + n, -((a + b*\text{ArcCosh}[c*x])/b)]/(-((a + b*\text{ArcCosh}[c*x])/b))^n))/(2*c*E^{(a/b)})$

## Rubi [C] (verified)

Result contains complex when optimal does not.

Time = 0.38 (sec) , antiderivative size = 126, normalized size of antiderivative = 1.09, number of steps used = 7, number of rules used = 6,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.600$ , Rules used = {6296, 25, 3042, 26, 3789, 2612}

Below are the steps used by Rubi to obtain the solution. The rule number used for the transformation is given above next to the arrow. The rules definitions used are listed below.

$$\begin{aligned}
 & \int (a + \text{barccosh}(cx))^n dx \\
 & \quad \downarrow \text{6296} \\
 & \frac{\int -(a + \text{barccosh}(cx))^n \sinh\left(\frac{a}{b} - \frac{a + \text{barccosh}(cx)}{b}\right) d(a + \text{barccosh}(cx))}{bc} \\
 & \quad \downarrow \text{25} \\
 & - \frac{\int (a + \text{barccosh}(cx))^n \sinh\left(\frac{a}{b} - \frac{a + \text{barccosh}(cx)}{b}\right) d(a + \text{barccosh}(cx))}{bc} \\
 & \quad \downarrow \text{3042} \\
 & \frac{\int -i(a + \text{barccosh}(cx))^n \sin\left(\frac{ia}{b} - \frac{i(a + \text{barccosh}(cx))}{b}\right) d(a + \text{barccosh}(cx))}{bc} \\
 & \quad \downarrow \text{26} \\
 & \frac{i \int (a + \text{barccosh}(cx))^n \sin\left(\frac{ia}{b} - \frac{i(a + \text{barccosh}(cx))}{b}\right) d(a + \text{barccosh}(cx))}{bc} \\
 & \quad \downarrow \text{3789}
 \end{aligned}$$

$$\frac{i\left(\frac{1}{2}i \int e^{-\operatorname{arccosh}(cx)}(a + \operatorname{barccosh}(cx))^n d(a + \operatorname{barccosh}(cx)) - \frac{1}{2}i \int e^{\operatorname{arccosh}(cx)}(a + \operatorname{barccosh}(cx))^n d(a + \operatorname{barccosh}(cx))\right)}{bc}$$

↓ 2612

$$\frac{i\left(-\frac{1}{2}ibe^{-\frac{a}{b}}(a + \operatorname{barccosh}(cx))^n \left(-\frac{a + \operatorname{barccosh}(cx)}{b}\right)^{-n} \Gamma\left(n + 1, -\frac{a + \operatorname{barccosh}(cx)}{b}\right) - \frac{1}{2}ibe^{a/b}(a + \operatorname{barccosh}(cx))^n\right)}{bc}$$

input

```
Int[(a + b*ArcCosh[c*x])^n, x]
```

output

```
(I*((( -1/2*I)*b*(a + b*ArcCosh[c*x])^n*Gamma[1 + n, -(a + b*ArcCosh[c*x])/b]))/(E^(a/b)*(-(a + b*ArcCosh[c*x])/b)^n) - ((I/2)*b*E^(a/b)*(a + b*ArcCosh[c*x])^n*Gamma[1 + n, (a + b*ArcCosh[c*x])/b])/((a + b*ArcCosh[c*x])/b)^n)/(b*c)
```

### Defintions of rubi rules used

rule 25

```
Int[-(Fx_), x_Symbol] :> Simp[Identity[-1] Int[Fx, x], x]
```

rule 26

```
Int[(Complex[0, a_]*(Fx_), x_Symbol] :> Simp[(Complex[Identity[0], a]) Int[Fx, x], x] /; FreeQ[a, x] && EqQ[a^2, 1]
```

rule 2612

```
Int[(F_)^((g_.)*((e_.) + (f_.)*(x_)))*((c_.) + (d_.)*(x_))^(m_), x_Symbol] :> Simp[(-F^(g*(e - c*(f/d))))*((c + d*x)^FracPart[m]/(d*(-f)*g*(Log[F]/d))^(IntPart[m] + 1)*((-f)*g*Log[F]*((c + d*x)/d)^FracPart[m]))*Gamma[m + 1, ((-f)*g*(Log[F]/d)*(c + d*x)], x] /; FreeQ[{F, c, d, e, f, g, m}, x] && !IntegerQ[m]
```

