

# Computer algebra independent integration tests

Summer 2022 edition

6-Hyperbolic-functions/6.5-Hyperbolic-secant/177-6.5.1-c+d-x<sup>m</sup>-  
a+b-sech<sup>n</sup>

Nasser M. Abbasi

September 27, 2022

Compiled on September 27, 2022 at 6:39am

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>detailed summary tables of results</b>	<b>19</b>
<b>3</b>	<b>Listing of integrals</b>	<b>29</b>
<b>4</b>	<b>Appendix</b>	<b>91</b>

# Chapter 1

## Introduction

### Local contents

1.1	Listing of CAS systems tested . . . . .	4
1.2	Results . . . . .	5
1.3	Time and leaf size Performance . . . . .	9
1.4	list of integrals that has no closed form antiderivative . . . . .	11
1.5	List of integrals solved by CAS but has no known antiderivative . . . . .	12
1.6	list of integrals solved by CAS but failed verification . . . . .	13
1.7	Timing . . . . .	13
1.8	Verification . . . . .	14
1.9	Important notes about some of the results . . . . .	14
1.10	Design of the test system . . . . .	17

This report gives the result of running the computer algebra independent integration test. The download section in the appendix contains links to download the problems in plain text format used for all CAS systems.

The number of integrals in this report is [ 16 ]. This is test number [ 177 ].

## 1.1 Listing of CAS systems tested

The following are the CAS systems tested:

1. Mathematica 13.1 (June 29, 2022) on windows 10.
2. Rubi 4.16.1 (Dec 19, 2018) on Mathematica 13.0.1 on windows 10.
3. Maple 2022.1 (June 1, 2022) on windows 10.
4. Maxima 5.46 (April 13, 2022) using Lisp SBCL 2.1.11.debian on Linux via sagemath 9.6.
5. Fricas 1.3.8 (June 21, 2022) based on sbcl 2.1.11.debian on Linux via sagemath 9.6.
6. Giac/Xcas 1.9.0-13 (July 3, 2022) on Linux via sagemath 9.6.
7. Sympy 1.10.1 (March 20, 2022) Using Python 3.10.4 on Linux.
8. Mupad using Matlab 2021a with Symbolic Math Toolbox Version 8.7 on windows 10.

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 called directly from Python.

## 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 ( 16 )	0.00 ( 0 )
Mathematica	100.00 ( 16 )	0.00 ( 0 )
Fricas	75.00 ( 12 )	25.00 ( 4 )
Maple	50.00 ( 8 )	50.00 ( 8 )
Maxima	31.25 ( 5 )	68.75 ( 11 )
Mupad	25.00 ( 4 )	75.00 ( 12 )
Giac	25.00 ( 4 )	75.00 ( 12 )
Sympy	18.75 ( 3 )	81.25 ( 13 )

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.

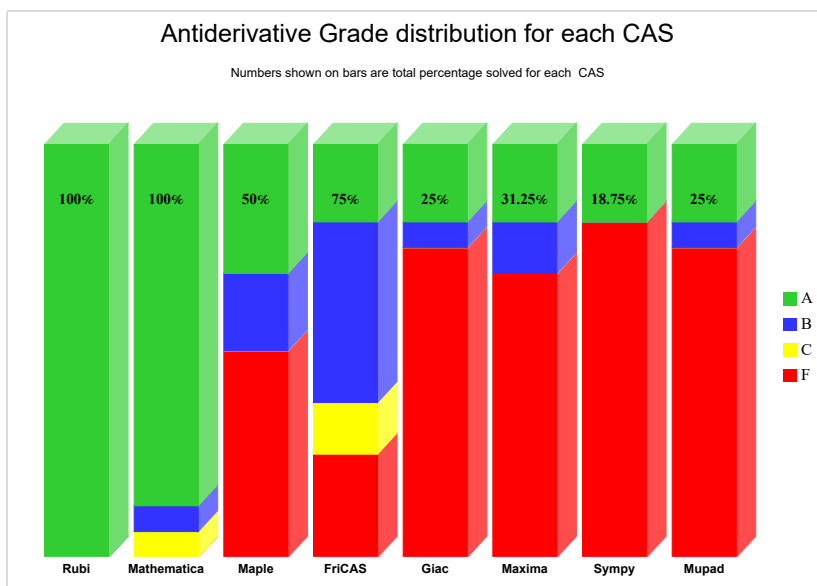
Table 1.2: Description of grading applied to integration result

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

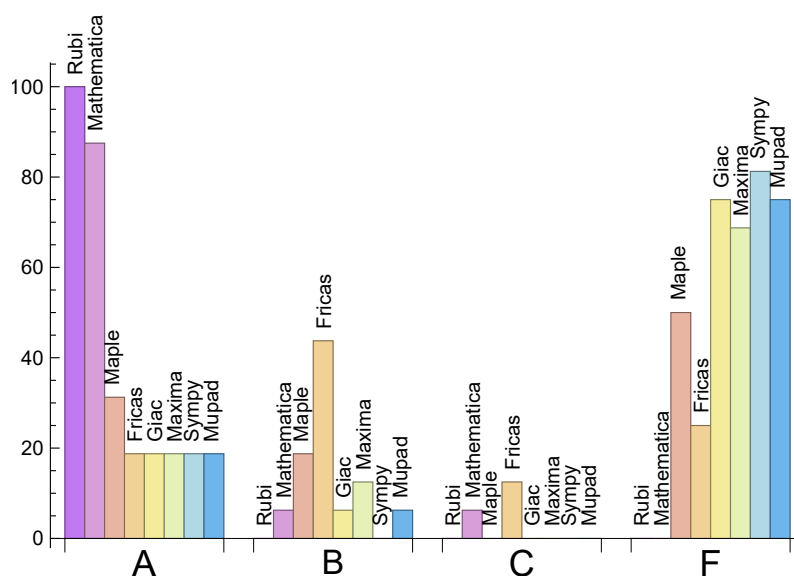
System	% A grade	% B grade	% C grade	% F grade
Rubi	100.00	0.00	0.00	0.00
Mathematica	87.50	6.25	6.25	0.00
Maple	31.25	18.75	0.00	50.00
Fricas	18.75	43.75	12.50	25.00
Mupad	N/A	6.25	0.00	75.00
Giac	18.75	6.25	0.00	75.00
Maxima	18.75	12.50	0.00	68.75
Sympy	18.75	0.00	0.00	81.25

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 CAS systems for each grade level.



The following table shows the distribution of the different types of failure for each CAS. There are 3 types of reasons why it can fail. The first is when CAS returns back the input within the time limit, which means it could not solve it. This is the typical normal failure **F**.

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

The third is due to an exception generated. Assigned **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 CAS. This type of error requires more investigations 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	4	0.00 %	0.00 %	100.00 %
Giac	12	100.00 %	0.00 %	0.00 %
Maxima	11	100.00 %	0.00 %	0.00 %
Sympy	13	100.00 %	0.00 %	0.00 %
Mupad	12	100.00 %	0.00 %	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.

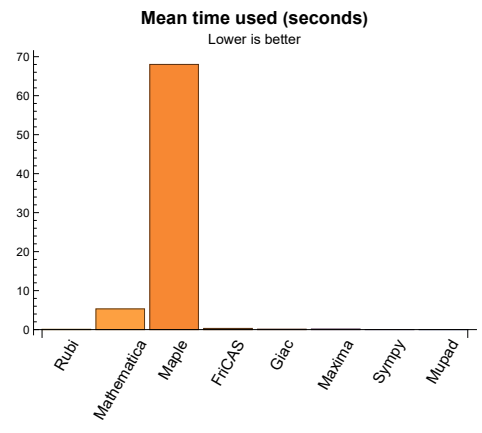
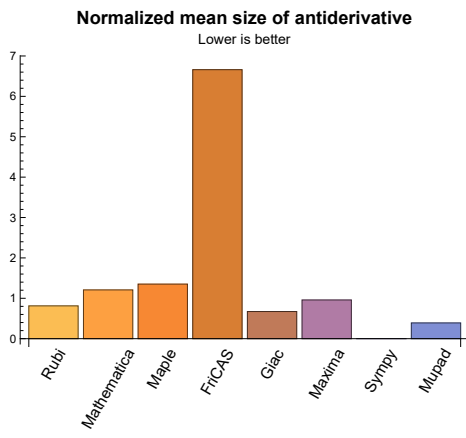
Similarly the **Normalized median** is relative to the median leaf size of the optimal.

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)	Mean size	Normalized mean	Median size	Normalized median
Rubi	0.07	81.13	0.81	63.50	1.00
Mathematica	5.32	129.88	1.21	91.00	1.42
Maple	68.01	103.88	1.35	79.00	1.81
Maxima	0.13	62.00	0.96	0.00	0.00
Fricas	0.30	989.17	6.66	401.00	4.16
Sympy	0.00	0.00	0.00	0.00	0.00
Giac	0.10	19.50	0.67	0.00	0.00
Mupad	0.02	11.75	0.39	-1.00	-0.05

Table 1.5: Time and leaf size performance for each CAS

The following are bar charts for the normalized leafsize and time used from the above table.



## **1.4 list of integrals that has no closed form antiderivative**

{4, 8, 12}

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

Rubi {}

Mathematica {}

Maple {}

Maxima {}

Fricas {}

Sympy {}

Giac {}

Mupad {}

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

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

**Rubi** {}

**Mathematica** {}

**Maple** Verification phase not implemented yet.

**Maxima** Verification phase not implemented yet.

**Fricas** Verification phase not implemented yet.

**Sympy** Verification phase not implemented yet.

**Giac** Verification phase not implemented yet.

**Mupad** Verification phase not implemented yet.

## 1.7 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.

## 1.8 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.

## 1.9 Important notes about some of the results

### 1.9.1 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.

### 1.9.2 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.

### 1.9.3 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
```

### 1.9.4 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  $\sin(x)^2/2$



## 1.10 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 "F"
13. String. Small string description of why the grade was given.
14. integer. 1 if result was verified or 0 if not verified.

*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



# Chapter 2

## detailed summary tables of results

### Local contents

2.1	List of integrals sorted by grade for each CAS . . . . .	20
2.2	Detailed conclusion table per each integral for all CAS systems . . . . .	23
2.3	Detailed conclusion table specific for Rubi results . . . . .	27

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

### Local contents

2.1.1	Rubi . . . . .	21
2.1.2	Mathematica . . . . .	21
2.1.3	Maple . . . . .	21
2.1.4	Maxima . . . . .	21
2.1.5	FriCAS . . . . .	21
2.1.6	Sympy . . . . .	22
2.1.7	Giac . . . . .	22
2.1.8	Mupad . . . . .	22

### 2.1.1 Rubi

A grade: { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 }

B grade: { }

C grade: { }

F grade: { }

### 2.1.2 Mathematica

A grade: { 1, 2, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 }

B grade: { 3 }

C grade: { 6 }

F grade: { }

### 2.1.3 Maple

A grade: { 3, 4, 7, 8, 12 }

B grade: { 5, 6, 11 }

C grade: { }

F grade: { 1, 2, 9, 10, 13, 14, 15, 16 }

### 2.1.4 Maxima

A grade: { 4, 8, 12 }

B grade: { 5, 7 }

C grade: { }

F grade: { 1, 2, 3, 6, 9, 10, 11, 13, 14, 15, 16 }

### 2.1.5 FriCAS

A grade: { 4, 8, 12 }

B grade: { 1, 2, 3, 7, 9, 10, 11 }

C grade: { 5, 6 }

F grade: { 13, 14, 15, 16 }

### 2.1.6 Sympy

A grade: { 4, 8, 12 }

B grade: { }

C grade: { }

F grade: { 1, 2, 3, 5, 6, 7, 9, 10, 11, 13, 14, 15, 16 }

### 2.1.7 Giac

A grade: { 4, 8, 12 }

B grade: { 7 }

C grade: { }

F grade: { 1, 2, 3, 5, 6, 9, 10, 11, 13, 14, 15, 16 }

### 2.1.8 Mupad

A grade: { 4, 8, 12 }

B grade: { 7 }

C grade: { }

F grade: { 1, 2, 3, 5, 6, 9, 10, 11, 13, 14, 15, 16 }

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

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

In this table, the column **N.S.** in the table below, which stands for **normalized size** is defined as  $\frac{\text{antiderivative leaf size}}{\text{optimal antiderivative leaf size}}$ . To help make the table fit, **Mathematica** was abbrevi-

	Problem 1	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
viated to <b>MMA</b> .	grade	A	A	A	F	F	B	F	F	F
	verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
	size	179	179	343	0	0	497	0	0	-1
	N.S.	1	1.00	1.92	0.00	0.00	2.78	0.00	0.00	-0.01
	time (sec)	N/A	0.096	0.155	180.000	0.000	0.398	0.000	0.000	0.000

Problem 2	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	A	A	A	F	F	B	F	F	F
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	119	119	199	0	0	305	0	0	-1
N.S.	1	1.00	1.67	0.00	0.00	2.56	0.00	0.00	-0.01
time (sec)	N/A	0.062	0.073	180.000	0.000	0.367	0.000	0.000	0.000

Problem 3	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	A	A	B	A	F	B	F	F	F
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	61	61	127	101	0	157	0	0	-1
N.S.	1	1.00	2.08	1.66	0.00	2.57	0.00	0.00	-0.02
time (sec)	N/A	0.028	0.100	0.500	0.000	0.402	0.000	0.000	0.000

Problem 4	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	N/A	A	A	A	A	A	A	A	A
verified	N/A	N/A	N/A	TBD	TBD	TBD	TBD	TBD	TBD
size	17	0	0	0	0	0	0	0	-1
N.S.	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.06
time (sec)	N/A	0.015	0.245	180.000	0.000	0.000	0.000	0.000	0.000

Problem 5	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	A	A	A	B	B	C	F	F	F
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	103	103	135	298	238	1332	0	0	-1
N.S.	1	1.00	1.31	2.89	2.31	12.93	0.00	0.00	-0.01
time (sec)	N/A	0.149	0.389	0.888	0.393	0.431	0.000	0.000	0.000

Problem 6	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	A	A	C	B	F	C	F	F	F
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	73	73	186	159	0	715	0	0	-1
N.S.	1	1.00	2.55	2.18	0.00	9.79	0.00	0.00	-0.01
time (sec)	N/A	0.097	4.932	0.886	0.000	0.374	0.000	0.000	0.000

Problem 7	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	A	A	A	A	B	B	F	B	B
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	29	29	51	57	72	161	0	78	50
N.S.	1	1.00	1.76	1.97	2.48	5.55	0.00	2.69	1.72
time (sec)	N/A	0.023	0.082	0.823	0.264	0.368	0.000	0.392	0.086

Problem 8	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	N/A	A	A	A	A	A	A	A	A
verified	N/A	N/A	N/A	TBD	TBD	TBD	TBD	TBD	TBD
size	19	0	0	0	0	0	0	0	-1
N.S.	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.05
time (sec)	N/A	0.026	5.675	180.000	0.000	0.000	0.000	0.000	0.000



Problem 9	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	A	A	A	F	F	B	F	F	F
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	296	296	455	0	0	4785	0	0	-1
N.S.	1	1.00	1.54	0.00	0.00	16.17	0.00	0.00	-0.00
time (sec)	N/A	0.155	4.019	0.071	0.000	0.469	0.000	0.000	0.000

Problem 10	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	A	A	A	F	F	B	F	F	F
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	175	175	270	0	0	2651	0	0	-1
N.S.	1	1.00	1.54	0.00	0.00	15.15	0.00	0.00	-0.01
time (sec)	N/A	0.099	1.371	0.073	0.000	0.437	0.000	0.000	0.000

Problem 11	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	A	A	A	B	F	B	F	F	F
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	102	102	178	216	0	1267	0	0	-1
N.S.	1	1.00	1.75	2.12	0.00	12.42	0.00	0.00	-0.01
time (sec)	N/A	0.047	2.109	0.991	0.000	0.387	0.000	0.000	0.000

Problem 12	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	N/A	A	A	A	A	A	A	A	A
verified	N/A	N/A	N/A	TBD	TBD	TBD	TBD	TBD	TBD
size	19	0	0	0	0	0	0	0	-1
N.S.	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.05
time (sec)	N/A	0.028	65.642	180.000	0.000	0.000	0.000	0.000	0.000

Problem 13	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	A	A	A	F	F	F(-2)	F	F	F
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	24	24	17	0	0	0	0	0	-1
N.S.	1	1.00	0.71	0.00	0.00	0.00	0.00	0.00	-0.04
time (sec)	N/A	0.062	0.054	0.070	0.000	0.000	0.000	0.000	0.000

Problem 14	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	A	A	A	F	F	F(-2)	F	F	F
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	24	24	17	0	0	0	0	0	-1
N.S.	1	1.00	0.71	0.00	0.00	0.00	0.00	0.00	-0.04
time (sec)	N/A	0.058	0.073	0.066	0.000	0.000	0.000	0.000	0.000

Problem 15	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	A	A	A	F	F	F(-2)	F	F	F
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	47	47	45	0	0	0	0	0	-1
N.S.	1	1.00	0.96	0.00	0.00	0.00	0.00	0.00	-0.02
time (sec)	N/A	0.074	0.071	0.066	0.000	0.000	0.000	0.000	0.000

Problem 16	Optimal	Rubi	MMA	Maple	Maxima	Fricas	Sympy	Giac	Mupad
grade	A	A	A	F	F	F(-2)	F	F	F
verified	N/A	Yes	Yes	TBD	TBD	TBD	TBD	TBD	TBD
size	66	66	55	0	0	0	0	0	-1
N.S.	1	1.00	0.83	0.00	0.00	0.00	0.00	0.00	-0.02
time (sec)	N/A	0.115	0.053	0.065	0.000	0.000	0.000	0.000	0.000

## 2.3 Detailed conclusion table specific for Rubi results

The following table is specific to Rubi. It gives additional statistics for each integral. the column **steps** is the number of steps used by Rubi to obtain the antiderivative. The **rules** column is the number of unique rules used. The **integrand size** column is the leaf size of the integrand. Finally the ratio  $\frac{\text{number of rules}}{\text{integrand size}}$  is given. The larger this ratio is, the harder the integral was to solve. In this test, problem number [16] had the largest ratio of [24]

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	9	5	1.00	14	0.357
2	A	7	4	1.00	14	0.286
3	A	5	3	1.00	12	0.250
4	A	0	0	0.00	0	0.000
5	A	6	6	1.00	16	0.375
6	A	5	5	1.00	16	0.312
7	A	2	2	1.00	14	0.143
8	A	0	0	0.00	0	0.000
9	A	15	8	1.00	16	0.500
10	A	9	6	1.00	16	0.375
11	A	6	4	1.00	14	0.286
12	A	0	0	0.00	0	0.000
13	A	4	2	1.00	20	0.100
14	A	4	2	1.00	20	0.100
15	A	5	2	1.00	20	0.100
16	A	7	5	1.00	24	0.208



# Chapter 3

## Listing of integrals

### Local contents

3.1	$\int (c + dx)^3 \operatorname{sech}(a + bx) dx$	30
3.2	$\int (c + dx)^2 \operatorname{sech}(a + bx) dx$	34
3.3	$\int (c + dx) \operatorname{sech}(a + bx) dx$	38
3.4	$\int \frac{\operatorname{sech}(a+bx)}{c+dx} dx$	41
3.5	$\int (c + dx)^3 \operatorname{sech}^2(a + bx) dx$	44
3.6	$\int (c + dx)^2 \operatorname{sech}^2(a + bx) dx$	49
3.7	$\int (c + dx) \operatorname{sech}^2(a + bx) dx$	53
3.8	$\int \frac{\operatorname{sech}^2(a+bx)}{c+dx} dx$	56
3.9	$\int (c + dx)^3 \operatorname{sech}^3(a + bx) dx$	59
3.10	$\int (c + dx)^2 \operatorname{sech}^3(a + bx) dx$	65
3.11	$\int (c + dx) \operatorname{sech}^3(a + bx) dx$	71
3.12	$\int \frac{\operatorname{sech}^3(a+bx)}{c+dx} dx$	75
3.13	$\int \left( \frac{x}{\operatorname{sech}^{\frac{3}{2}}(x)} - \frac{1}{3}x \sqrt{\operatorname{sech}(x)} \right) dx$	78
3.14	$\int \left( \frac{x}{\operatorname{sech}^{\frac{5}{2}}(x)} - \frac{3x}{5\sqrt{\operatorname{sech}(x)}} \right) dx$	81
3.15	$\int \left( \frac{x}{\operatorname{sech}^{\frac{7}{2}}(x)} - \frac{5}{21}x \sqrt{\operatorname{sech}(x)} \right) dx$	84
3.16	$\int \left( \frac{x^2}{\operatorname{sech}^{\frac{3}{2}}(x)} - \frac{1}{3}x^2 \sqrt{\operatorname{sech}(x)} \right) dx$	87

### 3.1 $\int (c + dx)^3 \operatorname{sech}(a + bx) dx$

**Optimal.** Leaf size=179

$$\frac{2(c + dx)^3 \operatorname{ArcTan}(e^{a+bx})}{b} - \frac{3id(c + dx)^2 \operatorname{PolyLog}(2, -ie^{a+bx})}{b^2} + \frac{3id(c + dx)^2 \operatorname{PolyLog}(2, ie^{a+bx})}{b^2} + \frac{6id^2(c + dx)}{b^2}$$

[Out]  $2*(d*x+c)^3*\arctan(\exp(b*x+a))/b-3*I*d*(d*x+c)^2*\operatorname{polylog}(2,-I*\exp(b*x+a))/b^2+3*I*d*(d*x+c)^2*\operatorname{polylog}(2,I*\exp(b*x+a))/b^2+6*I*d^2*(d*x+c)*\operatorname{polylog}(3,-I*\exp(b*x+a))/b^3-6*I*d^2*(d*x+c)*\operatorname{polylog}(3,I*\exp(b*x+a))/b^3-6*I*d^3*\operatorname{polylog}(4,-I*\exp(b*x+a))/b^4+6*I*d^3*\operatorname{polylog}(4,I*\exp(b*x+a))/b^4$

**Rubi [A]**

time = 0.10, antiderivative size = 179, normalized size of antiderivative = 1.00, number of steps used = 9, number of rules used = 5, integrand size = 14,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.357$ , Rules used = {4265, 2611, 6744, 2320, 6724}

$$\frac{2(c + dx)^3 \operatorname{ArcTan}(e^{a+bx})}{b} - \frac{6id^3 \operatorname{Li}_4(-ie^{a+bx})}{b^4} + \frac{6id^3 \operatorname{Li}_4(ie^{a+bx})}{b^4} + \frac{6id^2(c + dx) \operatorname{Li}_3(-ie^{a+bx})}{b^3} - \frac{6id^2(c + dx) \operatorname{Li}_3(ie^{a+bx})}{b^3} - \frac{3id(c + dx)^2 \operatorname{Li}_2(-ie^{a+bx})}{b^2} + \frac{3id(c + dx)^2 \operatorname{Li}_2(ie^{a+bx})}{b^2}$$

Antiderivative was successfully verified.

[In]  $\operatorname{Int}[(c + d*x)^3*\operatorname{Sech}[a + b*x], x]$

[Out]  $(2*(c + d*x)^3*\operatorname{ArcTan}[E^{(a + b*x)}])/b - ((3*I)*d*(c + d*x)^2*\operatorname{PolyLog}[2, (-I)*E^{(a + b*x)}])/b^2 + ((3*I)*d*(c + d*x)^2*\operatorname{PolyLog}[2, I*E^{(a + b*x)}])/b^2 + ((6*I)*d^2*(c + d*x)*\operatorname{PolyLog}[3, (-I)*E^{(a + b*x)}])/b^3 - ((6*I)*d^2*(c + d*x)*\operatorname{PolyLog}[3, I*E^{(a + b*x)}])/b^3 - ((6*I)*d^3*\operatorname{PolyLog}[4, (-I)*E^{(a + b*x)}])/b^4 + ((6*I)*d^3*\operatorname{PolyLog}[4, I*E^{(a + b*x)}])/b^4$

**Rule 2320**