rule 3042

```
Int[u_, x_Symbol] :> Int[DeactivateTrig[u, x], x] /; FunctionOfTrigOfLinearQ[u, x]
```

rule 3789

```
Int[((c_.) + (d_.)*(x_)^(m_.)*sin[(e_.) + (f_.)*(x_)], x_Symbol] := Simp[I/2 Int[(c + d*x)^m/E^(I*(e + f*x)), x], x] - Simp[I/2 Int[(c + d*x)^m*E^(I*(e + f*x)), x], x] /; FreeQ[{c, d, e, f, m}, x]
```

rule 6296

```
Int[((a_.) + ArcCosh[(c_.)*(x_)]*(b_.))^(n_), x_Symbol] := Simp[1/(b*c) Subst[Int[x^n*Sinh[-a/b + x/b], x], x, a + b*ArcCosh[c*x]], x] /; FreeQ[{a, b, c, n}, x]
```

**Maple [F]**

$$\int (a + b \operatorname{arccosh}(cx))^n dx$$

input

```
int((a+b*arccosh(c*x))^n,x)
```

output

```
int((a+b*arccosh(c*x))^n,x)
```

**Fricas [F]**

$$\int (a + b \operatorname{arccosh}(cx))^n dx = \int (b \operatorname{arccosh}(cx) + a)^n dx$$

input

```
integrate((a+b*arccosh(c*x))^n,x, algorithm="fricas")
```

output

```
integral((b*arccosh(c*x) + a)^n, x)
```

**Sympy [F]**

$$\int (a + b \operatorname{arccosh}(cx))^n dx = \int (a + b \operatorname{arcosh}(cx))^n dx$$

input `integrate((a+b*acosh(c*x))**n,x)`

output `Integral((a + b*acosh(c*x))**n, x)`

**Maxima [F]**

$$\int (a + b \operatorname{arccosh}(cx))^n dx = \int (b \operatorname{arcosh}(cx) + a)^n dx$$

input `integrate((a+b*arccosh(c*x))^n,x, algorithm="maxima")`

output `integrate((b*arccosh(c*x) + a)^n, x)`

**Giac [F]**

$$\int (a + b \operatorname{arccosh}(cx))^n dx = \int (b \operatorname{arcosh}(cx) + a)^n dx$$

input `integrate((a+b*arccosh(c*x))^n,x, algorithm="giac")`

output `integrate((b*arccosh(c*x) + a)^n, x)`

**Mupad [F(-1)]**

Timed out.

$$\int (a + b \operatorname{arccosh}(cx))^n dx = \int (a + b \operatorname{acosh}(cx))^n dx$$

input `int((a + b*acosh(c*x))^n,x)`output `int((a + b*acosh(c*x))^n, x)`**Reduce [F]**

$$\int (a + b \operatorname{arccosh}(cx))^n dx = \int (\operatorname{acosh}(cx) b + a)^n dx$$

input `int((a+b*acosh(c*x))^n,x)`output `int((acosh(c*x)*b + a)**n,x)`



# CHAPTER 4

## APPENDIX

4.1	Listing of Grading functions . . . . .	144
4.2	Links to plain text integration problems used in this report for each CAS .	162

### 4.1 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 Maple and for Mathematica/Rubi. There is a version for grading Sympy and version for use with Sagemath.

The following are links to the current source code.

The following are the listings of source code of the grading functions.

#### Mathematica and Rubi grading function

```
(* Original version thanks to Albert Rich emailed on 03/21/2017 *)
(* ::Package:: *)

(* Nasser: April 7, 2022. add second output which gives reason for the grade *)
(*                               Small rewrite of logic in main function to make it*)
(*                               match Maple's logic. No change in functionality otherwise*)

(* ::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_] := Module[{expnResult,expnOptimal,leafCountResult,leafCountOptimal},
  expnResult = ExpnType[result];
  expnOptimal = ExpnType[optimal];
  leafCountResult = LeafCount[result];
  leafCountOptimal = LeafCount[optimal];

  (*Print["expnResult=",expnResult," expnOptimal=",expnOptimal];*)
  If[expnResult<=expnOptimal,
    If[Not[FreeQ[result,Complex]], (*result contains complex*)
      If[Not[FreeQ[optimal,Complex]], (*optimal contains complex*)
        If[leafCountResult<=2*leafCountOptimal,
          finalresult={"A",""}
          ,(*ELSE*)
          finalresult={"B","Both result and optimal contain complex but leaf count
          ]
        ,(*ELSE*)
          finalresult={"C","Result contains complex when optimal does not."}
        ]
      ,(*ELSE*)(*result does not contains complex*)
        If[leafCountResult<=2*leafCountOptimal,
          finalresult={"A",""}
          ,(*ELSE*)
          finalresult={"B","Leaf count is larger than twice the leaf count of optimal.
          ]
        ]
      ,(*ELSE*)(*expnResult>expnOptimal*)
        If[FreeQ[result,Integrate] && FreeQ[result,Int],
          finalresult={"C","Result contains higher order function than in optimal. Order "
          ,

```

```

        finalresult={"F","Contains unresolved integral."}
    ]
];

finalresult
]

(* ::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]

```

## Maple grading function

```

# 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
#Nasser 12/22/2019 Added debug flag, added 'dilog' to special functions
#
#                   see problem 156, file Apostol_Problems
#Nasser 4/07/2022 add second output which gives reason for the grade

GradeAntiderivative := proc(result,optimal)
local leaf_count_result,
      leaf_count_optimal,
      ExpnType_result,
      ExpnType_optimal,
      debug:=false;

      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","result has leaf size over 500,000. Avoiding possible recursion issue";
      fi;

      leaf_count_optimal := leafcount(optimal);
      ExpnType_result := ExpnType(result);
      ExpnType_optimal := ExpnType(optimal);

      if debug then
          print("ExpnType_result",ExpnType_result," ExpnType_optimal=",ExpnType_optimal);
      fi;

# 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

#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. But no harm of keeping it here.
#just in case.

if not type(result,freeof('int')) then
    return "F","Result contains unresolved integral";
fi;

if ExpnType_result<=ExpnType_optimal then
    if debug then
        print("ExpnType_result<=ExpnType_optimal");
    fi;
    if is_contains_complex(result) then
        if is_contains_complex(optimal) then
            if debug then
                print("both result and optimal complex");
            fi;
            if leaf_count_result<=2*leaf_count_optimal then
                return "A"," ";
            else
                return "B",cat("Both result and optimal contain complex but leaf count of
                                convert(leaf_count_result,string)," vs. $2 (" ,
                                convert(leaf_count_optimal,string)," ) = ",convert(2*leaf

            end if
        else #result contains complex but optimal is not
            if debug then
                print("result contains complex but optimal is not");
            fi;
            return "C","Result contains complex when optimal does not.";
        fi;
    else # result do not contain complex
        # this assumes optimal do not as well. No check is needed here.
        if debug then
            print("result do not contain complex, this assumes optimal do not as well
        fi;

```

```

        if leaf_count_result<=2*leaf_count_optimal then
            if debug then
                print("leaf_count_result<=2*leaf_count_optimal");
            fi;
            return "A"," ";
        else
            if debug then
                print("leaf_count_result>2*leaf_count_optimal");
            fi;
            return "B",cat("Leaf count of result is larger than twice the leaf count of
                convert(leaf_count_result,string)," $ vs. $2(",
                convert(leaf_count_optimal,string),")=",convert(2*leaf_co
            fi;
        fi;
    else #ExpnType(result) > ExpnType(optimal)
        if debug then
            print("ExpnType(result) > ExpnType(optimal)");
        fi;
        return "C",cat("Result contains higher order function than in optimal. Order ",
            convert(ExpnType_result,string)," vs. order ",
            convert(ExpnType_optimal,string),".");
    fi;
end proc:

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

# 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 := 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' 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, dilog, 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;
```