```
Int[u_, x_Symbol] := With[{v = FunctionOfExponential[u, x]}, Dist[v/D[v, x], Subst[Int[FunctionOfExponentialFunction[u, x]/x, x], x, v], x] /; FunctionOfExponentialQ[u, x] && !MatchQ[u, (w_)*((a_.)*(v_)^(n_))^(m_)] /; FreeQ[{a, m, n}, x] && IntegerQ[m*n] && !MatchQ[u, E^((c_.)*((a_.) + (b_.)*x))*(F_)[v_] /; FreeQ[{a, b, c}, x] && InverseFunctionQ[F[x]]]
```

**Rule 2611**

```
Int[Log[1 + (e_.)*((F_)^((c_.)*((a_.) + (b_.)*(x_)))^(n_.))*((f_.) + (g_.)*(x_))^(m_.), x_Symbol] := Simp[(-f + g*x)^m*(PolyLog[2, (-e)*(F^(c*(a + b*x)))^n]/(b*c*n*Log[F])), x] + Dist[g*(m/(b*c*n*Log[F])), Int[(f + g*x)^(m-1)*PolyLog[2, (-e)*(F^(c*(a + b*x)))^n], x], x] /; FreeQ[{F, a, b, c, e, f, g, n}, x] && GtQ[m, 0]
```

**Rule 4265**



```
[Out] (I*((-2*I)*b^3*c^3*ArcTan[E^(a + b*x)] + 3*b^3*c^2*d*x*Log[1 - I*E^(a + b*x)] + 3*b^3*c*d^2*x^2*Log[1 - I*E^(a + b*x)] + b^3*d^3*x^3*Log[1 - I*E^(a + b*x)] - 3*b^3*c^2*d*x*Log[1 + I*E^(a + b*x)] - 3*b^3*c*d^2*x^2*Log[1 + I*E^(a + b*x)] - b^3*d^3*x^3*Log[1 + I*E^(a + b*x)] - 3*b^2*d*(c + d*x)^2*PolyLog[2, (-I)*E^(a + b*x)] + 3*b^2*d*(c + d*x)^2*PolyLog[2, I*E^(a + b*x)] + 6*b*c*d^2*PolyLog[3, (-I)*E^(a + b*x)] + 6*b*d^3*x*PolyLog[3, (-I)*E^(a + b*x)] - 6*b*c*d^2*PolyLog[3, I*E^(a + b*x)] - 6*b*d^3*x*PolyLog[3, I*E^(a + b*x)] - 6*d^3*PolyLog[4, (-I)*E^(a + b*x)] + 6*d^3*PolyLog[4, I*E^(a + b*x)])/b^4
```

**Maple** [F]

time = 180.00, size = 0, normalized size = 0.00

$$\int (dx + c)^3 \operatorname{sech}(bx + a) dx$$

Verification of antiderivative is not currently implemented for this CAS.

```
[In] int((d*x+c)^3*sech(b*x+a),x)
```

```
[Out] int((d*x+c)^3*sech(b*x+a),x)
```

**Maxima** [F]

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

```
[In] integrate((d*x+c)^3*sech(b*x+a),x, algorithm="maxima")
```

```
[Out] -2*c^3*arctan(e^(-b*x - a))/b + 2*integrate((d^3*x^3*e^a + 3*c*d^2*x^2*e^a + 3*c^2*d*x*e^a)*e^(b*x)/(e^(2*b*x + 2*a) + 1), x)
```

**Fricas** [B] Both result and optimal contain complex but leaf count of result is larger than twice the leaf count of optimal. 497 vs. 2(146) = 292.

time = 0.40, size = 497, normalized size = 2.78

Verification of antiderivative is not currently implemented for this CAS.

```
[In] integrate((d*x+c)^3*sech(b*x+a),x, algorithm="fricas")
```

```
[Out] (6*I*d^3*polylog(4, I*cosh(b*x + a) + I*sinh(b*x + a)) - 6*I*d^3*polylog(4, -I*cosh(b*x + a) - I*sinh(b*x + a)) - 3*(-I*b^2*d^3*x^2 - 2*I*b^2*c*d^2*x - I*b^2*c^2*d)*dilog(I*cosh(b*x + a) + I*sinh(b*x + a)) - 3*(I*b^2*d^3*x^2 + 2*I*b^2*c*d^2*x + I*b^2*c^2*d)*dilog(-I*cosh(b*x + a) - I*sinh(b*x + a)) + (I*b^3*c^3 - 3*I*a*b^2*c^2*d + 3*I*a^2*b*c*d^2 - I*a^3*d^3)*log(cosh(b*x
```



+ a) + sinh(b\*x + a) + I) + (-I\*b^3\*c^3 + 3\*I\*a\*b^2\*c^2\*d - 3\*I\*a^2\*b\*c\*d^2 + I\*a^3\*d^3)\*log(cosh(b\*x + a) + sinh(b\*x + a) - I) + (-I\*b^3\*d^3\*x^3 - 3\*I\*b^3\*c\*d^2\*x^2 - 3\*I\*b^3\*c^2\*d\*x - 3\*I\*a\*b^2\*c^2\*d + 3\*I\*a^2\*b\*c\*d^2 - I\*a^3\*d^3)\*log(I\*cosh(b\*x + a) + I\*sinh(b\*x + a) + 1) + (I\*b^3\*d^3\*x^3 + 3\*I\*b^3\*c\*d^2\*x^2 + 3\*I\*b^3\*c^2\*d\*x + 3\*I\*a\*b^2\*c^2\*d - 3\*I\*a^2\*b\*c\*d^2 + I\*a^3\*d^3)\*log(-I\*cosh(b\*x + a) - I\*sinh(b\*x + a) + 1) - 6\*(I\*b\*d^3\*x + I\*b\*c\*d^2)\*polylog(3, I\*cosh(b\*x + a) + I\*sinh(b\*x + a)) - 6\*(-I\*b\*d^3\*x - I\*b\*c\*d^2)\*polylog(3, -I\*cosh(b\*x + a) - I\*sinh(b\*x + a))/b^4

**Sympy** [F]

time = 0.00, size = 0, normalized size = 0.00

$$\int (c + dx)^3 \operatorname{sech}(a + bx) dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*\*3\*sech(b\*x+a),x)

[Out] Integral((c + d\*x)\*\*3\*sech(a + b\*x), x)

**Giac** [F]

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)^3\*sech(b\*x+a),x, algorithm="giac")

[Out] integrate((d\*x + c)^3\*sech(b\*x + a), x)

**Mupad** [F]

time = 0.00, size = -1, normalized size = -0.01

$$\int \frac{(c + dx)^3}{\cosh(a + bx)} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((c + d\*x)^3/cosh(a + b\*x),x)

[Out] int((c + d\*x)^3/cosh(a + b\*x), x)

## 3.2 $\int (c + dx)^2 \operatorname{sech}(a + bx) dx$

**Optimal.** Leaf size=119

$$\frac{2(c + dx)^2 \operatorname{ArcTan}(e^{a+bx})}{b} - \frac{2id(c + dx) \operatorname{PolyLog}(2, -ie^{a+bx})}{b^2} + \frac{2id(c + dx) \operatorname{PolyLog}(2, ie^{a+bx})}{b^2} + \frac{2id^2 \operatorname{PolyLog}(3, -I \exp(b*x+a))}{b^3}$$

[Out]  $2*(d*x+c)^2*\arctan(\exp(b*x+a))/b-2*I*d*(d*x+c)*\operatorname{polylog}(2,-I*\exp(b*x+a))/b^2+2*I*d*(d*x+c)*\operatorname{polylog}(2,I*\exp(b*x+a))/b^2+2*I*d^2*\operatorname{polylog}(3,-I*\exp(b*x+a))/b^3-2*I*d^2*\operatorname{polylog}(3,I*\exp(b*x+a))/b^3$

**Rubi [A]**

time = 0.06, antiderivative size = 119, normalized size of antiderivative = 1.00, number of steps used = 7, number of rules used = 4, integrand size = 14,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.286$ ,

Rules used = {4265, 2611, 2320, 6724}

$$\frac{2(c + dx)^2 \operatorname{ArcTan}(e^{a+bx})}{b} + \frac{2id^2 \operatorname{Li}_3(-ie^{a+bx})}{b^3} - \frac{2id^2 \operatorname{Li}_3(ie^{a+bx})}{b^3} - \frac{2id(c + dx) \operatorname{Li}_2(-ie^{a+bx})}{b^2} + \frac{2id(c + dx) \operatorname{Li}_2(ie^{a+bx})}{b^2}$$

Antiderivative was successfully verified.

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

[Out]  $(2*(c + d*x)^2*\operatorname{ArcTan}[E^{(a + b*x)}])/b - ((2*I)*d*(c + d*x)*\operatorname{PolyLog}[2, (-I)*E^{(a + b*x)}])/b^2 + ((2*I)*d*(c + d*x)*\operatorname{PolyLog}[2, I*E^{(a + b*x)}])/b^2 + ((2*I)*d^2*\operatorname{PolyLog}[3, (-I)*E^{(a + b*x)}])/b^3 - ((2*I)*d^2*\operatorname{PolyLog}[3, I*E^{(a + b*x)}])/b^3$

Rule 2320

```
Int[u_, x_Symbol] := With[{v = FunctionOfExponential[u, x]}, Dist[v/D[v, x], Subst[Int[FunctionOfExponentialFunction[u, x]/x, x], x, v], x] /; FunctionOfExponentialQ[u, x] && !MatchQ[u, (w_)*((a_.)*(v_)^(n_.))^(m_) /; FreeQ[{a, m, n}, x] && IntegerQ[m*n]] && !MatchQ[u, E^((c_.)*((a_.) + (b_.)*x))*(F_) [v_] /; FreeQ[{a, b, c}, x] && InverseFunctionQ[F[x]]]
```

Rule 2611

```
Int[Log[1 + (e_.)*((F_)^((c_.)*((a_.) + (b_.)*(x_)))^(n_.))*((f_.) + (g_.)*(x_))^(m_.), x_Symbol] := Simp[(-f + g*x)^m*(PolyLog[2, (-e)*(F^(c*(a + b*x)))^n]/(b*c*n*Log[F])), x] + Dist[g*(m/(b*c*n*Log[F])), Int[(f + g*x)^(m-1)*PolyLog[2, (-e)*(F^(c*(a + b*x)))^n], x], x] /; FreeQ[{F, a, b, c, e, f, g, n}, x] && GtQ[m, 0]
```

Rule 4265

```
Int[csc[(e_.) + Pi*(k_.) + (Complex[0, fz_])*(f_.)*(x_)]*((c_.) + (d_.)*(x_))^(m_.), x_Symbol] := Simp[-2*(c + d*x)^m*(ArcTanh[E^((-I)*e + f*fz*x)/E^((
```

```
I**Pi)]/(f*fz*I)), x] + (-Dist[d*(m/(f*fz*I)), Int[(c + d*x)^(m - 1)*Log[1
- E^((-I)*e + f*fz*x)/E^(I**Pi)], x], x] + Dist[d*(m/(f*fz*I)), Int[(c +
d*x)^(m - 1)*Log[1 + E^((-I)*e + f*fz*x)/E^(I**Pi)], x], x]) /; FreeQ[{c,
d, e, f, fz}, x] && IntegerQ[2*k] && IGtQ[m, 0]
```

### Rule 6724

```
Int[PolyLog[n_, (c_.)*((a_.) + (b_.)*(x_.))^(p_.)]/((d_.) + (e_.)*(x_.)), x_S
ymbol] := Simp[PolyLog[n + 1, c*(a + b*x)^p]/(e*p), x] /; FreeQ[{a, b, c, d
, e, n, p}, x] && EqQ[b*d, a*e]
```

### Rubi steps

$$\begin{aligned} \int (c + dx)^2 \operatorname{sech}(a + bx) dx &= \frac{2(c + dx)^2 \tan^{-1}(e^{a+bx})}{b} - \frac{(2id) \int (c + dx) \log(1 - ie^{a+bx}) dx}{b} + \frac{(2id) \int (c + dx) \log(1 + ie^{a+bx}) dx}{b} \\ &= \frac{2(c + dx)^2 \tan^{-1}(e^{a+bx})}{b} - \frac{2id(c + dx) \operatorname{Li}_2(-ie^{a+bx})}{b^2} + \frac{2id(c + dx) \operatorname{Li}_2(ie^{a+bx})}{b^2} \\ &= \frac{2(c + dx)^2 \tan^{-1}(e^{a+bx})}{b} - \frac{2id(c + dx) \operatorname{Li}_2(-ie^{a+bx})}{b^2} + \frac{2id(c + dx) \operatorname{Li}_2(ie^{a+bx})}{b^2} \\ &= \frac{2(c + dx)^2 \tan^{-1}(e^{a+bx})}{b} - \frac{2id(c + dx) \operatorname{Li}_2(-ie^{a+bx})}{b^2} + \frac{2id(c + dx) \operatorname{Li}_2(ie^{a+bx})}{b^2} \end{aligned}$$

### Mathematica [A]

time = 0.07, size = 199, normalized size = 1.67

$$\frac{i(-2b^2c^2 \operatorname{ArcTan}(e^{a+bx}) + 2b^2cdx \log(1 - ie^{a+bx}) + b^2d^2x^2 \log(1 - ie^{a+bx}) - 2b^2cdx \log(1 + ie^{a+bx}) - b^2d^2x^2 \log(1 + ie^{a+bx}) - 2bd(c + dx) \operatorname{PolyLog}(2, -ie^{a+bx}) + 2bd(c + dx) \operatorname{PolyLog}(2, ie^{a+bx}) + 2d^2 \operatorname{PolyLog}(3, -ie^{a+bx}) - 2d^2 \operatorname{PolyLog}(3, ie^{a+bx}))}{b^3}$$

Antiderivative was successfully verified.