## Sympy grading function

```
#Dec 24, 2019. Nasser M. Abbasi:
#      Port of original Maple grading function by
#      Albert Rich to use with Sympy/Python
#Dec 27, 2019 Nasser. Added `RootSum`. See problem 177, Timofeev file
#      added 'exp_polar'
from sympy import *

def leaf_count(expr):
  #sympy do not have leaf count function. This is approximation
  return round(1.7*count_ops(expr))

def is_sqrt(expr):
  if isinstance(expr,Pow):
    if expr.args[1] == Rational(1,2):
      return True
    else:
      return False
  else:
    return False

def is_elementary_function(func):
  return func in [exp,log,ln,sin,cos,tan,cot,sec,csc,
    asin,acos,atan,acot,asec,acsc,sinh,cosh,tanh,coth,sech,csch,
    asinh,acosh,atanh,acoth,asech,acsch
  ]

def is_special_function(func):
  return func in [ erf,erfc,erfi,
    fresnels,fresnelc,Ei,Ei,Li,Si,Ci,Shi,Chi,
    gamma,loggamma,digamma,zeta,polylog,LambertW,
    elliptic_f,elliptic_e,elliptic_pi,exp_polar
  ]
```

```

def is_hypergeometric_function(func):
    return func in [hyper]

def is_appell_function(func):
    return func in [appellf1]

def is_atom(expn):
    try:
        if expn.isAtom or isinstance(expn,int) or isinstance(expn,float):
            return True
        else:
            return False

    except AttributeError as error:
        return False

def expnType(expn):
    debug=False
    if debug:
        print("expn=",expn,"type(expn)=",type(expn))

    if is_atom(expn):
        return 1
    elif isinstance(expn,list):
        return max(map(expnType, expn)) #apply(max,map(ExpnType,expn))
    elif is_sqrt(expn):
        if isinstance(expn.args[0],Rational): #type(op(1,expn),'rational')
            return 1
        else:
            return max(2,expnType(expn.args[0])) #max(2,ExpnType(op(1,expn)))
    elif isinstance(expn,Pow): #type(expn,'^')
        if isinstance(expn.args[1],Integer): #type(op(2,expn),'integer')
            return expnType(expn.args[0]) #ExpnType(op(1,expn))
        elif isinstance(expn.args[1],Rational): #type(op(2,expn),'rational')
            if isinstance(expn.args[0],Rational): #type(op(1,expn),'rational')
                return 1
            else:
                return max(2,expnType(expn.args[0])) #max(2,ExpnType(op(1,expn)))
        else:
            return max(3,expnType(expn.args[0]),expnType(expn.args[1])) #max(3,ExpnType(op(1,expn)),ExpnType(op(2,expn)))
    elif isinstance(expn,Add) or isinstance(expn,Mul): #type(expn,'+') or type(expn,'*')

```

```

    m1 = expnType(expn.args[0])
    m2 = expnType(list(expn.args[1:]))
    return max(m1,m2) #max(ExpnType(op(1,expn)),max(ExpnType(rest(expn))))
elif is_elementary_function(expn.func): #ElementaryFunctionQ(op(0,expn))
    return max(3,expnType(expn.args[0])) #max(3,ExpnType(op(1,expn)))
elif is_special_function(expn.func): #SpecialFunctionQ(op(0,expn))
    m1 = max(map(expnType, list(expn.args)))
    return max(4,m1) #max(4,apply(max,map(ExpnType,[op(expn)])))
elif is_hypergeometric_function(expn.func): #HypergeometricFunctionQ(op(0,expn))
    m1 = max(map(expnType, list(expn.args)))
    return max(5,m1) #max(5,apply(max,map(ExpnType,[op(expn)])))
elif is_appell_function(expn.func):
    m1 = max(map(expnType, list(expn.args)))
    return max(6,m1) #max(5,apply(max,map(ExpnType,[op(expn)])))
elif isinstance(expn,RootSum):
    m1 = max(map(expnType, list(expn.args))) #Apply[Max,Append[Map[ExpnType,Apply[List,expn]],7]],
    return max(7,m1)
elif str(expn).find("Integral") != -1:
    m1 = max(map(expnType, list(expn.args)))
    return max(8,m1) #max(5,apply(max,map(ExpnType,[op(expn)])))
else:
    return 9

#main function
def grade_antiderivative(result,optimal):

    #print("Enter grade_antiderivative for sagemath")
    #print("Enter grade_antiderivative, result=",result," optimal=",optimal)

    leaf_count_result = leaf_count(result)
    leaf_count_optimal = leaf_count(optimal)

    #print("leaf_count_result=",leaf_count_result)
    #print("leaf_count_optimal=",leaf_count_optimal)

    expnType_result = expnType(result)
    expnType_optimal = expnType(optimal)

    if str(result).find("Integral") != -1:
        grade = "F"
        grade_annotation = ""

```

```

else:
    if expnType_result <= expnType_optimal:
        if result.has(I):
            if optimal.has(I): #both result and optimal complex
                if leaf_count_result <= 2*leaf_count_optimal:
                    grade = "A"
                    grade_annotation = ""
                else:
                    grade = "B"
                    grade_annotation = "Both result and optimal contain complex but leaf count of result is lar
            else: #result contains complex but optimal is not
                grade = "C"
                grade_annotation = "Result contains complex when optimal does not."
        else: # result do not contain complex, this assumes optimal do not as well
            if leaf_count_result <= 2*leaf_count_optimal:
                grade = "A"
                grade_annotation = ""
            else:
                grade = "B"
                grade_annotation = "Leaf count of result is larger than twice the leaf count of optimal. "+str(
        else:
            grade = "C"
            grade_annotation = "Result contains higher order function than in optimal. Order "+str(ExpnType

#print("Before returning. grade=",grade, " grade_annotation=",grade_annotation)

return grade, grade_annotation

```

## SageMath grading function

```

#Dec 24, 2019. Nasser: Ported original Maple grading function by
#       Albert Rich to use with Sagemath. This is used to
#       grade Fracas, Giac and Maxima results.
#Dec 24, 2019. Nasser: Added 'exp_integral_e' and 'sng', 'sin_integral'
#       'arctan2', 'floor', 'abs', 'log_integral'
#June 4, 2022 Made default grade_annotation "none" instead of "" due
#       issue later when reading the file.
#July 14, 2022. Added ellipticF. This is until they fix sagemath, then remove it.

```

```

from sage.all import *
from sage.symbolic.operators import add_vararg, mul_vararg

debug=False;

def tree_size(expr):
    r"""
    Return the tree size of this expression.
    """
    #print("Enter tree_size, expr is ",expr)

    if expr not in SR:
        # deal with lists, tuples, vectors
        return 1 + sum(tree_size(a) for a in expr)
    expr = SR(expr)
    x, aa = expr.operator(), expr.operands()
    if x is None:
        return 1
    else:
        return 1 + sum(tree_size(a) for a in aa)

def is_sqrt(expr):
    if expr.operator() == operator.pow: #isinstance(expr,Pow):
        if expr.operands()[1]==1/2: #expr.args[1] == Rational(1,2):
            if debug: print ("expr is sqrt")
            return True
        else:
            return False
    else:
        return False

def is_elementary_function(func):
    #debug=False
    m = func.name() in ['exp','log','ln',
        'sin','cos','tan','cot','sec','csc',
        'arcsin','arccos','arctan','arccot','arcsec','arccsc',
        'sinh','cosh','tanh','coth','sech','csch',
        'arcsinh','arccosh','arctanh','arcoth','arcsech','arcsch','sgn',
        'arctan2','floor','abs'
    ]
    if debug:

```

```

    if m:
        print ("func ", func , " is elementary_function")
    else:
        print ("func ", func , " is NOT elementary_function")

    return m

def is_special_function(func):
    #debug=False
    if debug:
        print ("type(func)=", type(func))

    m= func.name() in ['erf','erfc','erfi','fresnel_sin','fresnel_cos','Ei',
        'Ei','Li','Si','sin_integral','Ci','cos_integral','Shi','sinh_integral',
        'Chi','cosh_integral','gamma','log_gamma','psi','zeta',
        'polylog','lambert_w','elliptic_f','elliptic_e','ellipticF',
        'elliptic_pi','exp_integral_e','log_integral',
        'weierstrassPInverse','weierstrass','weierstrassP','weierstrassZeta',
        'weierstrassPPrime','weierstrassSigma']

    if debug:
        print ("m=",m)
    if m:
        print ("func ", func , " is special_function")
    else:
        print ("func ", func , " is NOT special_function")

    return m

def is_hypergeometric_function(func):
    return func.name() in ['hypergeometric','hypergeometric_M','hypergeometric_U']

def is_appell_function(func):
    return func.name() in ['hypergeometric'] #[appellf1] can't find this in sagemath

def is_atom(expn):

    #debug=False

```

```

if debug:
    print ("Enter is_atom, expn=",expn)

if not hasattr(expn, 'parent'):
    return False

#thanks to answer at https://ask.sagemath.org/question/49179/what-is-sagemath-equivalent-to-atomic
try:
    if expn.parent() is SR:
        return expn.operator() is None
    if expn.parent() in (ZZ, QQ, AA, QQbar):
        return expn in expn.parent() # Should always return True
    if hasattr(expn.parent(),"base_ring") and hasattr(expn.parent(),"gens"):
        return expn in expn.parent().base_ring() or expn in expn.parent().gens()

    return False

except AttributeError as error:
    print ("Exception,AttributeError in is_atom")
    print ("caught exception" , type(error).__name__ )
    return False

def expnType(expn):

    if debug:
        print (">>>>>Enter expnType, expn=", expn)
        print (">>>>>is_atom(expn)=", is_atom(expn))

    if is_atom(expn):
        return 1
    elif type(expn)==list: #instance(expn,list):
        return max(map(expnType, expn)) #apply(max,map(ExpnType,expn))
    elif is_sqrt(expn):
        if type(expn.operands()[0])==Rational: #type(instance(expn.args[0],Rational):
            return 1
        else:
            return max(2,expnType(expn.operands()[0])) #max(2,expnType(expn.args[0]))
    elif expn.operator() == operator.pow: #instance(expn,Pow)
        if type(expn.operands()[1])==Integer: #instance(expn.args[1],Integer)