```
[In] Integrate[(c + d*x)^2*Sech[a + b*x],x]
```

```
[Out] (I*((-2*I)*b^2*c^2*ArcTan[E^(a + b*x)] + 2*b^2*c*d*x*Log[1 - I*E^(a + b*x)]
+ b^2*d^2*x^2*Log[1 - I*E^(a + b*x)] - 2*b^2*c*d*x*Log[1 + I*E^(a + b*x)]
- b^2*d^2*x^2*Log[1 + I*E^(a + b*x)] - 2*b*d*(c + d*x)*PolyLog[2, (-I)*E^(a
+ b*x)] + 2*b*d*(c + d*x)*PolyLog[2, I*E^(a + b*x)] + 2*d^2*PolyLog[3, (-I
)*E^(a + b*x)] - 2*d^2*PolyLog[3, I*E^(a + b*x)]))/b^3
```

### Maple [F]

time = 180.00, size = 0, normalized size = 0.00

$$\int (dx + c)^2 \operatorname{sech}(bx + a) dx$$

Verification of antiderivative is not currently implemented for this CAS.



Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)^2\*sech(b\*x+a),x, algorithm="giac")

[Out] integrate((d\*x + c)^2\*sech(b\*x + a), x)

**Mupad [F]**

time = 0.00, size = -1, normalized size = -0.01

$$\int \frac{(c + dx)^2}{\cosh(a + bx)} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((c + d\*x)^2/cosh(a + b\*x),x)

[Out] int((c + d\*x)^2/cosh(a + b\*x), x)

### 3.3 $\int (c + dx)\operatorname{sech}(a + bx) dx$

Optimal. Leaf size=61

$$\frac{2(c + dx)\operatorname{ArcTan}(e^{a+bx})}{b} - \frac{id\operatorname{PolyLog}(2, -ie^{a+bx})}{b^2} + \frac{id\operatorname{PolyLog}(2, ie^{a+bx})}{b^2}$$

[Out]  $2*(d*x+c)*\arctan(\exp(b*x+a))/b-I*d*\operatorname{polylog}(2,-I*\exp(b*x+a))/b^2+I*d*\operatorname{polylog}(2,I*\exp(b*x+a))/b^2$

Rubi [A]

time = 0.03, antiderivative size = 61, normalized size of antiderivative = 1.00, number of steps used = 5, number of rules used = 3, integrand size = 12,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.250$ , Rules used = {4265, 2317, 2438}

$$\frac{2(c + dx)\operatorname{ArcTan}(e^{a+bx})}{b} - \frac{id\operatorname{Li}_2(-ie^{a+bx})}{b^2} + \frac{id\operatorname{Li}_2(ie^{a+bx})}{b^2}$$

Antiderivative was successfully verified.

[In] `Int[(c + d*x)*Sech[a + b*x],x]`

[Out]  $(2*(c + d*x)*\operatorname{ArcTan}[E^{(a + b*x)}])/b - (I*d*\operatorname{PolyLog}[2, (-I)*E^{(a + b*x)}])/b^2 + (I*d*\operatorname{PolyLog}[2, I*E^{(a + b*x)}])/b^2$

Rule 2317

```
Int[Log[(a_) + (b_.)*((F_)^((e_.)*((c_.) + (d_.)*(x_))))^(n_.)], x_Symbol]
:> Dist[1/(d*e*n*Log[F]), Subst[Int[Log[a + b*x]/x, x], x, (F^(e*(c + d*x)))^n], x] /; FreeQ[{F, a, b, c, d, e, n}, x] && GtQ[a, 0]
```

Rule 2438

```
Int[Log[(c_.)*((d_) + (e_.)*(x_)^(n_.))]/(x_), x_Symbol] :> Simp[-PolyLog[2, (-c)*e*x^n]/n, x] /; FreeQ[{c, d, e, n}, x] && EqQ[c*d, 1]
```

Rule 4265

```
Int[csc[(e_.) + Pi*(k_.) + (Complex[0, fz_])*(f_.)*(x_)]*((c_.) + (d_.)*(x_))^(m_.), x_Symbol] :> Simp[-2*(c + d*x)^m*(ArcTanh[E^((-I)*e + f*fz*x)/E^(I*k*Pi)]/(f*fz*I)), x] + (-Dist[d*(m/(f*fz*I)), Int[(c + d*x)^(m - 1)*Log[1 - E^((-I)*e + f*fz*x)/E^(I*k*Pi)], x], x] + Dist[d*(m/(f*fz*I)), Int[(c + d*x)^(m - 1)*Log[1 + E^((-I)*e + f*fz*x)/E^(I*k*Pi)], x], x]) /; FreeQ[{c, d, e, f, fz}, x] && IntegerQ[2*k] && IGtQ[m, 0]
```

Rubi steps

$$\begin{aligned}
\int (c + dx) \operatorname{sech}(a + bx) dx &= \frac{2(c + dx) \tan^{-1}(e^{a+bx})}{b} - \frac{(id) \int \log(1 - ie^{a+bx}) dx}{b} + \frac{(id) \int \log(1 + ie^{a+bx})}{b} \\
&= \frac{2(c + dx) \tan^{-1}(e^{a+bx})}{b} - \frac{(id) \operatorname{Subst}\left(\int \frac{\log(1-ix)}{x} dx, x, e^{a+bx}\right)}{b^2} + \frac{(id) \operatorname{Subst}\left(\int \frac{\log(1+ix)}{x} dx, x, e^{a+bx}\right)}{b^2} \\
&= \frac{2(c + dx) \tan^{-1}(e^{a+bx})}{b} - \frac{id \operatorname{Li}_2(-ie^{a+bx})}{b^2} + \frac{id \operatorname{Li}_2(ie^{a+bx})}{b^2}
\end{aligned}$$

**Mathematica [B]** Both result and optimal contain complex but leaf count is larger than twice the leaf count of optimal. 127 vs.  $2(61) = 122$ .

time = 0.10, size = 127, normalized size = 2.08

$$\frac{bc \operatorname{ArcTan}(\sinh(a + bx)) + \frac{1}{2} d \left( -((-2ia + \pi - 2ibx) (\log(1 - ie^{a+bx}) - \log(1 + ie^{a+bx}))) + (-2ia + \pi) \log(\cot(\frac{1}{4}(2ia + \pi + 2ibx))) \right) - 2i(\operatorname{PolyLog}(2, -ie^{a+bx}) - \operatorname{PolyLog}(2, ie^{a+bx}))}{b^2}$$

Antiderivative was successfully verified.

[In] Integrate[(c + d\*x)\*Sech[a + b\*x], x]

[Out] (b\*c\*ArcTan[Sinh[a + b\*x]] + (d\*(-(((2\*I)\*a + Pi - (2\*I)\*b\*x)\*(Log[1 - I\*E^(a + b\*x)] - Log[1 + I\*E^(a + b\*x)])) + ((-2\*I)\*a + Pi)\*Log[Cot[((2\*I)\*a + Pi + (2\*I)\*b\*x)/4]] - (2\*I)\*(PolyLog[2, (-I)\*E^(a + b\*x)] - PolyLog[2, I\*E^(a + b\*x)])))/2)/b^2

**Maple [A]**

time = 0.50, size = 101, normalized size = 1.66

method	result
derivativedivides	$\frac{d(i(bx+a)(\ln(1-ie^{bx+a})-\ln(1+ie^{bx+a}))-i \operatorname{dilog}(1+ie^{bx+a})+i \operatorname{dilog}(1-ie^{bx+a}))}{b} - \frac{2da \arctan(e^{bx+a})}{b} + 2c \arctan(e^{bx+a})$
default	$\frac{d(i(bx+a)(\ln(1-ie^{bx+a})-\ln(1+ie^{bx+a}))-i \operatorname{dilog}(1+ie^{bx+a})+i \operatorname{dilog}(1-ie^{bx+a}))}{b} - \frac{2da \arctan(e^{bx+a})}{b} + 2c \arctan(e^{bx+a})$
risch	$\frac{2c \arctan(e^{bx+a})}{b} - \frac{id \ln(1+ie^{bx+a})x}{b} - \frac{id \ln(1+ie^{bx+a})a}{b^2} + \frac{id \ln(1-ie^{bx+a})x}{b} + \frac{id \ln(1-ie^{bx+a})a}{b^2} - \frac{id \operatorname{dilog}(1+ie^{bx+a})}{b^2} + \frac{id \operatorname{dilog}(1-ie^{bx+a})}{b^2}$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((d\*x+c)\*sech(b\*x+a), x, method=\_RETURNVERBOSE)

[Out] 1/b\*(d/b\*(I\*(b\*x+a)\*(ln(1-I\*exp(b\*x+a))-ln(1+I\*exp(b\*x+a)))-I\*dilog(1+I\*exp(b\*x+a))+I\*dilog(1-I\*exp(b\*x+a)))-2\*d/b\*a\*arctan(exp(b\*x+a))+2\*c\*arctan(exp(b\*x+a)))

**Maxima [F]**

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*sech(b\*x+a),x, algorithm="maxima")

[Out] 2\*d\*integrate(x\*e^(b\*x + a)/(e^(2\*b\*x + 2\*a) + 1), x) - 2\*c\*arctan(e^(-b\*x - a))/b

**Fricas** [B] Both result and optimal contain complex but leaf count of result is larger than twice the leaf count of optimal. 157 vs. 2(48) = 96.

time = 0.40, size = 157, normalized size = 2.57

$i dL_2(i \cosh(bx+a) + i \sinh(bx+a)) - i dL_2(-i \cosh(bx+a) - i \sinh(bx+a)) + (ibc - i ad) \log(\cosh(bx+a) + \sinh(bx+a) + 1) + (-ibc + i ad) \log(\cosh(bx+a) + \sinh(bx+a) - 1) + (-ibdx - i ad) \log(i \cosh(bx+a) + i \sinh(bx+a) + 1) + (ibdx + i ad) \log(-i \cosh(bx+a) - i \sinh(bx+a) + 1)$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*sech(b\*x+a),x, algorithm="fricas")

[Out] (I\*d\*dilog(I\*cosh(b\*x + a) + I\*sinh(b\*x + a)) - I\*d\*dilog(-I\*cosh(b\*x + a) - I\*sinh(b\*x + a)) + (I\*b\*c - I\*a\*d)\*log(cosh(b\*x + a) + sinh(b\*x + a) + I) + (-I\*b\*c + I\*a\*d)\*log(cosh(b\*x + a) + sinh(b\*x + a) - I) + (-I\*b\*d\*x - I\*a\*d)\*log(I\*cosh(b\*x + a) + I\*sinh(b\*x + a) + 1) + (I\*b\*d\*x + I\*a\*d)\*log(-I\*cosh(b\*x + a) - I\*sinh(b\*x + a) + 1))/b^2

**Sympy** [F]

time = 0.00, size = 0, normalized size = 0.00

$$\int (c + dx) \operatorname{sech}(a + bx) dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*sech(b\*x+a),x)

[Out] Integral((c + d\*x)\*sech(a + b\*x), x)

**Giac** [F]

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*sech(b\*x+a),x, algorithm="giac")

[Out] integrate((d\*x + c)\*sech(b\*x + a), x)

**Mupad** [F]

time = 0.00, size = -1, normalized size = -0.02

$$\int \frac{c + dx}{\cosh(a + bx)} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((c + d\*x)/cosh(a + b\*x),x)

[Out] int((c + d\*x)/cosh(a + b\*x), x)



### 3.4 $\int \frac{\operatorname{sech}(a+bx)}{c+dx} dx$

Optimal. Leaf size=17

$$\operatorname{Int}\left(\frac{\operatorname{sech}(a+bx)}{c+dx}, x\right)$$

[Out] Unintegrable(sech(b\*x+a)/(d\*x+c), x)

Rubi [A]

time = 0.02, antiderivative size = 0, normalized size of antiderivative = 0.00, number of steps used = 0, number of rules used = 0, integrand size = 0,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.000$ , Rules used = {}

$$\int \frac{\operatorname{sech}(a+bx)}{c+dx} dx$$

Verification is not applicable to the result.

[In] Int[Sech[a + b\*x]/(c + d\*x), x]

[Out] Defer[Int][Sech[a + b\*x]/(c + d\*x), x]

Rubi steps

$$\int \frac{\operatorname{sech}(a+bx)}{c+dx} dx = \int \frac{\operatorname{sech}(a+bx)}{c+dx} dx$$

Mathematica [A]

time = 0.24, size = 0, normalized size = 0.00

$$\int \frac{\operatorname{sech}(a+bx)}{c+dx} dx$$

Verification is not applicable to the result.

[In] Integrate[Sech[a + b\*x]/(c + d\*x), x]

[Out] Integrate[Sech[a + b\*x]/(c + d\*x), x]

Maple [A]

time = 180.00, size = 0, normalized size = 0.00

$$\int \frac{\operatorname{sech}(bx+a)}{dx+c} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `int(sech(b*x+a)/(d*x+c),x)`

[Out] `int(sech(b*x+a)/(d*x+c),x)`

**Maxima** [A]

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(sech(b*x+a)/(d*x+c),x, algorithm="maxima")`

[Out] `integrate(sech(b*x + a)/(d*x + c), x)`

**Fricas** [A]

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(sech(b*x+a)/(d*x+c),x, algorithm="fricas")`

[Out] `integral(sech(b*x + a)/(d*x + c), x)`

**Sympy** [A]

time = 0.00, size = 0, normalized size = 0.00

$$\int \frac{\operatorname{sech}(a + bx)}{c + dx} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(sech(b*x+a)/(d*x+c),x)`

[Out] `Integral(sech(a + b*x)/(c + d*x), x)`

**Giac** [A]

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(sech(b*x+a)/(d*x+c),x, algorithm="giac")`

[Out] `integrate(sech(b*x + a)/(d*x + c), x)`

**Mupad** [A]

time = 0.00, size = -1, normalized size = -0.06

$$\int \frac{1}{\cosh(a + bx)(c + dx)} dx$$

Verification of antiderivative is not currently implemented for this CAS.

```
[In] int(1/(cosh(a + b*x)*(c + d*x)),x)
```

```
[Out] int(1/(cosh(a + b*x)*(c + d*x)), x)
```

### 3.5 $\int (c + dx)^3 \operatorname{sech}^2(a + bx) dx$

**Optimal.** Leaf size=103

$$\frac{(c + dx)^3}{b} - \frac{3d(c + dx)^2 \log(1 + e^{2(a+bx)})}{b^2} - \frac{3d^2(c + dx) \operatorname{PolyLog}(2, -e^{2(a+bx)})}{b^3} + \frac{3d^3 \operatorname{PolyLog}(3, -e^{2(a+bx)})}{2b^4} + \frac{(c + dx)^3 \tanh(a + bx)}{b}$$

[Out]  $(d*x+c)^3/b-3*d*(d*x+c)^2*\ln(1+\exp(2*b*x+2*a))/b^2-3*d^2*(d*x+c)*\operatorname{polylog}(2, -\exp(2*b*x+2*a))/b^3+3/2*d^3*\operatorname{polylog}(3, -\exp(2*b*x+2*a))/b^4+(d*x+c)^3*\tanh(b*x+a)/b$

**Rubi [A]**

time = 0.15, antiderivative size = 103, normalized size of antiderivative = 1.00, number of steps used = 6, number of rules used = 6, integrand size = 16,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.375$ ,

Rules used = {4269, 3799, 2221, 2611, 2320, 6724}

$$\frac{3d^3 \operatorname{Li}_3(-e^{2(a+bx)})}{2b^4} - \frac{3d^2(c + dx) \operatorname{Li}_2(-e^{2(a+bx)})}{b^3} - \frac{3d(c + dx)^2 \log(e^{2(a+bx)} + 1)}{b^2} + \frac{(c + dx)^3 \tanh(a + bx)}{b} + \frac{(c + dx)^3}{b}$$

Antiderivative was successfully verified.

[In]  $\operatorname{Int}[(c + d*x)^3*\operatorname{Sech}[a + b*x]^2, x]$

[Out]  $(c + d*x)^3/b - (3*d*(c + d*x)^2*\operatorname{Log}[1 + E^(2*(a + b*x))])/b^2 - (3*d^2*(c + d*x)*\operatorname{PolyLog}[2, -E^(2*(a + b*x))])/b^3 + (3*d^3*\operatorname{PolyLog}[3, -E^(2*(a + b*x))])/b^4 + ((c + d*x)^3*\operatorname{Tanh}[a + b*x])/b$

Rule 2221

$\operatorname{Int}[(((F_)^((g_)*(e_) + (f_)*(x_)))^{(n_)*((c_) + (d_)*(x_))^{(m_))}/((a_) + (b_)*((F_)^((g_)*(e_) + (f_)*(x_)))^{(n_)}), x\_Symbol] := \operatorname{Simp}[(c + d*x)^m/(b*f*g*n*\operatorname{Log}[F])]*\operatorname{Log}[1 + b*((F^(g*(e + f*x)))^n/a)], x] - \operatorname{Dist}[d*(m/(b*f*g*n*\operatorname{Log}[F])), \operatorname{Int}[(c + d*x)^{(m-1)}*\operatorname{Log}[1 + b*((F^(g*(e + f*x)))^n/a)], x], x] /; \operatorname{FreeQ}\{F, a, b, c, d, e, f, g, n\}, x\} \&\& \operatorname{IGtQ}[m, 0]$

Rule 2320

$\operatorname{Int}[u_, x\_Symbol] := \operatorname{With}\{v = \operatorname{FunctionOfExponential}[u, x]\}, \operatorname{Dist}[v/D[v, x], \operatorname{Subst}[\operatorname{Int}[\operatorname{FunctionOfExponentialFunction}[u, x]/x, x], x, v], x] /; \operatorname{FunctionOfExponentialQ}[u, x] \&\& !\operatorname{MatchQ}[u, (w_)*((a_)*(v_)^{(n_)})^{(m_)} /; \operatorname{FreeQ}\{a, m, n\}, x] \&\& \operatorname{IntegerQ}[m*n] \&\& !\operatorname{MatchQ}[u, E^((c_)*((a_) + (b_)*x))* (F_)[v_] /; \operatorname{FreeQ}\{a, b, c\}, x] \&\& \operatorname{InverseFunctionQ}[F[x]]]$

Rule 2611

$\operatorname{Int}[\operatorname{Log}[1 + (e_)*((F_)^((c_)*((a_) + (b_)*(x_)))^{(n_)})*((f_) + (g_)*(x_))^{(m_)}], x\_Symbol] := \operatorname{Simp}[(-f + g*x)^m*(\operatorname{PolyLog}[2, (-e)*(F^(c*(a + b*x)))^n]/(b*c*n*\operatorname{Log}[F])), x] + \operatorname{Dist}[g*(m/(b*c*n*\operatorname{Log}[F])), \operatorname{Int}[(f + g*x)^m$

- 1)\*PolyLog[2, (-e)\*(F^(c\*(a + b\*x)))^n], x], x] /; FreeQ[{F, a, b, c, e, f, g, n}, x] && GtQ[m, 0]

### Rule 3799

Int[((c\_.) + (d\_.)\*(x\_))^(m\_.)\*tan[(e\_.) + (Complex[0, fz\_])\*(f\_.)\*(x\_)], x\_Symbol] := Simp[(-I)\*((c + d\*x)^(m + 1)/(d\*(m + 1))), x] + Dist[2\*I, Int[(c + d\*x)^m\*(E^(2\*((-I)\*e + f\*fz\*x)))/(1 + E^(2\*((-I)\*e + f\*fz\*x)))], x], x] /; FreeQ[{c, d, e, f, fz}, x] && IGtQ[m, 0]

### Rule 4269

Int[csc[(e\_.) + (f\_.)\*(x\_)]^2\*((c\_.) + (d\_.)\*(x\_))^(m\_.), x\_Symbol] := Simp[(-(c + d\*x)^m)\*(Cot[e + f\*x]/f), x] + Dist[d\*(m/f), Int[(c + d\*x)^(m - 1)\*Cot[e + f\*x], x], x] /; FreeQ[{c, d, e, f}, x] && GtQ[m, 0]

### Rule 6724

Int[PolyLog[n\_, (c\_.)\*((a\_.) + (b\_.)\*(x\_))^(p\_.)]/((d\_.) + (e\_.)\*(x\_)), x\_Symbol] := Simp[PolyLog[n + 1, c\*(a + b\*x)^p]/(e\*p), x] /; FreeQ[{a, b, c, d, e, n, p}, x] && EqQ[b\*d, a\*e]

### Rubi steps

$$\begin{aligned}
 \int (c + dx)^3 \operatorname{sech}^2(a + bx) dx &= \frac{(c + dx)^3 \tanh(a + bx)}{b} - \frac{(3d) \int (c + dx)^2 \tanh(a + bx) dx}{b} \\
 &= \frac{(c + dx)^3}{b} + \frac{(c + dx)^3 \tanh(a + bx)}{b} - \frac{(6d) \int \frac{e^{2(a+bx)}(c+dx)^2}{1+e^{2(a+bx)}} dx}{b} \\
 &= \frac{(c + dx)^3}{b} - \frac{3d(c + dx)^2 \log(1 + e^{2(a+bx)})}{b^2} + \frac{(c + dx)^3 \tanh(a + bx)}{b} + \frac{(6d^2)}{b^3} \\
 &= \frac{(c + dx)^3}{b} - \frac{3d(c + dx)^2 \log(1 + e^{2(a+bx)})}{b^2} - \frac{3d^2(c + dx) \operatorname{Li}_2(-e^{2(a+bx)})}{b^3} + \frac{(6d^2)}{b^3} \\
 &= \frac{(c + dx)^3}{b} - \frac{3d(c + dx)^2 \log(1 + e^{2(a+bx)})}{b^2} - \frac{3d^2(c + dx) \operatorname{Li}_2(-e^{2(a+bx)})}{b^3} + \frac{(6d^2)}{b^3} \\
 &= \frac{(c + dx)^3}{b} - \frac{3d(c + dx)^2 \log(1 + e^{2(a+bx)})}{b^2} - \frac{3d^2(c + dx) \operatorname{Li}_2(-e^{2(a+bx)})}{b^3} + \frac{(6d^2)}{b^3}
 \end{aligned}$$

### Mathematica [A]

time = 0.39, size = 135, normalized size = 1.31

$$\frac{d \left( \frac{4b^3 e^{2ax} (3c^2 + 3cdx + d^2 x^2)}{1 + e^{2a}} - 6b^2 (c + dx)^2 \log(1 + e^{2(a+bx)}) - 6bd(c + dx) \operatorname{PolyLog}(2, -e^{2(a+bx)}) + 3d^2 \operatorname{PolyLog}(3, -e^{2(a+bx)}) \right)}{2b^4} + \frac{(c + dx)^3 \operatorname{sech}(a) \operatorname{sech}(a + bx) \sinh(bx)}{b}$$

Antiderivative was successfully verified.

[In] Integrate[(c + d\*x)^3\*Sech[a + b\*x]^2,x]

[Out] (d\*((4\*b^3\*E^(2\*a)\*x\*(3\*c^2 + 3\*c\*d\*x + d^2\*x^2))/(1 + E^(2\*a)) - 6\*b^2\*(c + d\*x)^2\*Log[1 + E^(2\*(a + b\*x))] - 6\*b\*d\*(c + d\*x)\*PolyLog[2, -E^(2\*(a + b\*x))] + 3\*d^2\*PolyLog[3, -E^(2\*(a + b\*x))]))/(2\*b^4) + ((c + d\*x)^3\*Sech[a]\*Sech[a + b\*x]\*Sinh[b\*x])/b

**Maple [B]** Leaf count of result is larger than twice the leaf count of optimal. 297 vs. 2(101) = 202.

time = 0.89, size = 298, normalized size = 2.89

method	result
risch	$-\frac{2(d^3x^3+3cd^2x^2+3c^2dx+c^3)}{(e^{2bx+2a}+1)b} + \frac{6dc^2 \ln(e^{bx+a})}{b^2} - \frac{3dc^2 \ln(e^{2bx+2a}+1)}{b^2} + \frac{6d^3a^2 \ln(e^{bx+a})}{b^4} + \frac{2d^3x^3}{b} - \frac{6d^3a^2x}{b^3} - \frac{4d^3a^3}{b^4}$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((d\*x+c)^3\*sech(b\*x+a)^2,x,method=\_RETURNVERBOSE)

[Out] -2\*(d^3\*x^3+3\*c\*d^2\*x^2+3\*c^2\*d\*x+c^3)/(exp(2\*b\*x+2\*a)+1)/b+6/b^2\*d\*c^2\*ln(exp(b\*x+a))-3/b^2\*d\*c^2\*ln(exp(2\*b\*x+2\*a)+1)+6/b^4\*d^3\*a^2\*ln(exp(b\*x+a))+2/b\*d^3\*x^3-6/b^3\*d^3\*a^2\*x-4/b^4\*d^3\*a^3-3/b^2\*d^3\*ln(exp(2\*b\*x+2\*a)+1)\*x^2-3/b^3\*d^3\*polylog(2,-exp(2\*b\*x+2\*a))\*x+3/2\*d^3\*polylog(3,-exp(2\*b\*x+2\*a))/b^4-12/b^3\*d^2\*a\*c\*ln(exp(b\*x+a))+6/b\*d^2\*c\*x^2+12/b^2\*d^2\*a\*c\*x+6/b^3\*d^2\*c\*a^2-6/b^2\*d^2\*c\*ln(exp(2\*b\*x+2\*a)+1)\*x-3/b^3\*d^2\*c\*polylog(2,-exp(2\*b\*x+2\*a))

**Maxima [B]** Leaf count of result is larger than twice the leaf count of optimal. 238 vs. 2(100) = 200.

time = 0.39, size = 238, normalized size = 2.31

$$3c^2d\left(\frac{2xe^{(2bx+2a)}}{be^{(2bx+2a)}+b} - \frac{\log((e^{(2bx+2a)}+1)e^{-2a})}{b^2}\right) - \frac{3(2bx \log(e^{(2bx+2a)}+1) + \text{Li}_2(-e^{(2bx+2a)}))cd^2}{b^3} + \frac{2c^2}{b(e^{-(2bx-2a)}+1)} - \frac{2(d^3x^3+3cd^2x^2)}{be^{(2bx+2a)}+b} - \frac{3(2b^2x^2 \log(e^{(2bx+2a)}+1) + 2bx \text{Li}_2(-e^{(2bx+2a)}) - \text{Li}_2(-e^{(2bx-2a)}))d^2}{2b^4} + \frac{2(b^3d^3x^3+3b^2cd^2x^2)}{b^4}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)^3\*sech(b\*x+a)^2,x, algorithm="maxima")

[Out] 3\*c^2\*d\*(2\*x\*e^(2\*b\*x + 2\*a)/(b\*e^(2\*b\*x + 2\*a) + b) - log((e^(2\*b\*x + 2\*a) + 1)\*e^(-2\*a)))/b^2 - 3\*(2\*b\*x\*log(e^(2\*b\*x + 2\*a) + 1) + dilog(-e^(2\*b\*x + 2\*a)))\*c\*d^2/b^3 + 2\*c^3/(b\*(e^(-2\*b\*x - 2\*a) + 1)) - 2\*(d^3\*x^3 + 3\*c\*d^2\*x^2)/(b\*e^(2\*b\*x + 2\*a) + b) - 3/2\*(2\*b^2\*x^2\*log(e^(2\*b\*x + 2\*a) + 1) + 2\*b\*x\*dilog(-e^(2\*b\*x + 2\*a)) - polylog(3, -e^(2\*b\*x + 2\*a)))\*d^3/b^4 + 2\*(b^3\*d^3\*x^3 + 3\*b^3\*c\*d^2\*x^2)/b^4

**Fricas [C]** Result contains complex when optimal does not.

time = 0.43, size = 1332, normalized size = 12.93

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)^3\*sech(b\*x+a)^2,x, algorithm="fricas")

[Out]  $-(2*b^3*c^3 - 6*a*b^2*c^2*d + 6*a^2*b*c*d^2 - 2*a^3*d^3 - 2*(b^3*d^3*x^3 + 3*b^3*c*d^2*x^2 + 3*b^3*c^2*d*x + 3*a*b^2*c^2*d - 3*a^2*b*c*d^2 + a^3*d^3)*\cosh(b*x + a)^2 - 4*(b^3*d^3*x^3 + 3*b^3*c*d^2*x^2 + 3*b^3*c^2*d*x + 3*a*b^2*c^2*d - 3*a^2*b*c*d^2 + a^3*d^3)*\cosh(b*x + a)*\sinh(b*x + a) - 2*(b^3*d^3*x^3 + 3*b^3*c*d^2*x^2 + 3*b^3*c^2*d*x + 3*a*b^2*c^2*d - 3*a^2*b*c*d^2 + a^3*d^3)*\sinh(b*x + a)^2 + 6*(b*d^3*x + b*c*d^2 + (b*d^3*x + b*c*d^2)*\cosh(b*x + a)^2 + 2*(b*d^3*x + b*c*d^2)*\cosh(b*x + a)*\sinh(b*x + a) + (b*d^3*x + b*c*d^2)*\sinh(b*x + a)^2)*\operatorname{dilog}(I*\cosh(b*x + a) + I*\sinh(b*x + a)) + 6*(b*d^3*x + b*c*d^2 + (b*d^3*x + b*c*d^2)*\cosh(b*x + a)^2 + 2*(b*d^3*x + b*c*d^2)*\cosh(b*x + a)*\sinh(b*x + a) + (b*d^3*x + b*c*d^2)*\sinh(b*x + a)^2)*\operatorname{dilog}(-I*\cosh(b*x + a) - I*\sinh(b*x + a)) + 3*(b^2*c^2*d - 2*a*b*c*d^2 + a^2*d^3 + (b^2*c^2*d - 2*a*b*c*d^2 + a^2*d^3)*\cosh(b*x + a)^2 + 2*(b^2*c^2*d - 2*a*b*c*d^2 + a^2*d^3)*\cosh(b*x + a)*\sinh(b*x + a) + (b^2*c^2*d - 2*a*b*c*d^2 + a^2*d^3)*\sinh(b*x + a)^2)*\log(\cosh(b*x + a) + \sinh(b*x + a) + I) + 3*(b^2*c^2*d - 2*a*b*c*d^2 + a^2*d^3 + (b^2*c^2*d - 2*a*b*c*d^2 + a^2*d^3)*\cosh(b*x + a)^2 + 2*(b^2*c^2*d - 2*a*b*c*d^2 + a^2*d^3)*\cosh(b*x + a)*\sinh(b*x + a) + (b^2*c^2*d - 2*a*b*c*d^2 + a^2*d^3)*\sinh(b*x + a)^2)*\log(\cosh(b*x + a) + \sinh(b*x + a) - I) + 3*(b^2*d^3*x^2 + 2*b^2*c*d^2*x + 2*a*b*c*d^2 - a^2*d^3 + (b^2*d^3*x^2 + 2*b^2*c*d^2*x + 2*a*b*c*d^2 - a^2*d^3)*\cosh(b*x + a)^2 + 2*(b^2*d^3*x^2 + 2*b^2*c*d^2*x + 2*a*b*c*d^2 - a^2*d^3)*\cosh(b*x + a)*\sinh(b*x + a) + (b^2*d^3*x^2 + 2*b^2*c*d^2*x + 2*a*b*c*d^2 - a^2*d^3)*\sinh(b*x + a)^2)*\log(I*\cosh(b*x + a) + I*\sinh(b*x + a) + 1) + 3*(b^2*d^3*x^2 + 2*b^2*c*d^2*x + 2*a*b*c*d^2 - a^2*d^3 + (b^2*d^3*x^2 + 2*b^2*c*d^2*x + 2*a*b*c*d^2 - a^2*d^3)*\cosh(b*x + a)^2 + 2*(b^2*d^3*x^2 + 2*b^2*c*d^2*x + 2*a*b*c*d^2 - a^2*d^3)*\cosh(b*x + a)*\sinh(b*x + a) + (b^2*d^3*x^2 + 2*b^2*c*d^2*x + 2*a*b*c*d^2 - a^2*d^3)*\sinh(b*x + a)^2)*\log(-I*\cosh(b*x + a) - I*\sinh(b*x + a) + 1) - 6*(d^3*\cosh(b*x + a)^2 + 2*d^3*\cosh(b*x + a)*\sinh(b*x + a) + d^3*\sinh(b*x + a)^2 + d^3)*\operatorname{polylog}(3, I*\cosh(b*x + a) + I*\sinh(b*x + a)) - 6*(d^3*\cosh(b*x + a)^2 + 2*d^3*\cosh(b*x + a)*\sinh(b*x + a) + d^3*\sinh(b*x + a)^2 + d^3)*\operatorname{polylog}(3, -I*\cosh(b*x + a) - I*\sinh(b*x + a))/(b^4*\cosh(b*x + a)^2 + 2*b^4*\cosh(b*x + a)*\sinh(b*x + a) + b^4*\sinh(b*x + a)^2 + b^4)$

Sympy [F]

time = 0.00, size = 0, normalized size = 0.00

$$\int (c + dx)^3 \operatorname{sech}^2(a + bx) dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*\*3\*sech(b\*x+a)\*\*2,x)

[Out] Integral((c + d\*x)\*\*3\*sech(a + b\*x)\*\*2, x)

**Giac [F]**

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)^3\*sech(b\*x+a)^2,x, algorithm="giac")

[Out] integrate((d\*x + c)^3\*sech(b\*x + a)^2, x)

**Mupad [F]**

time = 0.00, size = -1, normalized size = -0.01

$$\int \frac{(c + dx)^3}{\cosh(a + bx)^2} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((c + d\*x)^3/cosh(a + b\*x)^2,x)

[Out] int((c + d\*x)^3/cosh(a + b\*x)^2, x)



### 3.6 $\int (c + dx)^2 \operatorname{sech}^2(a + bx) dx$

**Optimal.** Leaf size=73

$$\frac{(c + dx)^2}{b} - \frac{2d(c + dx) \log(1 + e^{2(a+bx)})}{b^2} - \frac{d^2 \operatorname{PolyLog}(2, -e^{2(a+bx)})}{b^3} + \frac{(c + dx)^2 \tanh(a + bx)}{b}$$

[Out]  $(d*x+c)^2/b-2*d*(d*x+c)*\ln(1+\exp(2*b*x+2*a))/b^2-d^2*\operatorname{polylog}(2,-\exp(2*b*x+2*a))/b^3+(d*x+c)^2*\tanh(b*x+a)/b$

**Rubi [A]**

time = 0.10, antiderivative size = 73, normalized size of antiderivative = 1.00, number of steps used = 5, number of rules used = 5, integrand size = 16,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.312$ , Rules used = {4269, 3799, 2221, 2317, 2438}

$$-\frac{d^2 \operatorname{Li}_2(-e^{2(a+bx)})}{b^3} - \frac{2d(c + dx) \log(e^{2(a+bx)} + 1)}{b^2} + \frac{(c + dx)^2 \tanh(a + bx)}{b} + \frac{(c + dx)^2}{b}$$

Antiderivative was successfully verified.

[In] `Int[(c + d*x)^2*Sech[a + b*x]^2,x]`

[Out]  $(c + d*x)^2/b - (2*d*(c + d*x)*\operatorname{Log}[1 + E^{2*(a + b*x)}])/b^2 - (d^2*\operatorname{PolyLog}[2, -E^{2*(a + b*x)}])/b^3 + ((c + d*x)^2*\operatorname{Tanh}[a + b*x])/b$

Rule 2221

```
Int[(((F_)^((g_)*((e_) + (f_)*(x_))))^(n_))*((c_) + (d_)*(x_))^(m_)]/
((a_) + (b_)*((F_)^((g_)*((e_) + (f_)*(x_))))^(n_)), x_Symbol] :> Simp
[((c + d*x)^m/(b*f*g*n*Log[F]))*Log[1 + b*((F^(g*(e + f*x)))^n/a)], x] - Di
st[d*(m/(b*f*g*n*Log[F])), Int[(c + d*x)^(m - 1)*Log[1 + b*((F^(g*(e + f*x)
))^n/a)], x], x] /; FreeQ[{F, a, b, c, d, e, f, g, n}, x] && IGtQ[m, 0]
```

Rule 2317

```
Int[Log[(a_) + (b_)*((F_)^((e_)*((c_) + (d_)*(x_))))^(n_)], x_Symbol]
:> Dist[1/(d*e*n*Log[F]), Subst[Int[Log[a + b*x]/x, x], x, (F^(e*(c + d*x)
))^n], x] /; FreeQ[{F, a, b, c, d, e, n}, x] && GtQ[a, 0]
```

Rule 2438

```
Int[Log[(c_)*((d_) + (e_)*(x_))^(n_)]/(x_), x_Symbol] :> Simp[-PolyLog[2
, (-c)*e*x^n]/n, x] /; FreeQ[{c, d, e, n}, x] && EqQ[c*d, 1]
```

Rule 3799

```
Int[(((c_) + (d_)*(x_))^(m_))*tan[(e_) + (Complex[0, fz_])*(f_)*(x_)], x
_Symbol] :> Simp[(-I)*((c + d*x)^(m + 1)/(d*(m + 1))), x] + Dist[2*I, Int[(
```

```
c + d*x)^m*(E^(2*((-I)*e + f*fz*x))/(1 + E^(2*((-I)*e + f*fz*x))))), x], x]
/; FreeQ[{c, d, e, f, fz}, x] && IGtQ[m, 0]
```

### Rule 4269

```
Int[csc[(e_.) + (f_.)*(x_)]^2*((c_.) + (d_.)*(x_))^(m_.), x_Symbol] := Simp
[(-(c + d*x)^m)*(Cot[e + f*x]/f), x] + Dist[d*(m/f), Int[(c + d*x)^(m - 1)*
Cot[e + f*x], x], x] /; FreeQ[{c, d, e, f}, x] && GtQ[m, 0]
```

### Rubi steps

$$\begin{aligned} \int (c + dx)^2 \operatorname{sech}^2(a + bx) dx &= \frac{(c + dx)^2 \tanh(a + bx)}{b} - \frac{(2d) \int (c + dx) \tanh(a + bx) dx}{b} \\ &= \frac{(c + dx)^2}{b} + \frac{(c + dx)^2 \tanh(a + bx)}{b} - \frac{(4d) \int \frac{e^{2(a+bx)}(c+dx)}{1+e^{2(a+bx)}} dx}{b} \\ &= \frac{(c + dx)^2}{b} - \frac{2d(c + dx) \log(1 + e^{2(a+bx)})}{b^2} + \frac{(c + dx)^2 \tanh(a + bx)}{b} + \frac{(2d^2) \int}{b} \\ &= \frac{(c + dx)^2}{b} - \frac{2d(c + dx) \log(1 + e^{2(a+bx)})}{b^2} + \frac{(c + dx)^2 \tanh(a + bx)}{b} + \frac{d^2 \operatorname{Subst}}{b} \\ &= \frac{(c + dx)^2}{b} - \frac{2d(c + dx) \log(1 + e^{2(a+bx)})}{b^2} - \frac{d^2 \operatorname{Li}_2(-e^{2(a+bx)})}{b^3} + \frac{(c + dx)^2 \tanh(a + bx)}{b} \end{aligned}$$

**Mathematica [C]** Result contains complex when optimal does not.

time = 4.93, size = 186, normalized size = 2.55

$\operatorname{sech}(a) (2b \operatorname{erf}(\operatorname{coth}(a) \log(\cosh(a + bx)) + bx \operatorname{sinh}(a)) + d^2 (\operatorname{coth}(a) (r \log(1 + e^{2a}) - 2bx \log(1 - e^{2b \operatorname{tanh}^{-1}(\operatorname{coth}(a))}) - r(\log(\cosh(bx)) - 2 \operatorname{tanh}^{-1}(\operatorname{coth}(a)) (bx + \log(1 - e^{2b \operatorname{tanh}^{-1}(\operatorname{coth}(a))}) - \log(1 \operatorname{sinh}(bx + \operatorname{tanh}^{-1}(\operatorname{coth}(a)))))) + \operatorname{PolyLog}(2, e^{2b \operatorname{tanh}^{-1}(\operatorname{coth}(a))}) + \psi^2 e^{-2 \operatorname{tanh}^{-1}(\operatorname{coth}(a))} \sqrt{-\operatorname{coth}^2(a)} \operatorname{sinh}(a)) + \psi^2 (c + dx) \operatorname{sech}(a + bx) \operatorname{sinh}(bx))$

Antiderivative was successfully verified.