```



```

    return expnType(expn.operands()[0]) #expnType(expn.args[0])
elif type(expn.operands()[1])==Rational: #isinstance(expn.args[1],Rational)
    if type(expn.operands()[0])==Rational: #isinstance(expn.args[0],Rational)
        return 1
    else:
        return max(2,expnType(expn.operands()[0])) #max(2,expnType(expn.args[0]))
else:
    return max(3,expnType(expn.operands()[0]),expnType(expn.operands()[1])) #max(3,expnType(expn
elif expn.operator() == add_vararg or expn.operator() == mul_vararg: #isinstance(expn,Add) or isins
    m1 = expnType(expn.operands()[0]) #expnType(expn.args[0])
    m2 = expnType(expn.operands()[1:]) #expnType(list(expn.args[1:]))
    return max(m1,m2) #max(ExpnType(op(1,expn)),max(ExpnType(rest(expn))))
elif is_elementary_function(expn.operator()): #is_elementary_function(expn.func)
    return max(3,expnType(expn.operands()[0]))
elif is_special_function(expn.operator()): #is_special_function(expn.func)
    m1 = max(map(expnType, expn.operands())) #max(map(expnType, list(expn.args)))
    return max(4,m1) #max(4,m1)
elif is_hypergeometric_function(expn.operator()): #is_hypergeometric_function(expn.func)
    m1 = max(map(expnType, expn.operands())) #max(map(expnType, list(expn.args)))
    return max(5,m1) #max(5,m1)
elif is_appell_function(expn.operator()):
    m1 = max(map(expnType, expn.operands())) #max(map(expnType, list(expn.args)))
    return max(6,m1) #max(6,m1)
elif str(expn).find("Integral") != -1: #this will never happen, since it
    #is checked before calling the grading function that is passed.
    #but kept it here.
    m1 = max(map(expnType, expn.operands())) #max(map(expnType, list(expn.args)))
    return max(8,m1) #max(5,apply(max,map(ExpnType,[op(expn)])))
else:
    return 9

#main function
def grade_antiderivative(result,optimal):

if debug:
    print ("Enter grade_antiderivative for sagemath")
    print ("Enter grade_antiderivative, result=",result)
    print ("Enter grade_antiderivative, optimal=",optimal)
    print ("type(anti)=", type(result))
    print ("type(optimal)=", type(optimal))

```

```

leaf_count_result = tree_size(result) #leaf_count(result)
leaf_count_optimal = tree_size(optimal) #leaf_count(optimal)

#if debug: print ("leaf_count_result=", leaf_count_result, "leaf_count_optimal=",leaf_count_optimal)

expnType_result = expnType(result)
expnType_optimal = expnType(optimal)

if debug: print ("expnType_result=", expnType_result, "expnType_optimal=",expnType_optimal)

if expnType_result <= expnType_optimal:
    if result.has(I):
        if optimal.has(I): #both result and optimal complex
            if leaf_count_result <= 2*leaf_count_optimal:
                grade = "A"
                grade_annotation = "none"
            else:
                grade = "B"
                grade_annotation = "Both result and optimal contain complex but leaf count of result is larger"
        else: #result contains complex but optimal is not
            grade = "C"
            grade_annotation = "Result contains complex when optimal does not."
    else: # result do not contain complex, this assumes optimal do not as well
        if leaf_count_result <= 2*leaf_count_optimal:
            grade = "A"
            grade_annotation = "none"
        else:
            grade = "B"
            grade_annotation = "Leaf count of result is larger than twice the leaf count of optimal. "+str(leaf_count_result - 2*leaf_count_optimal)
    else:
        grade = "C"
        grade_annotation = "Result contains higher order function than in optimal. Order "+str(expnType_result - expnType_optimal)

print("Before returning. grade=",grade, " grade_annotation=",grade_annotation)

return grade, grade_annotation

```

## 4.2 Links to plain text integration problems used in this report for each CAS

1. Mathematica integration problems as .m file
2. Maple integration problems as .txt file
3. Sagemath integration problems as .sage file
4. Reduce integration problems as .txt file
5. Mupad integration problems as .txt file
6. Sympy integration problems as .py file