```
[In] Integrate[(c + d*x)^2*Sech[a + b*x]^2,x]
```

```
[Out] (Sech[a]*(2*b*c*d*(-Cosh[a]*Log[Cosh[a + b*x]]) + b*x*Sinh[a]) + d^2*(Cosh
[a]*(I*Pi*Log[1 + E^(2*b*x)] - 2*b*x*Log[1 - E^(-2*(b*x + ArcTanh[Coth[a]])
]) - I*Pi*(b*x + Log[Cosh[b*x]]) - 2*ArcTanh[Coth[a]]*(b*x + Log[1 - E^(-2*
(b*x + ArcTanh[Coth[a]])])) - Log[I*Sinh[b*x + ArcTanh[Coth[a]]])) + PolyLog
[2, E^(-2*(b*x + ArcTanh[Coth[a]])])) + (b^2*x^2*sqrt[-Csch[a]^2]*Sinh[a])/
E^ArcTanh[Coth[a]] + b^2*(c + d*x)^2*Sech[a + b*x]*Sinh[b*x])/b^3
```

**Maple [B]** Leaf count of result is larger than twice the leaf count of optimal. 158 vs. 2(73) = 146.

time = 0.89, size = 159, normalized size = 2.18

method	result
risch	$-\frac{2(d^2x^2+2cdx+c^2)}{(e^{2bx+2a}+1)b} - \frac{2dc \ln(e^{2bx+2a}+1)}{b^2} + \frac{4dc \ln(e^{bx+a})}{b^2} + \frac{2d^2x^2}{b} + \frac{4d^2ax}{b^2} + \frac{2d^2a^2}{b^3} - \frac{2d^2 \ln(e^{2bx+2a}+1)x}{b^2} - \frac{d^2 \text{poly}}{b^2}$

Verification of antiderivative is not currently implemented for this CAS.

[In] `int((d*x+c)^2*sech(b*x+a)^2,x,method=_RETURNVERBOSE)`

[Out] 
$$-2*(d^2*x^2+2*c*d*x+c^2)/(exp(2*b*x+2*a)+1)/b-2/b^2*d*c*\ln(exp(2*b*x+2*a)+1)+4/b^2*d*c*\ln(exp(b*x+a))+2/b*d^2*x^2+4/b^2*d^2*a*x+2/b^3*d^2*a^2-2/b^2*d^2*\ln(exp(2*b*x+2*a)+1)*x-d^2*\text{polylog}(2,-exp(2*b*x+2*a))/b^3-4/b^3*d^2*a*\ln(exp(b*x+a))$$

**Maxima** [F]

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((d*x+c)^2*sech(b*x+a)^2,x, algorithm="maxima")`

[Out] 
$$-2*d^2*(x^2/(b*e^{(2*b*x + 2*a)} + b) - 2*\integrate(x/(b*e^{(2*b*x + 2*a)} + b), x)) + 2*c*d*(2*x*e^{(2*b*x + 2*a)}/(b*e^{(2*b*x + 2*a)} + b) - \log((e^{(2*b*x + 2*a)} + 1)*e^{(-2*a)})/b^2) + 2*c^2/(b*(e^{(-2*b*x - 2*a)} + 1))$$

**Fricas** [C] Result contains complex when optimal does not.

time = 0.37, size = 715, normalized size = 9.79

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((d*x+c)^2*sech(b*x+a)^2,x, algorithm="fricas")`

[Out] 
$$-2*(b^2*c^2 - 2*a*b*c*d + a^2*d^2 - (b^2*d^2*x^2 + 2*b^2*c*d*x + 2*a*b*c*d - a^2*d^2)*\cosh(b*x + a)^2 - 2*(b^2*d^2*x^2 + 2*b^2*c*d*x + 2*a*b*c*d - a^2*d^2)*\cosh(b*x + a)*\sinh(b*x + a) - (b^2*d^2*x^2 + 2*b^2*c*d*x + 2*a*b*c*d - a^2*d^2)*\sinh(b*x + a)^2 + (d^2*\cosh(b*x + a)^2 + 2*d^2*\cosh(b*x + a)*\sinh(b*x + a) + d^2*\sinh(b*x + a)^2 + d^2)*\text{dilog}(I*\cosh(b*x + a) + I*\sinh(b*x + a)) + (d^2*\cosh(b*x + a)^2 + 2*d^2*\cosh(b*x + a)*\sinh(b*x + a) + d^2*\sinh(b*x + a)^2 + d^2)*\text{dilog}(-I*\cosh(b*x + a) - I*\sinh(b*x + a)) + (b*c*d - a*d^2 + (b*c*d - a*d^2)*\cosh(b*x + a)^2 + 2*(b*c*d - a*d^2)*\cosh(b*x + a)*\sinh(b*x + a) + (b*c*d - a*d^2)*\sinh(b*x + a)^2)*\log(\cosh(b*x + a) + \sinh(b*x + a) + I) + (b*c*d - a*d^2 + (b*c*d - a*d^2)*\cosh(b*x + a)^2 + 2*(b*c*d - a*d^2)*\cosh(b*x + a)*\sinh(b*x + a) + (b*c*d - a*d^2)*\sinh(b*x + a)^2)*\log(\cosh(b*x + a) + \sinh(b*x + a) - I) + (b*d^2*x + a*d^2 + (b*d^2*x + a*d^2)*\cosh$$

$$(b*x + a)^2 + 2*(b*d^2*x + a*d^2)*\cosh(b*x + a)*\sinh(b*x + a) + (b*d^2*x + a*d^2)*\sinh(b*x + a)^2*\log(I*\cosh(b*x + a) + I*\sinh(b*x + a) + 1) + (b*d^2*x + a*d^2 + (b*d^2*x + a*d^2)*\cosh(b*x + a)^2 + 2*(b*d^2*x + a*d^2)*\cosh(b*x + a)*\sinh(b*x + a) + (b*d^2*x + a*d^2)*\sinh(b*x + a)^2*\log(-I*\cosh(b*x + a) - I*\sinh(b*x + a) + 1))/(b^3*\cosh(b*x + a)^2 + 2*b^3*\cosh(b*x + a)*\sinh(b*x + a) + b^3*\sinh(b*x + a)^2 + b^3)$$

**Sympy [F]**

time = 0.00, size = 0, normalized size = 0.00

$$\int (c + dx)^2 \operatorname{sech}^2(a + bx) dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*\*2\*sech(b\*x+a)\*\*2,x)

[Out] Integral((c + d\*x)\*\*2\*sech(a + b\*x)\*\*2, x)

**Giac [F]**

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)^2\*sech(b\*x+a)^2,x, algorithm="giac")

[Out] integrate((d\*x + c)^2\*sech(b\*x + a)^2, x)

**Mupad [F]**

time = 0.00, size = -1, normalized size = -0.01

$$\int \frac{(c + dx)^2}{\cosh(a + bx)^2} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((c + d\*x)^2/cosh(a + b\*x)^2,x)

[Out] int((c + d\*x)^2/cosh(a + b\*x)^2, x)

### 3.7 $\int (c + dx)\operatorname{sech}^2(a + bx) dx$

Optimal. Leaf size=29

$$-\frac{d \log(\cosh(a + bx))}{b^2} + \frac{(c + dx) \tanh(a + bx)}{b}$$

[Out]  $-d*\ln(\cosh(b*x+a))/b^2+(d*x+c)*\tanh(b*x+a)/b$

**Rubi** [A]

time = 0.02, antiderivative size = 29, normalized size of antiderivative = 1.00, number of steps used = 2, number of rules used = 2, integrand size = 14,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.143$ , Rules used = {4269, 3556}

$$\frac{(c + dx) \tanh(a + bx)}{b} - \frac{d \log(\cosh(a + bx))}{b^2}$$

Antiderivative was successfully verified.

[In]  $\text{Int}[(c + d*x)*\text{Sech}[a + b*x]^2, x]$

[Out]  $-((d*\text{Log}[\text{Cosh}[a + b*x]])/b^2) + ((c + d*x)*\text{Tanh}[a + b*x])/b$

Rule 3556

$\text{Int}[\tan[(c_.) + (d_.)*(x_.)], x\_Symbol] \rightarrow \text{Simp}[-\text{Log}[\text{RemoveContent}[\text{Cos}[c + d*x], x]]/d, x] /; \text{FreeQ}\{c, d\}, x]$

Rule 4269

$\text{Int}[\text{csc}[(e_.) + (f_.)*(x_.)]^2*((c_.) + (d_.)*(x_.))^{(m_.)}, x\_Symbol] \rightarrow \text{Simp}[(-(c + d*x)^m)*(\text{Cot}[e + f*x]/f), x] + \text{Dist}[d*(m/f), \text{Int}[(c + d*x)^{(m-1)}*\text{Cot}[e + f*x], x], x] /; \text{FreeQ}\{c, d, e, f\}, x] \&\& \text{GtQ}[m, 0]$

Rubi steps

$$\begin{aligned} \int (c + dx)\operatorname{sech}^2(a + bx) dx &= \frac{(c + dx) \tanh(a + bx)}{b} - \frac{d \int \tanh(a + bx) dx}{b} \\ &= -\frac{d \log(\cosh(a + bx))}{b^2} + \frac{(c + dx) \tanh(a + bx)}{b} \end{aligned}$$

**Mathematica** [A]

time = 0.08, size = 51, normalized size = 1.76

$$-\frac{d \log(\cosh(a + bx))}{b^2} + \frac{dx \operatorname{sech}(a) \operatorname{sech}(a + bx) \sinh(bx)}{b} + \frac{dx \tanh(a)}{b} + \frac{c \tanh(a + bx)}{b}$$

Antiderivative was successfully verified.

[In] Integrate[(c + d\*x)\*Sech[a + b\*x]^2,x]

[Out] -((d\*Log[Cosh[a + b\*x]])/b^2) + (d\*x\*Sech[a]\*Sech[a + b\*x]\*Sinh[b\*x])/b + (d\*x\*Tanh[a])/b + (c\*Tanh[a + b\*x])/b

**Maple [A]**

time = 0.82, size = 57, normalized size = 1.97

method	result	size
risch	$\frac{2dx}{b} + \frac{2da}{b^2} - \frac{2(dx+c)}{(e^{2bx+2a}+1)b} - \frac{d \ln(e^{2bx+2a}+1)}{b^2}$	57

Verification of antiderivative is not currently implemented for this CAS.

[In] int((d\*x+c)\*sech(b\*x+a)^2,x,method=\_RETURNVERBOSE)

[Out] 2/b\*d\*x+2/b^2\*d\*a-2\*(d\*x+c)/(exp(2\*b\*x+2\*a)+1)/b-d/b^2\*ln(exp(2\*b\*x+2\*a)+1)

**Maxima [B]** Leaf count of result is larger than twice the leaf count of optimal. 72 vs. 2(29) = 58.

time = 0.26, size = 72, normalized size = 2.48

$$d \left( \frac{2xe^{(2bx+2a)}}{be^{(2bx+2a)}+b} - \frac{\log((e^{(2bx+2a)}+1)e^{(-2a)})}{b^2} \right) + \frac{2c}{b(e^{(-2bx-2a)}+1)}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*sech(b\*x+a)^2,x, algorithm="maxima")

[Out] d\*(2\*x\*e^(2\*b\*x + 2\*a)/(b\*e^(2\*b\*x + 2\*a) + b) - log((e^(2\*b\*x + 2\*a) + 1)\*e^(-2\*a)))/b^2) + 2\*c/(b\*(e^(-2\*b\*x - 2\*a) + 1))

**Fricas [B]** Leaf count of result is larger than twice the leaf count of optimal. 161 vs. 2(29) = 58.

time = 0.37, size = 161, normalized size = 5.55

$$\frac{2bdx \cosh(bx+a)^2 + 4bdx \cosh(bx+a) \sinh(bx+a) + 2bdx \sinh(bx+a)^2 - 2bc - (d \cosh(bx+a)^2 + 2d \cosh(bx+a) \sinh(bx+a) + d \sinh(bx+a)^2 + d) \log\left(\frac{2 \cosh(bx+a)}{\cosh(bx+a) - \sinh(bx+a)}\right)}{b^2 \cosh(bx+a)^2 + 2b^2 \cosh(bx+a) \sinh(bx+a) + b^2 \sinh(bx+a)^2 + b^2}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*sech(b\*x+a)^2,x, algorithm="fricas")

[Out] (2\*b\*d\*x\*cosh(b\*x + a)^2 + 4\*b\*d\*x\*cosh(b\*x + a)\*sinh(b\*x + a) + 2\*b\*d\*x\*sinh(b\*x + a)^2 - 2\*b\*c - (d\*cosh(b\*x + a)^2 + 2\*d\*cosh(b\*x + a)\*sinh(b\*x + a) + d\*sinh(b\*x + a)^2 + d)\*log(2\*cosh(b\*x + a)/(cosh(b\*x + a) - sinh(b\*x + a))))/(b^2\*cosh(b\*x + a)^2 + 2\*b^2\*cosh(b\*x + a)\*sinh(b\*x + a) + b^2\*sinh(b\*x + a)^2 + b^2)

**Sympy [F]**

time = 0.00, size = 0, normalized size = 0.00

$$\int (c + dx) \operatorname{sech}^2(a + bx) dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*sech(b\*x+a)\*\*2,x)

[Out] Integral((c + d\*x)\*sech(a + b\*x)\*\*2, x)

**Giac [B]** Leaf count of result is larger than twice the leaf count of optimal. 78 vs. 2(29) = 58.  
time = 0.39, size = 78, normalized size = 2.69

$$\frac{2bdxe^{(2bx+2a)} - de^{(2bx+2a)} \log(e^{(2bx+2a)} + 1) - 2bc - d \log(e^{(2bx+2a)} + 1)}{b^2e^{(2bx+2a)} + b^2}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*sech(b\*x+a)^2,x, algorithm="giac")

[Out] (2\*b\*d\*x\*e^(2\*b\*x + 2\*a) - d\*e^(2\*b\*x + 2\*a)\*log(e^(2\*b\*x + 2\*a) + 1) - 2\*b\*c - d\*log(e^(2\*b\*x + 2\*a) + 1))/(b^2\*e^(2\*b\*x + 2\*a) + b^2)

**Mupad [B]**

time = 0.09, size = 50, normalized size = 1.72

$$\frac{2dx}{b} - \frac{2(c+dx)}{b(e^{2a+2bx}+1)} - \frac{d \ln(e^{2a}e^{2bx}+1)}{b^2}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((c + d\*x)/cosh(a + b\*x)^2,x)

[Out] (2\*d\*x)/b - (2\*(c + d\*x))/(b\*(exp(2\*a + 2\*b\*x) + 1)) - (d\*log(exp(2\*a)\*exp(2\*b\*x) + 1))/b^2

### 3.8 $\int \frac{\operatorname{sech}^2(a+bx)}{c+dx} dx$

**Optimal.** Leaf size=19

$$\operatorname{Int}\left(\frac{\operatorname{sech}^2(a+bx)}{c+dx}, x\right)$$

[Out] Unintegrable(sech(b\*x+a)^2/(d\*x+c), x)

**Rubi [A]**

time = 0.03, antiderivative size = 0, normalized size of antiderivative = 0.00, number of steps used = 0, number of rules used = 0, integrand size = 0,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.000$ , Rules used = {}

$$\int \frac{\operatorname{sech}^2(a+bx)}{c+dx} dx$$

Verification is not applicable to the result.

[In] Int[Sech[a + b\*x]^2/(c + d\*x), x]

[Out] Defer[Int][Sech[a + b\*x]^2/(c + d\*x), x]

Rubi steps

$$\int \frac{\operatorname{sech}^2(a+bx)}{c+dx} dx = \int \frac{\operatorname{sech}^2(a+bx)}{c+dx} dx$$

**Mathematica [A]**

time = 5.68, size = 0, normalized size = 0.00

$$\int \frac{\operatorname{sech}^2(a+bx)}{c+dx} dx$$

Verification is not applicable to the result.

[In] Integrate[Sech[a + b\*x]^2/(c + d\*x), x]

[Out] Integrate[Sech[a + b\*x]^2/(c + d\*x), x]

**Maple [A]**

time = 180.00, size = 0, normalized size = 0.00

$$\int \frac{\operatorname{sech}(bx+a)^2}{dx+c} dx$$

Verification of antiderivative is not currently implemented for this CAS.



[In] `int(sech(b*x+a)^2/(d*x+c),x)`

[Out] `int(sech(b*x+a)^2/(d*x+c),x)`

**Maxima** [A]

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(sech(b*x+a)^2/(d*x+c),x, algorithm="maxima")`

[Out] `-4*d*integrate(1/2/(b*d^2*x^2 + 2*b*c*d*x + b*c^2 + (b*d^2*x^2*e^(2*a) + 2*b*c*d*x*e^(2*a) + b*c^2*e^(2*a))*e^(2*b*x)), x) - 2/(b*d*x + b*c + (b*d*x*e^(2*a) + b*c*e^(2*a))*e^(2*b*x))`

**Fricas** [A]

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(sech(b*x+a)^2/(d*x+c),x, algorithm="fricas")`

[Out] `integral(sech(b*x + a)^2/(d*x + c), x)`

**Sympy** [A]

time = 0.00, size = 0, normalized size = 0.00

$$\int \frac{\operatorname{sech}^2(a + bx)}{c + dx} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(sech(b*x+a)**2/(d*x+c),x)`

[Out] `Integral(sech(a + b*x)**2/(c + d*x), x)`

**Giac** [A]

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(sech(b*x+a)^2/(d*x+c),x, algorithm="giac")`

[Out] `integrate(sech(b*x + a)^2/(d*x + c), x)`

**Mupad [A]**

time = 0.00, size = -1, normalized size = -0.05

$$\int \frac{1}{\cosh(a + bx)^2 (c + dx)} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int(1/(cosh(a + b\*x)^2\*(c + d\*x)),x)

[Out] int(1/(cosh(a + b\*x)^2\*(c + d\*x)), x)

### 3.9 $\int (c + dx)^3 \operatorname{sech}^3(a + bx) dx$

**Optimal.** Leaf size=296

$$-\frac{6d^2(c+dx)\operatorname{ArcTan}(e^{a+bx})}{b^3} + \frac{(c+dx)^3\operatorname{ArcTan}(e^{a+bx})}{b} + \frac{3id^3\operatorname{PolyLog}(2, -ie^{a+bx})}{b^4} - \frac{3id(c+dx)^2\operatorname{PolyLog}(2, -ie^{a+bx})}{2b^2}$$

```
[Out] -6*d^2*(d*x+c)*arctan(exp(b*x+a))/b^3+(d*x+c)^3*arctan(exp(b*x+a))/b+3*I*d^3*polylog(2,-I*exp(b*x+a))/b^4-3/2*I*d*(d*x+c)^2*polylog(2,-I*exp(b*x+a))/b^2-3*I*d^3*polylog(2,I*exp(b*x+a))/b^4+3/2*I*d*(d*x+c)^2*polylog(2,I*exp(b*x+a))/b^2+3*I*d^2*(d*x+c)*polylog(3,-I*exp(b*x+a))/b^3-3*I*d^2*(d*x+c)*polylog(3,I*exp(b*x+a))/b^3-3*I*d^3*polylog(4,-I*exp(b*x+a))/b^4+3*I*d^3*polylog(4,I*exp(b*x+a))/b^4+3/2*d*(d*x+c)^2*sech(b*x+a)/b^2+1/2*(d*x+c)^3*sech(b*x+a)*tanh(b*x+a)/b
```

**Rubi [A]**

time = 0.16, antiderivative size = 296, normalized size of antiderivative = 1.00, number of steps used = 15, number of rules used = 8, integrand size = 16,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.500$ , Rules used = {4271, 4265, 2317, 2438, 2611, 6744, 2320, 6724}

$$-\frac{6d^2(c+dx)\operatorname{ArcTan}(e^{a+bx})}{b^3} + \frac{(c+dx)^3\operatorname{ArcTan}(e^{a+bx})}{b} + \frac{3id^3\operatorname{PolyLog}(2, -ie^{a+bx})}{b^4} - \frac{3id(c+dx)^2\operatorname{PolyLog}(2, -ie^{a+bx})}{2b^2} + \frac{3id(c+dx)^2\operatorname{PolyLog}(2, Ie^{a+bx})}{2b^2} + \frac{3d(c+dx)^2\operatorname{sech}(a+bx)}{2b^2} + \frac{(c+dx)^3\operatorname{tanh}(a+bx)\operatorname{sech}(a+bx)}{2b}$$

Antiderivative was successfully verified.

```
[In] Int[(c + d*x)^3*Sech[a + b*x]^3,x]
```

```
[Out] (-6*d^2*(c + d*x)*ArcTan[E^(a + b*x)])/b^3 + ((c + d*x)^3*ArcTan[E^(a + b*x)])/b + (((3*I)*d^3*PolyLog[2, (-I)*E^(a + b*x)])/b^4 - (((3*I)/2)*d*(c + d*x)^2*PolyLog[2, (-I)*E^(a + b*x)])/b^2 - (((3*I)*d^3*PolyLog[2, I*E^(a + b*x)])/b^4 + (((3*I)/2)*d*(c + d*x)^2*PolyLog[2, I*E^(a + b*x)])/b^2 + ((3*I)*d^2*(c + d*x)*PolyLog[3, (-I)*E^(a + b*x)])/b^3 - ((3*I)*d^2*(c + d*x)*PolyLog[3, I*E^(a + b*x)])/b^3 - ((3*I)*d^3*PolyLog[4, (-I)*E^(a + b*x)])/b^4 + ((3*I)*d^3*PolyLog[4, I*E^(a + b*x)])/b^4 + (3*d*(c + d*x)^2*Sech[a + b*x])/(2*b^2) + ((c + d*x)^3*Sech[a + b*x]*Tanh[a + b*x])/(2*b)
```

**Rule 2317**

```
Int[Log[(a_) + (b_.)*((F_)^((e_.)*((c_.) + (d_.)*(x_))))^(n_.)], x_Symbol]
:> Dist[1/(d*e*n*Log[F]), Subst[Int[Log[a + b*x]/x, x], x, (F^(e*(c + d*x)))^n], x] /; FreeQ[{F, a, b, c, d, e, n}, x] && GtQ[a, 0]
```

**Rule 2320**

```
Int[u_, x_Symbol] :> With[{v = FunctionOfExponential[u, x]}, Dist[v/D[v, x], Subst[Int[FunctionOfExponentialFunction[u, x]/x, x], x, v], x] /; FunctionOfExponentialQ[u, x] && !MatchQ[u, (w_.)*((a_.)*(v_)^(n_.))^(m_.) /; FreeQ[{a, m, n}, x] && IntegerQ[m*n]] && !MatchQ[u, E^((c_.)*((a_.) + (b_.)*x))*
```

$(F\_)[v\_]$  /; FreeQ[{a, b, c}, x] && InverseFunctionQ[F[x]]

#### Rule 2438

Int[Log[(c\_.)\*((d\_) + (e\_.)\*(x\_)^(n\_.))]/(x\_), x\_Symbol] := Simp[-PolyLog[2, (-c)\*e\*x^n]/n, x] /; FreeQ[{c, d, e, n}, x] && EqQ[c\*d, 1]

#### Rule 2611

Int[Log[1 + (e\_.)\*((F\_)^((c\_.)\*((a\_.) + (b\_.)\*(x\_))))^(n\_.)]\*((f\_.) + (g\_.)\*(x\_)^(m\_.), x\_Symbol] := Simp[(-f + g\*x)^m\*(PolyLog[2, (-e)\*(F^(c\*(a + b\*x)))^n]/(b\*c\*n\*Log[F])), x] + Dist[g\*(m/(b\*c\*n\*Log[F])), Int[(f + g\*x)^(m - 1)\*PolyLog[2, (-e)\*(F^(c\*(a + b\*x)))^n], x], x] /; FreeQ[{F, a, b, c, e, f, g, n}, x] && GtQ[m, 0]

#### Rule 4265

Int[csc[(e\_.) + Pi\*(k\_.) + (Complex[0, fz\_])\*(f\_.)\*(x\_)]\*((c\_.) + (d\_.)\*(x\_)^(m\_.), x\_Symbol] := Simp[-2\*(c + d\*x)^m\*(ArcTanh[E^((-I)\*e + f\*fz\*x)/E^(I\*k\*Pi)]/(f\*fz\*I)), x] + (-Dist[d\*(m/(f\*fz\*I)), Int[(c + d\*x)^(m - 1)\*Log[1 - E^((-I)\*e + f\*fz\*x)/E^(I\*k\*Pi)], x], x] + Dist[d\*(m/(f\*fz\*I)), Int[(c + d\*x)^(m - 1)\*Log[1 + E^((-I)\*e + f\*fz\*x)/E^(I\*k\*Pi)], x], x]) /; FreeQ[{c, d, e, f, fz}, x] && IntegerQ[2\*k] && IGtQ[m, 0]

#### Rule 4271

Int[(csc[(e\_.) + (f\_.)\*(x\_)]\*(b\_.))^(n\_.)\*((c\_.) + (d\_.)\*(x\_)^(m\_.), x\_Symbol] := Simp[(-b^2)\*(c + d\*x)^m\*Cot[e + f\*x]\*((b\*Csc[e + f\*x])^(n - 2)/(f\*(n - 1))), x] + (Dist[b^2\*d^2\*m\*((m - 1)/(f^2\*(n - 1)\*(n - 2))), Int[(c + d\*x)^(m - 2)\*(b\*Csc[e + f\*x])^(n - 2), x], x] + Dist[b^2\*((n - 2)/(n - 1)), Int[(c + d\*x)^m\*(b\*Csc[e + f\*x])^(n - 2), x], x] - Simp[b^2\*d\*m\*(c + d\*x)^(m - 1)\*((b\*Csc[e + f\*x])^(n - 2)/(f^2\*(n - 1)\*(n - 2))), x]) /; FreeQ[{b, c, d, e, f}, x] && GtQ[n, 1] && NeQ[n, 2] && GtQ[m, 1]

#### Rule 6724

Int[PolyLog[n\_, (c\_.)\*((a\_.) + (b\_.)\*(x\_)^(p\_.))]/((d\_.) + (e\_.)\*(x\_)), x\_Symbol] := Simp[PolyLog[n + 1, c\*(a + b\*x)^p]/(e\*p), x] /; FreeQ[{a, b, c, d, e, n, p}, x] && EqQ[b\*d, a\*e]

#### Rule 6744

Int[((e\_.) + (f\_.)\*(x\_)^(m\_.))\*PolyLog[n\_, (d\_.)\*((F\_)^((c\_.)\*((a\_.) + (b\_.)\*(x\_))))^(p\_.)], x\_Symbol] := Simp[(e + f\*x)^m\*(PolyLog[n + 1, d\*(F^(c\*(a + b\*x)))^p]/(b\*c\*p\*Log[F])), x] - Dist[f\*(m/(b\*c\*p\*Log[F])), Int[(e + f\*x)^(m - 1)\*PolyLog[n + 1, d\*(F^(c\*(a + b\*x)))^p], x], x] /; FreeQ[{F, a, b, c,

d, e, f, n, p}, x] && GtQ[m, 0]

### Rubi steps

$$\begin{aligned}
 \int (c + dx)^3 \operatorname{sech}^3(a + bx) dx &= \frac{3d(c + dx)^2 \operatorname{sech}(a + bx)}{2b^2} + \frac{(c + dx)^3 \operatorname{sech}(a + bx) \tanh(a + bx)}{2b} + \frac{1}{2} \int (c + dx)^2 \operatorname{sech}^2(a + bx) dx \\
 &= -\frac{6d^2(c + dx) \tan^{-1}(e^{a+bx})}{b^3} + \frac{(c + dx)^3 \tan^{-1}(e^{a+bx})}{b} + \frac{3d(c + dx)^2 \operatorname{sech}(a + bx)}{2b^2} \\
 &= -\frac{6d^2(c + dx) \tan^{-1}(e^{a+bx})}{b^3} + \frac{(c + dx)^3 \tan^{-1}(e^{a+bx})}{b} - \frac{3id(c + dx)^2 \operatorname{Li}_2(-ie^{a+bx})}{2b^2} \\
 &= -\frac{6d^2(c + dx) \tan^{-1}(e^{a+bx})}{b^3} + \frac{(c + dx)^3 \tan^{-1}(e^{a+bx})}{b} + \frac{3id^3 \operatorname{Li}_2(-ie^{a+bx})}{b^4} \\
 &= -\frac{6d^2(c + dx) \tan^{-1}(e^{a+bx})}{b^3} + \frac{(c + dx)^3 \tan^{-1}(e^{a+bx})}{b} + \frac{3id^3 \operatorname{Li}_2(-ie^{a+bx})}{b^4} \\
 &= -\frac{6d^2(c + dx) \tan^{-1}(e^{a+bx})}{b^3} + \frac{(c + dx)^3 \tan^{-1}(e^{a+bx})}{b} + \frac{3id^3 \operatorname{Li}_2(-ie^{a+bx})}{b^4}
 \end{aligned}$$

### Mathematica [A]

time = 4.02, size = 455, normalized size = 1.54

Antiderivative was successfully verified.

[In] Integrate[(c + d\*x)^3\*Sech[a + b\*x]^3,x]

[Out] (I\*((-2\*I)\*b^3\*c^3\*ArcTan[E^(a + b\*x)] + (12\*I)\*b\*c\*d^2\*ArcTan[E^(a + b\*x)] + 3\*b^3\*c^2\*d\*x\*Log[1 - I\*E^(a + b\*x)] - 6\*b\*d^3\*x\*Log[1 - I\*E^(a + b\*x)] + 3\*b^3\*c\*d^2\*x^2\*Log[1 - I\*E^(a + b\*x)] + b^3\*d^3\*x^3\*Log[1 - I\*E^(a + b\*x)]) - 3\*b^3\*c^2\*d\*x\*Log[1 + I\*E^(a + b\*x)] + 6\*b\*d^3\*x\*Log[1 + I\*E^(a + b\*x)] - 3\*b^3\*c\*d^2\*x^2\*Log[1 + I\*E^(a + b\*x)] - b^3\*d^3\*x^3\*Log[1 + I\*E^(a + b\*x)]) - 3\*d\*(-2\*d^2 + b^2\*(c + d\*x)^2)\*PolyLog[2, (-I)\*E^(a + b\*x)] + 3\*d\*(-2\*d^2 + b^2\*(c + d\*x)^2)\*PolyLog[2, I\*E^(a + b\*x)] + 6\*b\*c\*d^2\*PolyLog[3, (-I)\*E^(a + b\*x)] + 6\*b\*d^3\*x\*PolyLog[3, (-I)\*E^(a + b\*x)] - 6\*b\*c\*d^2\*PolyLog[3, I\*E^(a + b\*x)] - 6\*b\*d^3\*x\*PolyLog[3, I\*E^(a + b\*x)] - 6\*d^3\*PolyLog[4, (-I)\*E^(a + b\*x)] + 6\*d^3\*PolyLog[4, I\*E^(a + b\*x)]) + b^2\*(c + d\*x)^2\*Sech[a + b\*x]\*(3\*d + b\*(c + d\*x)\*Tanh[a + b\*x])/(2\*b^4)

### Maple [F]

time = 0.07, size = 0, normalized size = 0.00

$$\int (dx + c)^3 \operatorname{sech}(bx + a)^3 dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `int((d*x+c)^3*sech(b*x+a)^3,x)`

[Out] `int((d*x+c)^3*sech(b*x+a)^3,x)`

**Maxima** [F]

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((d*x+c)^3*sech(b*x+a)^3,x, algorithm="maxima")`

[Out]  $b^2d^3 \int x^3 e^{(bx+a)} / (b^2 e^{(2bx+2a)} + b^2) dx + 3b^2c^2 d^2 \int x^2 e^{(bx+a)} / (b^2 e^{(2bx+2a)} + b^2) dx + 3b^2c^2 d \int x e^{(bx+a)} / (b^2 e^{(2bx+2a)} + b^2) dx - c^3 (\arctan(e^{(-bx-a)})/b - (e^{(-bx-a)} - e^{(-3bx-3a)}) / (b(2e^{(-2bx-2a)} + e^{(-4bx-4a)} + 1))) - 6d^3 \int x e^{(bx+a)} / (b^2 e^{(2bx+2a)} + b^2) dx - 6cd^2 \arctan(e^{(bx+a)})/b^3 + ((bd^3x^3e^{(3a)} + 3c^2d^2e^{(3a)} + 3(bcd^2 + d^3)x^2e^{(3a)} + 3(b^2cd + 2cd^2)x e^{(3a)})e^{(3bx)} - (bd^3x^3e^a - 3c^2d^2e^a + 3(bcd^2 - d^3)x^2e^a + 3(b^2cd - 2cd^2)x e^a)e^{(bx)}) / (b^2 e^{(4bx+4a)} + 2b^2 e^{(2bx+2a)} + b^2)$

**Fricas** [B] Both result and optimal contain complex but leaf count of result is larger than twice the leaf count of optimal. 4785 vs.  $2(242) = 484$ .

time = 0.47, size = 4785, normalized size = 16.17

Too large to display

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((d*x+c)^3*sech(b*x+a)^3,x, algorithm="fricas")`

[Out]  $1/2(2(b^3d^3x^3 + b^3c^3 + 3b^2c^2d + 3(b^3cd^2 + b^2d^3))x^2 + 3(b^3c^2d + 2b^2cd^2)x) \cosh(bx+a)^3 + 6(b^3d^3x^3 + b^3c^3 + 3b^2c^2d + 3(b^3cd^2 + b^2d^3))x^2 + 3(b^3c^2d + 2b^2cd^2)x \cosh(bx+a) \sinh(bx+a)^2 + 2(b^3d^3x^3 + b^3c^3 + 3b^2c^2d + 3(b^3cd^2 + b^2d^3))x^2 + 3(b^3c^2d + 2b^2cd^2)x \sinh(bx+a)^3 - 2(b^3d^3x^3 + b^3c^3 - 3b^2c^2d + 3(b^3cd^2 - b^2d^3))x^2 + 3(b^3c^2d - 2b^2cd^2)x \cosh(bx+a) - 3(-Ib^2d^3x^2 - 2Ib^2cd^2x - Ib^2c^2d + 2Ib^2d^3) \cosh(bx+a)^4 + 4(-Ib^2d^3x^2 - 2Ib^2cd^2x - Ib^2c^2d + 2Ib^2d^3) \cosh(bx+a) \sinh(bx+a)^3 + (-Ib^2d^3x^2 - 2Ib^2cd^2x - Ib^2c^2d + 2Ib^2d^3) \sinh(bx+a)^4 + 2Ib^2d^3 + 2(-Ib^2d^3x^2 - 2Ib^2cd^2x - Ib^2c^2d + 2Ib^2d^3) \cosh(bx+a)^2 + 2(-Ib^2d^3x$

$$\begin{aligned}
&^2 - 2*I*b^2*c*d^2*x - I*b^2*c^2*d + 2*I*d^3 + 3*(-I*b^2*d^3*x^2 - 2*I*b^2*c*d^2*x - I*b^2*c^2*d + 2*I*d^3)*\cosh(b*x + a)^2*\sinh(b*x + a)^2 + 4*((-I*b^2*d^3*x^2 - 2*I*b^2*c*d^2*x - I*b^2*c^2*d + 2*I*d^3)*\cosh(b*x + a)^3 + (-I*b^2*d^3*x^2 - 2*I*b^2*c*d^2*x - I*b^2*c^2*d + 2*I*d^3)*\cosh(b*x + a))*\sinh(b*x + a))*\operatorname{dilog}(I*\cosh(b*x + a) + I*\sinh(b*x + a)) - 3*(I*b^2*d^3*x^2 + 2*I*b^2*c*d^2*x + I*b^2*c^2*d + (I*b^2*d^3*x^2 + 2*I*b^2*c*d^2*x + I*b^2*c^2*d - 2*I*d^3)*\cosh(b*x + a)^4 + 4*(I*b^2*d^3*x^2 + 2*I*b^2*c*d^2*x + I*b^2*c^2*d - 2*I*d^3)*\cosh(b*x + a)*\sinh(b*x + a)^3 + (I*b^2*d^3*x^2 + 2*I*b^2*c*d^2*x + I*b^2*c^2*d - 2*I*d^3)*\sinh(b*x + a)^4 - 2*I*d^3 + 2*(I*b^2*d^3*x^2 + 2*I*b^2*c*d^2*x + I*b^2*c^2*d - 2*I*d^3)*\cosh(b*x + a)^2 + 2*(I*b^2*d^3*x^2 + 2*I*b^2*c*d^2*x + I*b^2*c^2*d - 2*I*d^3 + 3*(I*b^2*d^3*x^2 + 2*I*b^2*c*d^2*x + I*b^2*c^2*d - 2*I*d^3)*\cosh(b*x + a)^2)*\sinh(b*x + a)^2 + 4*((I*b^2*d^3*x^2 + 2*I*b^2*c*d^2*x + I*b^2*c^2*d - 2*I*d^3)*\cosh(b*x + a)^3 + (I*b^2*d^3*x^2 + 2*I*b^2*c*d^2*x + I*b^2*c^2*d - 2*I*d^3)*\cosh(b*x + a))*\sinh(b*x + a))*\operatorname{dilog}(-I*\cosh(b*x + a) - I*\sinh(b*x + a)) + (I*b^3*c^3 - 3*I*a*b^2*c^2*d + 3*I*(a^2 - 2)*b*c*d^2 + (I*b^3*c^3 - 3*I*a*b^2*c^2*d + 3*I*(a^2 - 2)*b*c*d^2 - I*(a^3 - 6*a)*d^3)*\cosh(b*x + a)^4 - 4*(-I*b^3*c^3 + 3*I*a*b^2*c^2*d - 3*I*(a^2 - 2)*b*c*d^2 + I*(a^3 - 6*a)*d^3)*\cosh(b*x + a)*\sinh(b*x + a)^3 + (I*b^3*c^3 - 3*I*a*b^2*c^2*d + 3*I*(a^2 - 2)*b*c*d^2 - I*(a^3 - 6*a)*d^3)*\sinh(b*x + a)^4 - I*(a^3 - 6*a)*d^3 - 2*(-I*b^3*c^3 + 3*I*a*b^2*c^2*d - 3*I*(a^2 - 2)*b*c*d^2 + I*(a^3 - 6*a)*d^3)*\cosh(b*x + a)^2 - 2*(-I*b^3*c^3 + 3*I*a*b^2*c^2*d - 3*I*(a^2 - 2)*b*c*d^2 + I*(a^3 - 6*a)*d^3 + 3*(-I*b^3*c^3 + 3*I*a*b^2*c^2*d - 3*I*(a^2 - 2)*b*c*d^2 + I*(a^3 - 6*a)*d^3)*\cosh(b*x + a)^2)*\sinh(b*x + a)^2 - 4*((-I*b^3*c^3 + 3*I*a*b^2*c^2*d - 3*I*(a^2 - 2)*b*c*d^2 + I*(a^3 - 6*a)*d^3)*\cosh(b*x + a)^3 + (-I*b^3*c^3 + 3*I*a*b^2*c^2*d - 3*I*(a^2 - 2)*b*c*d^2 + I*(a^3 - 6*a)*d^3)*\cosh(b*x + a))*\sinh(b*x + a))*\log(\cosh(b*x + a) + \sinh(b*x + a) + I) + (-I*b^3*c^3 + 3*I*a*b^2*c^2*d - 3*I*(a^2 - 2)*b*c*d^2 + (-I*b^3*c^3 + 3*I*a*b^2*c^2*d - 3*I*(a^2 - 2)*b*c*d^2 + I*(a^3 - 6*a)*d^3)*\cosh(b*x + a)^4 - 4*(I*b^3*c^3 - 3*I*a*b^2*c^2*d + 3*I*(a^2 - 2)*b*c*d^2 - I*(a^3 - 6*a)*d^3)*\cosh(b*x + a)*\sinh(b*x + a)^3 + (-I*b^3*c^3 + 3*I*a*b^2*c^2*d - 3*I*(a^2 - 2)*b*c*d^2 + I*(a^3 - 6*a)*d^3)*\sinh(b*x + a)^4 + I*(a^3 - 6*a)*d^3 - 2*(I*b^3*c^3 - 3*I*a*b^2*c^2*d + 3*I*(a^2 - 2)*b*c*d^2 - I*(a^3 - 6*a)*d^3)*\cosh(b*x + a)^2 - 2*(I*b^3*c^3 - 3*I*a*b^2*c^2*d + 3*I*(a^2 - 2)*b*c*d^2 - I*(a^3 - 6*a)*d^3 + 3*(I*b^3*c^3 - 3*I*a*b^2*c^2*d + 3*I*(a^2 - 2)*b*c*d^2 - I*(a^3 - 6*a)*d^3)*\cosh(b*x + a)^2)*\sinh(b*x + a)^2 - 4*((I*b^3*c^3 - 3*I*a*b^2*c^2*d + 3*I*(a^2 - 2)*b*c*d^2 - I*(a^3 - 6*a)*d^3)*\cosh(b*x + a)^3 + (I*b^3*c^3 - 3*I*a*b^2*c^2*d + 3*I*(a^2 - 2)*b*c*d^2 - I*(a^3 - 6*a)*d^3)*\cosh(b*x + a))*\sinh(b*x + a))*\log(\cosh(b*x + a) + \sinh(b*x + a) - I) + (-I*b^3*d^3*x^3 - 3*I*b^3*c*d^2*x^2 - 3*I*a*b^2*c^2*d + 3*I*a^2*b*c*d^2 + (-I*b^3*d^3*x^3 - 3*I*b^3*c*d^2*x^2 - 3*I*a*b^2*c^2*d + 3*I*a^2*b*c*d^2 - I*(a^3 - 6*a)*d^3 - 3*I*(b^3*c^2*d - 2*b*d^3)*x)*\cosh(b*x + a)^4 - 4*(I*b^3*d^3*x^3 + 3*I*b^3*c*d^2*x^2 + 3*I*a*b^2*c^2*d - 3*I*a^2*b*c*d^2 + I*(a^3 - 6*a)*d^3 + 3*I*(b^3*c^2*d - 2*b*d^3)*x)*\cosh(b*x + a)*\sinh(b*x + a)^3 + (-I*b^3*d^3*x^3 - 3*I*b^3*c*d^2*x^2 - 3*I*a*b^2*c^2*d + 3*I*a^2*b*c*d^2 - I*(a^3 - 6*a)*d^3 - 3*I*(b^3*c^2*d - 2
\end{aligned}$$

```

*b*d^3)*x)*sinh(b*x + a)^4 - I*(a^3 - 6*a)*d^3 - 2*(I*b^3*d^3*x^3 + 3*I*b^3
*c*d^2*x^2 + 3*I*a*b^2*c^2*d - 3*I*a^2*b*c*d^2 + I*(a^3 - 6*a)*d^3 + 3*I*(b
^3*c^2*d - 2*b*d^3)*x)*cosh(b*x + a)^2 - 2*(I*b^3*d^3*x^3 + 3*I*b^3*c*d^2*x
^2 + 3*I*a*b^2*c^2*d - 3*I*a^2*b*c*d^2 + I*(a^3 - 6*a)*d^3 + 3*(I*b^3*d^3*x
^3 + 3*I*b^3*c*d^2*x^2 + 3*I*a*b^2*c^2*d - 3*I*a^2*b*c*d^2 + I*(a^3 - 6*a)*
d^3 + 3*I*(b^3*c^2*d - 2*b*d^3)*x)*cosh(b*x + a)^2 + 3*I*(b^3*c^2*d - 2*b*d
^3)*x)*sinh(b*x + a)^2 - 3*I*(b^3*c^2*d - 2*b*d^3)*x - 4*((I*b^3*d^3*x^3 +
3*I*b^3*c*d^2*x^2 + 3*I*a*b^2*c^2*d - 3*I*a^2*b*c*d^2 + I*(a^3 - 6*a)*d^3 +
3*I*(b^3*c^2*d - 2*b*d^3)*x)*cosh(b*x + a)^3 + (I*b^3*d^3*x^3 + 3*I*b^3*c*
d^2*x^2 + 3*I*a*b^2*c^2*d - 3*I*a^2*b*c*d^2 + I...

```

**Sympy [F]**

time = 0.00, size = 0, normalized size = 0.00

$$\int (c + dx)^3 \operatorname{sech}^3(a + bx) dx$$

Verification of antiderivative is not currently implemented for this CAS.

```
[In] integrate((d*x+c)**3*sech(b*x+a)**3,x)
```

```
[Out] Integral((c + d*x)**3*sech(a + b*x)**3, x)
```

**Giac [F]**

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

```
[In] integrate((d*x+c)^3*sech(b*x+a)^3,x, algorithm="giac")
```

```
[Out] integrate((d*x + c)^3*sech(b*x + a)^3, x)
```

**Mupad [F]**

time = 0.00, size = -1, normalized size = -0.00

$$\int \frac{(c + dx)^3}{\cosh(a + bx)^3} dx$$

Verification of antiderivative is not currently implemented for this CAS.

```
[In] int((c + d*x)^3/cosh(a + b*x)^3,x)
```

```
[Out] int((c + d*x)^3/cosh(a + b*x)^3, x)
```



### 3.10 $\int (c + dx)^2 \operatorname{sech}^3(a + bx) dx$

**Optimal.** Leaf size=175

$$\frac{(c + dx)^2 \operatorname{ArcTan}(e^{a+bx})}{b} - \frac{d^2 \operatorname{ArcTan}(\sinh(a + bx))}{b^3} - \frac{id(c + dx) \operatorname{PolyLog}(2, -ie^{a+bx})}{b^2} + \frac{id(c + dx) \operatorname{PolyLog}(2, ie^{a+bx})}{b^2}$$

[Out]  $(d*x+c)^2*\arctan(\exp(b*x+a))/b-d^2*\arctan(\sinh(b*x+a))/b^3-I*d*(d*x+c)*\operatorname{polylog}(2,-I*\exp(b*x+a))/b^2+I*d*(d*x+c)*\operatorname{polylog}(2,I*\exp(b*x+a))/b^2+I*d^2*\operatorname{polylog}(3,-I*\exp(b*x+a))/b^3-I*d^2*\operatorname{polylog}(3,I*\exp(b*x+a))/b^3+d*(d*x+c)*\operatorname{sech}(b*x+a)/b^2+1/2*(d*x+c)^2*\operatorname{sech}(b*x+a)*\tanh(b*x+a)/b$

**Rubi [A]**

time = 0.10, antiderivative size = 175, normalized size of antiderivative = 1.00, number of steps used = 9, number of rules used = 6, integrand size = 16,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.375$ ,

Rules used = {4271, 3855, 4265, 2611, 2320, 6724}

$$\frac{d^2 \operatorname{ArcTan}(\sinh(a + bx))}{b^3} + \frac{(c + dx)^2 \operatorname{ArcTan}(e^{a+bx})}{b} + \frac{id^2 \operatorname{Li}_3(-ie^{a+bx})}{b^3} - \frac{id^2 \operatorname{Li}_3(ie^{a+bx})}{b^3} - \frac{id(c + dx) \operatorname{Li}_2(-ie^{a+bx})}{b^2} + \frac{id(c + dx) \operatorname{Li}_2(ie^{a+bx})}{b^2} + \frac{d(c + dx) \operatorname{sech}(a + bx)}{b^2} + \frac{(c + dx)^2 \tanh(a + bx) \operatorname{sech}(a + bx)}{2b}$$

Antiderivative was successfully verified.

[In]  $\operatorname{Int}[(c + d*x)^2*\operatorname{Sech}[a + b*x]^3,x]$

[Out]  $((c + d*x)^2*\operatorname{ArcTan}[E^{(a + b*x)}])/b - (d^2*\operatorname{ArcTan}[\operatorname{Sinh}[a + b*x]])/b^3 - (I*d*(c + d*x)*\operatorname{PolyLog}[2, (-I)*E^{(a + b*x)}])/b^2 + (I*d*(c + d*x)*\operatorname{PolyLog}[2, I*E^{(a + b*x)}])/b^2 + (I*d^2*\operatorname{PolyLog}[3, (-I)*E^{(a + b*x)}])/b^3 - (I*d^2*\operatorname{PolyLog}[3, I*E^{(a + b*x)}])/b^3 + (d*(c + d*x)*\operatorname{Sech}[a + b*x])/b^2 + ((c + d*x)^2*\operatorname{Sech}[a + b*x]*\operatorname{Tanh}[a + b*x])/(2*b)$

**Rule 2320**

```
Int[u_, x_Symbol] := With[{v = FunctionOfExponential[u, x]}, Dist[v/D[v, x], Subst[Int[FunctionOfExponentialFunction[u, x]/x, x], x, v], x] /; FunctionOfExponentialQ[u, x] && !MatchQ[u, (w_)*((a_.)*(v_)^(n_))^(m_) /; FreeQ[{a, m, n}, x] && IntegerQ[m*n]] && !MatchQ[u, E^((c_.)*((a_.) + (b_.)*x))*(F_) [v_] /; FreeQ[{a, b, c}, x] && InverseFunctionQ[F[x]]]
```

**Rule 2611**

```
Int[Log[1 + (e_.)*((F_)^((c_.)*((a_.) + (b_.)*(x_)))^(n_.)]*((f_.) + (g_.)*(x_))^(m_.), x_Symbol] := Simp[(-(f + g*x)^m)*(PolyLog[2, (-e)*(F^(c*(a + b*x)))^n]/(b*c*n*Log[F])), x] + Dist[g*(m/(b*c*n*Log[F])), Int[(f + g*x)^(m - 1)*PolyLog[2, (-e)*(F^(c*(a + b*x)))^n], x], x] /; FreeQ[{F, a, b, c, e, f, g, n}, x] && GtQ[m, 0]
```

**Rule 3855**

```
Int[csc[(c_.) + (d_.)*(x_)], x_Symbol] := Simp[-ArcTanh[Cos[c + d*x]]/d, x]
/; FreeQ[{c, d}, x]
```

### Rule 4265

```
Int[csc[(e_.) + Pi*(k_.) + (Complex[0, fz_])*(f_.)*(x_)]*((c_.) + (d_.)*(x_
))^m_], x_Symbol] := Simp[-2*(c + d*x)^m*(ArcTanh[E^((-I)*e + f*fz*x)/E^(
I*k*Pi)]/(f*fz*I)), x] + (-Dist[d*(m/(f*fz*I)), Int[(c + d*x)^(m - 1)*Log[1
- E^((-I)*e + f*fz*x)/E^(I*k*Pi)], x], x] + Dist[d*(m/(f*fz*I)), Int[(c +
d*x)^(m - 1)*Log[1 + E^((-I)*e + f*fz*x)/E^(I*k*Pi)], x], x]) /; FreeQ[{c,
d, e, f, fz}, x] && IntegerQ[2*k] && IGtQ[m, 0]
```

### Rule 4271

```
Int[(csc[(e_.) + (f_.)*(x_)]*(b_.))^n_)*((c_.) + (d_.)*(x_))^m_, x_Symbo
l] := Simp[(-b^2)*(c + d*x)^m*Cot[e + f*x]*((b*Csc[e + f*x])^(n - 2)/(f*(n
- 1))), x] + (Dist[b^2*d^2*m*((m - 1)/(f^2*(n - 1)*(n - 2))), Int[(c + d*x)
^(m - 2)*(b*Csc[e + f*x])^(n - 2), x], x] + Dist[b^2*((n - 2)/(n - 1)), Int
[(c + d*x)^m*(b*Csc[e + f*x])^(n - 2), x], x] - Simp[b^2*d*m*(c + d*x)^(m -
1)*((b*Csc[e + f*x])^(n - 2)/(f^2*(n - 1)*(n - 2))), x]) /; FreeQ[{b, c, d
, e, f}, x] && GtQ[n, 1] && NeQ[n, 2] && GtQ[m, 1]
```

### Rule 6724

```
Int[PolyLog[n_, (c_.)*((a_.) + (b_.)*(x_))^p_]/((d_.) + (e_.)*(x_)), x_S
ymbol] := Simp[PolyLog[n + 1, c*(a + b*x)^p]/(e*p), x] /; FreeQ[{a, b, c, d
, e, n, p}, x] && EqQ[b*d, a*e]
```

### Rubi steps

$$\begin{aligned}
\int (c + dx)^2 \operatorname{sech}^3(a + bx) dx &= \frac{d(c + dx) \operatorname{sech}(a + bx)}{b^2} + \frac{(c + dx)^2 \operatorname{sech}(a + bx) \tanh(a + bx)}{2b} + \frac{1}{2} \int (c + dx)^2 \operatorname{sech}^2(a + bx) dx \\
&= \frac{(c + dx)^2 \tan^{-1}(e^{a+bx})}{b} - \frac{d^2 \tan^{-1}(\sinh(a + bx))}{b^3} + \frac{d(c + dx) \operatorname{sech}(a + bx)}{b^2} + \frac{1}{2} \int (c + dx)^2 \operatorname{sech}^2(a + bx) dx \\
&= \frac{(c + dx)^2 \tan^{-1}(e^{a+bx})}{b} - \frac{d^2 \tan^{-1}(\sinh(a + bx))}{b^3} - \frac{id(c + dx) \operatorname{Li}_2(-ie^{a+bx})}{b^2} + \frac{1}{2} \int (c + dx)^2 \operatorname{sech}^2(a + bx) dx \\
&= \frac{(c + dx)^2 \tan^{-1}(e^{a+bx})}{b} - \frac{d^2 \tan^{-1}(\sinh(a + bx))}{b^3} - \frac{id(c + dx) \operatorname{Li}_2(-ie^{a+bx})}{b^2} + \frac{1}{2} \int (c + dx)^2 \operatorname{sech}^2(a + bx) dx \\
&= \frac{(c + dx)^2 \tan^{-1}(e^{a+bx})}{b} - \frac{d^2 \tan^{-1}(\sinh(a + bx))}{b^3} - \frac{id(c + dx) \operatorname{Li}_2(-ie^{a+bx})}{b^2} + \frac{1}{2} \int (c + dx)^2 \operatorname{sech}^2(a + bx) dx
\end{aligned}$$

### Mathematica [A]

time = 1.37, size = 270, normalized size = 1.54

$(-2b^2 \operatorname{ArcTan}(e^{bx+a}) + 4b^2 \operatorname{ArcTan}(e^{2bx+2a}) + 2b^2 dx \log(1 - e^{bx+a}) + b^2 d^2 \log(1 - e^{2bx+2a}) - 2b^2 dx \log(1 + e^{bx+a}) - b^2 d^2 \log(1 + e^{2bx+2a}) - 2b(c+dx) \operatorname{PolyLog}(2, -e^{bx+a}) + 2b(c+dx) \operatorname{PolyLog}(2, e^{bx+a}) + 2b^2 \operatorname{PolyLog}(3, -e^{bx+a}) - 2b^2 \operatorname{PolyLog}(3, e^{bx+a}) + b^2(c+dx)^2 \operatorname{sech}^2(a+bx) \sinh(bx) + b(c+dx) \operatorname{sech}(a+bx) (2d+bc+dx) \tanh(a))$

Antiderivative was successfully verified.

[In] Integrate[(c + d\*x)^2\*Sech[a + b\*x]^3,x]

[Out]  $(I*((-2*I)*b^2*c^2*\operatorname{ArcTan}[E^{(a+b*x)}] + (4*I)*d^2*\operatorname{ArcTan}[E^{(a+b*x)}] + 2*b^2*c*d*x*\operatorname{Log}[1 - I*E^{(a+b*x)}] + b^2*d^2*x^2*\operatorname{Log}[1 - I*E^{(a+b*x)}] - 2*b^2*c*d*x*\operatorname{Log}[1 + I*E^{(a+b*x)}] - b^2*d^2*x^2*\operatorname{Log}[1 + I*E^{(a+b*x)}] - 2*b*d*(c+d*x)*\operatorname{PolyLog}[2, (-I)*E^{(a+b*x)}] + 2*b*d*(c+d*x)*\operatorname{PolyLog}[2, I*E^{(a+b*x)}] + 2*d^2*\operatorname{PolyLog}[3, (-I)*E^{(a+b*x)}] - 2*d^2*\operatorname{PolyLog}[3, I*E^{(a+b*x)}]) + b^2*(c+d*x)^2*\operatorname{Sech}[a]*\operatorname{Sech}[a+b*x]^2*\operatorname{Sinh}[b*x] + b*(c+d*x)*\operatorname{Sech}[a+b*x]*(2*d+b*(c+d*x)*\operatorname{Tanh}[a]))/(2*b^3)$

Maple [F]

time = 0.07, size = 0, normalized size = 0.00

$$\int (dx + c)^2 \operatorname{sech}(bx + a)^3 dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((d\*x+c)^2\*sech(b\*x+a)^3,x)

[Out] int((d\*x+c)^2\*sech(b\*x+a)^3,x)

Maxima [F]

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)^2\*sech(b\*x+a)^3,x, algorithm="maxima")

[Out]  $b^2*d^2*\operatorname{integrate}(x^2*e^{(b*x+a)}/(b^2*e^{(2*b*x+2*a)}+b^2), x) + 2*b^2*c*d*\operatorname{integrate}(x*e^{(b*x+a)}/(b^2*e^{(2*b*x+2*a)}+b^2), x) - c^2*(\operatorname{arctan}(e^{(-b*x-a)})/b - (e^{(-b*x-a)} - e^{(-3*b*x-3*a)})/(b*(2*e^{(-2*b*x-2*a)} + e^{(-4*b*x-4*a)} + 1))) + ((b*d^2*x^2*e^{(3*a)} + 2*c*d*e^{(3*a)} + 2*(b*c*d + d^2)*x*e^{(3*a)})*e^{(3*b*x)} - (b*d^2*x^2*e^a - 2*c*d*e^a + 2*(b*c*d - d^2)*x*e^a)*e^{(b*x)})/(b^2*e^{(4*b*x+4*a)} + 2*b^2*e^{(2*b*x+2*a)} + b^2) - 2*d^2*a*\operatorname{rctan}(e^{(b*x+a)})/b^3$

Fricas [B] Both result and optimal contain complex but leaf count of result is larger than twice the leaf count of optimal. 2651 vs.  $2(150) = 300$ .

time = 0.44, size = 2651, normalized size = 15.15

Too large to display



$$\begin{aligned} &^2*c*d*x + 2*I*a*b*c*d - I*a^2*d^2)*\cosh(b*x + a))*\sinh(b*x + a))*\log(I*\cos \\ &h(b*x + a) + I*\sinh(b*x + a) + 1) + (I*b^2*d^2*x^2 + 2*I*b^2*c*d*x + (I*b^2 \\ &*d^2*x^2 + 2*I*b^2*c*d*x + 2*I*a*b*c*d - I*a^2*d^2)*\cosh(b*x + a)^4 - 4*(-I \\ &*b^2*d^2*x^2 - 2*I*b^2*c*d*x - 2*I*a*b*c*d + I*a^2*d^2)*\cosh(b*x + a)*\sinh( \\ &b*x + a)^3 + (I*b^2*d^2*x^2 + 2*I*b^2*c*d*x + 2*I*a*b*c*d - I*a^2*d^2)*\sinh \\ &(b*x + a)^4 + 2*I*a*b*c*d - I*a^2*d^2 - 2*(-I*b^2*d^2*x^2 - 2*I*b^2*c*d*x - \\ &2*I*a*b*c*d + I*a^2*d^2)*\cosh(b*x + a)^2 - 2*(-I*b^2*d^2*x^2 - 2*I*b^2*c*d \\ &*x - 2*I*a*b*c*d + I*a^2*d^2 + 3*(-I*b^2*d^2*x^2 - 2*I*b^2*c*d*x - 2*I*a*b* \\ &c*d + I*a^2*d^2)*\cosh(b*x + a)^2)*\sinh(b*x + a)^2 - 4*((-I*b^2*d^2*x^2 - 2* \\ &I*b^2*c*d*x - 2*I*a*b*c*d + I*a^2*d^2)*\cosh(b*x + a)^3 + (-I*b^2*d^2*x^2 - \\ &2*I*b^2*c*d*x - 2*I*a*b*c*d + I*a^2*d^2)*\cosh(b*x + a))*\sinh(b*x + a))*\log( \\ &-I*\cosh(b*x + a) - I*\sinh(b*x + a) + 1) - 2*(I*d^2*\cosh(b*x + a)^4 + 4*I*d^ \\ &2*\cosh(b*x + a)*\sinh(b*x + a)^3 + I*d^2*\sinh(b*x + a)^4 + 2*I*d^2*\cosh(b*x \\ &+ a)^2 + 2*(3*I*d^2*\cosh(b*x + a)^2 + I*d^2)*\sinh(b*x + a)^2 + I*d^2 + 4*(I \\ &*d^2*\cosh(b*x + a)^3 + I*d^2*\cosh(b*x + a))*\sinh(b*x + a))*\text{polylog}(3, I*\cos \\ &h(b*x + a) + I*\sinh(b*x + a)) - 2*(-I*d^2*\cosh(b*x + a)^4 - 4*I*d^2*\cosh(b* \\ &x + a)*\sinh(b*x + a)^3 - I*d^2*\sinh(b*x + a)^4 - 2*I*d^2*\cosh(b*x + a)^2 + \\ &2*(-3*I*d^2*\cosh(b*x + a)^2 - I*d^2)*\sinh(b*x + a)^2 - I*d^2 + 4*(-I*d^2*\co \\ &sh(b*x + a)^3 - I*d^2*\cosh(b*x + a))*\sinh(b*x + a))*\text{polylog}(3, -I*\cosh(b*x \\ &+ a) - I*\sinh(b*x + a)) - 2*(b^2*d^2*x^2 + b^2*c^2 - 2*b*c*d - 3*(b^2*d^2*x \\ &^2 + b^2*c^2 + 2*b*c*d + 2*(b^2*c*d + b*d^2)*x)*\cosh(b*x + a)^2 + 2*(b^2*c* \\ &d - b*d^2)*x)*\sinh(b*x + a))/(b^3*\cosh(b*x + a)^4 + 4*b^3*\cosh(b*x + a)*\sin \\ &h(b*x + a)^3 + b^3*\sinh(b*x + a)^4 + 2*b^3*\cosh(b*x + a)^2 + b^3 + 2*(3*b^3 \\ &*\cosh(b*x + a)^2 + b^3)*\sinh(b*x + a)^2 + 4*(b^3*\cosh(b*x + a)^3 + b^3*\cosh \\ &(b*x + a))*\sinh(b*x + a)) \end{aligned}$$

**Sympy** [F]

time = 0.00, size = 0, normalized size = 0.00

$$\int (c + dx)^2 \operatorname{sech}^3(a + bx) dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*\*2\*sech(b\*x+a)\*\*3,x)

[Out] Integral((c + d\*x)\*\*2\*sech(a + b\*x)\*\*3, x)

**Giac** [F]

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)^2\*sech(b\*x+a)^3,x, algorithm="giac")

[Out] integrate((d\*x + c)^2\*sech(b\*x + a)^3, x)

**Mupad [F]**

time = 0.00, size = -1, normalized size = -0.01

$$\int \frac{(c + dx)^2}{\cosh(a + bx)^3} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((c + d\*x)^2/cosh(a + b\*x)^3,x)

[Out] int((c + d\*x)^2/cosh(a + b\*x)^3, x)

### 3.11 $\int (c + dx) \operatorname{sech}^3(a + bx) dx$

Optimal. Leaf size=102

$$\frac{(c + dx) \operatorname{ArcTan}(e^{a+bx})}{b} - \frac{id \operatorname{PolyLog}(2, -ie^{a+bx})}{2b^2} + \frac{id \operatorname{PolyLog}(2, ie^{a+bx})}{2b^2} + \frac{d \operatorname{sech}(a + bx)}{2b^2} + \frac{(c + dx) \operatorname{sech}(a + bx)}{2b}$$

[Out] (d\*x+c)\*arctan(exp(b\*x+a))/b-1/2\*I\*d\*polylog(2,-I\*exp(b\*x+a))/b^2+1/2\*I\*d\*polylog(2,I\*exp(b\*x+a))/b^2+1/2\*d\*sech(b\*x+a)/b^2+1/2\*(d\*x+c)\*sech(b\*x+a)\*tanh(b\*x+a)/b

Rubi [A]

time = 0.05, antiderivative size = 102, normalized size of antiderivative = 1.00, number of steps used = 6, number of rules used = 4, integrand size = 14,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.286$ , Rules used = {4270, 4265, 2317, 2438}

$$\frac{(c + dx) \operatorname{ArcTan}(e^{a+bx})}{b} - \frac{id \operatorname{Li}_2(-ie^{a+bx})}{2b^2} + \frac{id \operatorname{Li}_2(ie^{a+bx})}{2b^2} + \frac{d \operatorname{sech}(a + bx)}{2b^2} + \frac{(c + dx) \tanh(a + bx) \operatorname{sech}(a + bx)}{2b}$$

Antiderivative was successfully verified.

[In] Int[(c + d\*x)\*Sech[a + b\*x]^3,x]

[Out] ((c + d\*x)\*ArcTan[E^(a + b\*x)]/b - ((I/2)\*d\*PolyLog[2, (-I)\*E^(a + b\*x)]/b^2 + ((I/2)\*d\*PolyLog[2, I\*E^(a + b\*x)]/b^2 + (d\*Sech[a + b\*x])/(2\*b^2) + ((c + d\*x)\*Sech[a + b\*x]\*Tanh[a + b\*x])/(2\*b))

Rule 2317

Int[Log[(a\_) + (b\_.)\*((F\_)^(e\_.)\*((c\_.) + (d\_.)\*(x\_)))^(n\_.)], x\_Symbol] :> Dist[1/(d\*e\*n\*Log[F]), Subst[Int[Log[a + b\*x]/x, x], x, (F^(e\*(c + d\*x)))^n], x] /; FreeQ[{F, a, b, c, d, e, n}, x] && GtQ[a, 0]

Rule 2438

Int[Log[(c\_.)\*((d\_) + (e\_.)\*(x\_)^(n\_.))]/(x\_), x\_Symbol] :> Simp[-PolyLog[2, (-c)\*e\*x^n]/n, x] /; FreeQ[{c, d, e, n}, x] && EqQ[c\*d, 1]

Rule 4265

Int[csc[(e\_.) + Pi\*(k\_.) + (Complex[0, fz\_])\*(f\_.)\*(x\_)]\*((c\_.) + (d\_.)\*(x\_))^(m\_.), x\_Symbol] :> Simp[-2\*(c + d\*x)^m\*(ArcTanh[E^((-I)\*e + f\*fz\*x)/E^(I\*k\*Pi)]/(f\*fz\*I)), x] + (-Dist[d\*(m/(f\*fz\*I)), Int[(c + d\*x)^(m - 1)\*Log[1 - E^((-I)\*e + f\*fz\*x)/E^(I\*k\*Pi)], x], x] + Dist[d\*(m/(f\*fz\*I)), Int[(c + d\*x)^(m - 1)\*Log[1 + E^((-I)\*e + f\*fz\*x)/E^(I\*k\*Pi)], x], x]) /; FreeQ[{c, d, e, f, fz}, x] && IntegerQ[2\*k] && IGtQ[m, 0]

Rule 4270

```
Int[(csc[(e_.) + (f_.)*(x_)]*(b_.))^(n_)*((c_.) + (d_.)*(x_)), x_Symbol] :=
  Simp[(-b^2)*(c + d*x)*Cot[e + f*x]*((b*Csc[e + f*x])^(n - 2)/(f*(n - 1))),
  x] + (Dist[b^2*((n - 2)/(n - 1)), Int[(c + d*x)*(b*Csc[e + f*x])^(n - 2),
  x], x] - Simp[b^2*d*((b*Csc[e + f*x])^(n - 2)/(f^2*(n - 1)*(n - 2))), x]) /
; FreeQ[{b, c, d, e, f}, x] && GtQ[n, 1] && NeQ[n, 2]
```

Rubi steps

$$\begin{aligned} \int (c + dx) \operatorname{sech}^3(a + bx) dx &= \frac{d \operatorname{sech}(a + bx)}{2b^2} + \frac{(c + dx) \operatorname{sech}(a + bx) \tanh(a + bx)}{2b} + \frac{1}{2} \int (c + dx) \operatorname{sech}(a + bx) dx \\ &= \frac{(c + dx) \tan^{-1}(e^{a+bx})}{b} + \frac{d \operatorname{sech}(a + bx)}{2b^2} + \frac{(c + dx) \operatorname{sech}(a + bx) \tanh(a + bx)}{2b} \\ &= \frac{(c + dx) \tan^{-1}(e^{a+bx})}{b} + \frac{d \operatorname{sech}(a + bx)}{2b^2} + \frac{(c + dx) \operatorname{sech}(a + bx) \tanh(a + bx)}{2b} \\ &= \frac{(c + dx) \tan^{-1}(e^{a+bx})}{b} - \frac{id \operatorname{Li}_2(-ie^{a+bx})}{2b^2} + \frac{id \operatorname{Li}_2(ie^{a+bx})}{2b^2} + \frac{d \operatorname{sech}(a + bx)}{2b^2} + \end{aligned}$$

**Mathematica [A]**

time = 2.11, size = 178, normalized size = 1.75

$\frac{bc \operatorname{ArcTan}(\sinh(a + bx)) + \frac{1}{2} d (-(-2ia + \pi - 2ibx) (\log(1 - ie^{a+bx}) - \log(1 + ie^{a+bx}))) + (-2ia + \pi) \log(\cot(\frac{1}{2}(2ia + \pi + 2ibx))) - 2(\operatorname{PolyLog}(2, -ie^{a+bx}) - \operatorname{PolyLog}(2, ie^{a+bx})) + b d \operatorname{sech}(a) \operatorname{sech}^2(a + bx) \sinh(bx) + d \operatorname{sech}(a + bx) (1 + bx \tanh(a)) + b c \operatorname{sech}(a + bx) \tanh(a + bx)}{2b^2}$

Antiderivative was successfully verified.

[In] Integrate[(c + d\*x)\*Sech[a + b\*x]^3,x]

[Out] (b\*c\*ArcTan[Sinh[a + b\*x]] + (d\*(-((-2\*I)\*a + Pi - (2\*I)\*b\*x)\*(Log[1 - I\*E^(a + b\*x)] - Log[1 + I\*E^(a + b\*x)])) + ((-2\*I)\*a + Pi)\*Log[Cot[((2\*I)\*a + Pi + (2\*I)\*b\*x)/4]] - (2\*I)\*(PolyLog[2, (-I)\*E^(a + b\*x)] - PolyLog[2, I\*E^(a + b\*x)]))/2 + b\*d\*x\*Sech[a]\*Sech[a + b\*x]^2\*Sinh[b\*x] + d\*Sech[a + b\*x]\*(1 + b\*x\*Tanh[a]) + b\*c\*Sech[a + b\*x]\*Tanh[a + b\*x])/(2\*b^2)

**Maple [B]** Both result and optimal contain complex but leaf count of result is larger than twice the leaf count of optimal. 215 vs. 2(87) = 174.

time = 0.99, size = 216, normalized size = 2.12

method	result
risch	$\frac{e^{bx+a} (bdx e^{2bx+2a} + bc e^{2bx+2a} - bdx + e^{2bx+2a} d - bc + d)}{b^2 (e^{2bx+2a} + 1)^2} + \frac{c \operatorname{arctan}(e^{bx+a})}{b} - \frac{id \ln(1 + ie^{bx+a}) x}{2b} - \frac{id \ln(1 + ie^{bx+a}) a}{2b^2} + \frac{id \ln(1 - ie^{bx+a}) x}{2b} + \frac{id \ln(1 - ie^{bx+a}) a}{2b^2}$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((d\*x+c)\*sech(b\*x+a)^3,x,method=\_RETURNVERBOSE)



[Out]  $\frac{\exp(b*x+a)*(b*d*x*\exp(2*b*x+2*a)+b*c*\exp(2*b*x+2*a)-b*d*x*\exp(2*b*x+2*a)*d-b*c+d)}{b^2*(\exp(2*b*x+2*a)+1)^2+1/b*c*\arctan(\exp(b*x+a))-1/2*I/b*d*\ln(1+I*\exp(b*x+a))*x-1/2*I/b^2*d*\ln(1+I*\exp(b*x+a))*a+1/2*I/b*d*\ln(1-I*\exp(b*x+a))*x+1/2*I/b^2*d*\ln(1-I*\exp(b*x+a))*a-1/2*I/b^2*d*dilog(1+I*\exp(b*x+a))+1/2*I/b^2*d*dilog(1-I*\exp(b*x+a))-1/b^2*d*a*\arctan(\exp(b*x+a))}$

**Maxima** [F]

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((d*x+c)*sech(b*x+a)^3,x, algorithm="maxima")`

[Out]  $d*(((b*x*e^{3*a} + e^{3*a})*e^{3*b*x} - (b*x*e^a - e^a)*e^{b*x}))/b^2*e^{4*b*x + 4*a} + 2*b^2*e^{2*b*x + 2*a} + b^2) + 8*\integrate(1/8*x*e^{b*x + a}/(e^{2*b*x + 2*a} + 1), x) - c*(\arctan(e^{-b*x - a})/b - (e^{-b*x - a} - e^{-3*b*x - 3*a}))/b*(2*e^{-2*b*x - 2*a} + e^{-4*b*x - 4*a} + 1))$

**Fricas** [B] Both result and optimal contain complex but leaf count of result is larger than twice the leaf count of optimal. 1267 vs.  $2(81) = 162$ .

time = 0.39, size = 1267, normalized size = 12.42

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate((d*x+c)*sech(b*x+a)^3,x, algorithm="fricas")`

[Out]  $\frac{1}{2}*(2*(b*d*x + b*c + d)*\cosh(b*x + a)^3 + 6*(b*d*x + b*c + d)*\cosh(b*x + a)*\sinh(b*x + a)^2 + 2*(b*d*x + b*c + d)*\sinh(b*x + a)^3 - 2*(b*d*x + b*c - d)*\cosh(b*x + a) + (I*d*\cosh(b*x + a)^4 + 4*I*d*\cosh(b*x + a)*\sinh(b*x + a)^3 + I*d*\sinh(b*x + a)^4 + 2*I*d*\cosh(b*x + a)^2 - 2*(-3*I*d*\cosh(b*x + a)^2 - I*d)*\sinh(b*x + a)^2 - 4*(-I*d*\cosh(b*x + a)^3 - I*d*\cosh(b*x + a))*\sinh(b*x + a) + I*d)*dilog(I*\cosh(b*x + a) + I*\sinh(b*x + a)) + (-I*d*\cosh(b*x + a)^4 - 4*I*d*\cosh(b*x + a)*\sinh(b*x + a)^3 - I*d*\sinh(b*x + a)^4 - 2*I*d*\cosh(b*x + a)^2 - 2*(3*I*d*\cosh(b*x + a)^2 + I*d)*\sinh(b*x + a)^2 - 4*(I*d*\cosh(b*x + a)^3 + I*d*\cosh(b*x + a))*\sinh(b*x + a) - I*d)*dilog(-I*\cosh(b*x + a) - I*\sinh(b*x + a)) + ((I*b*c - I*a*d)*\cosh(b*x + a)^4 - 4*(-I*b*c + I*a*d)*\cosh(b*x + a)*\sinh(b*x + a)^3 + (I*b*c - I*a*d)*\sinh(b*x + a)^4 - 2*(-I*b*c + I*a*d)*\cosh(b*x + a)^2 - 2*(3*(-I*b*c + I*a*d)*\cosh(b*x + a)^2 - I*b*c + I*a*d)*\sinh(b*x + a)^2 + I*b*c - I*a*d - 4*((-I*b*c + I*a*d)*\cosh(b*x + a)^3 + (-I*b*c + I*a*d)*\cosh(b*x + a))*\sinh(b*x + a))*\log(\cosh(b*x + a) + \sinh(b*x + a) + I) + ((-I*b*c + I*a*d)*\cosh(b*x + a)^4 - 4*(I*b*c - I*a*d)*\cosh(b*x + a)*\sinh(b*x + a)^3 + (-I*b*c + I*a*d)*\sinh(b*x + a)^4 - 2*(I*b*c - I*a*d)*\cosh(b*x + a)^2 - 2*(3*(I*b*c - I*a*d)*\cosh(b*x + a)^2 + I*b*c - I*a*d)*\sinh(b*x + a)^2 - I*b*c + I*a*d - 4*((I*b*c - I*a*d)*\cosh(b*x +$

$a)^3 + (I*b*c - I*a*d)*\cosh(b*x + a))*\sinh(b*x + a))*\log(\cosh(b*x + a) + \sinh(b*x + a) - I) + ((-I*b*d*x - I*a*d)*\cosh(b*x + a)^4 - 4*(I*b*d*x + I*a*d)*\cosh(b*x + a)*\sinh(b*x + a)^3 + (-I*b*d*x - I*a*d)*\sinh(b*x + a)^4 - I*b*d*x - 2*(I*b*d*x + I*a*d)*\cosh(b*x + a)^2 - 2*(I*b*d*x + 3*(I*b*d*x + I*a*d))*\cosh(b*x + a)^2 + I*a*d)*\sinh(b*x + a)^2 - I*a*d - 4*((I*b*d*x + I*a*d)*\cosh(b*x + a)^3 + (I*b*d*x + I*a*d)*\cosh(b*x + a))*\sinh(b*x + a))*\log(I*\cosh(b*x + a) + I*\sinh(b*x + a) + 1) + ((I*b*d*x + I*a*d)*\cosh(b*x + a)^4 - 4*(-I*b*d*x - I*a*d)*\cosh(b*x + a)*\sinh(b*x + a)^3 + (I*b*d*x + I*a*d)*\sinh(b*x + a)^4 + I*b*d*x - 2*(-I*b*d*x - I*a*d)*\cosh(b*x + a)^2 - 2*(-I*b*d*x + 3*(-I*b*d*x - I*a*d)*\cosh(b*x + a)^2 - I*a*d)*\sinh(b*x + a)^2 + I*a*d - 4*((-I*b*d*x - I*a*d)*\cosh(b*x + a)^3 + (-I*b*d*x - I*a*d)*\cosh(b*x + a))*\sinh(b*x + a))*\log(-I*\cosh(b*x + a) - I*\sinh(b*x + a) + 1) - 2*(b*d*x - 3*(b*d*x + b*c + d)*\cosh(b*x + a)^2 + b*c - d)*\sinh(b*x + a))/(b^2*\cosh(b*x + a)^4 + 4*b^2*\cosh(b*x + a)*\sinh(b*x + a)^3 + b^2*\sinh(b*x + a)^4 + 2*b^2*\cosh(b*x + a)^2 + 2*(3*b^2*\cosh(b*x + a)^2 + b^2)*\sinh(b*x + a)^2 + b^2 + 4*(b^2*\cosh(b*x + a)^3 + b^2*\cosh(b*x + a))*\sinh(b*x + a))$

**Sympy** [F]

time = 0.00, size = 0, normalized size = 0.00

$$\int (c + dx) \operatorname{sech}^3(a + bx) dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*sech(b\*x+a)\*\*3,x)

[Out] Integral((c + d\*x)\*sech(a + b\*x)\*\*3, x)

**Giac** [F]

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate((d\*x+c)\*sech(b\*x+a)^3,x, algorithm="giac")

[Out] integrate((d\*x + c)\*sech(b\*x + a)^3, x)

**Mupad** [F]

time = 0.00, size = -1, normalized size = -0.01

$$\int \frac{c + dx}{\cosh(a + bx)^3} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int((c + d\*x)/cosh(a + b\*x)^3,x)

[Out] int((c + d\*x)/cosh(a + b\*x)^3, x)

### 3.12 $\int \frac{\operatorname{sech}^3(a+bx)}{c+dx} dx$

**Optimal.** Leaf size=19

$$\text{Int}\left(\frac{\operatorname{sech}^3(a+bx)}{c+dx}, x\right)$$

[Out] Unintegrable(sech(b\*x+a)^3/(d\*x+c), x)

**Rubi [A]**

time = 0.03, antiderivative size = 0, normalized size of antiderivative = 0.00, number of steps used = 0, number of rules used = 0, integrand size = 0,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.000$ ,

Rules used = {}

$$\int \frac{\operatorname{sech}^3(a+bx)}{c+dx} dx$$

Verification is not applicable to the result.

[In] Int[Sech[a + b\*x]^3/(c + d\*x), x]

[Out] Defer[Int][Sech[a + b\*x]^3/(c + d\*x), x]

Rubi steps

$$\int \frac{\operatorname{sech}^3(a+bx)}{c+dx} dx = \int \frac{\operatorname{sech}^3(a+bx)}{c+dx} dx$$

**Mathematica [A]**

time = 65.64, size = 0, normalized size = 0.00

$$\int \frac{\operatorname{sech}^3(a+bx)}{c+dx} dx$$

Verification is not applicable to the result.

[In] Integrate[Sech[a + b\*x]^3/(c + d\*x), x]

[Out] Integrate[Sech[a + b\*x]^3/(c + d\*x), x]

**Maple [A]**

time = 180.00, size = 0, normalized size = 0.00

$$\int \frac{\operatorname{sech}(bx+a)^3}{dx+c} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `int(sech(b*x+a)^3/(d*x+c),x)`

[Out] `int(sech(b*x+a)^3/(d*x+c),x)`

**Maxima** [A]

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(sech(b*x+a)^3/(d*x+c),x, algorithm="maxima")`

[Out] 
$$\frac{((b*d*x*e^{(3*a)} + (b*c - d)*e^{(3*a)})*e^{(3*b*x)} - (b*d*x*e^a + (b*c + d)*e^a)*e^{(b*x)})}{(b^2*d^2*x^2 + 2*b^2*c*d*x + b^2*c^2 + (b^2*d^2*x^2*e^{(4*a)} + 2*b^2*c*d*x*e^{(4*a)} + b^2*c^2*e^{(4*a)})*e^{(4*b*x)} + 2*(b^2*d^2*x^2*e^{(2*a)} + 2*b^2*c*d*x*e^{(2*a)} + b^2*c^2*e^{(2*a)})*e^{(2*b*x)})} + 8*\integrate(1/8*(b^2*d^2*x^2*e^a + 2*b^2*c*d*x*e^a + (b^2*c^2 - 2*d^2)*e^a)*e^{(b*x)}/(b^2*d^3*x^3 + 3*b^2*c*d^2*x^2 + 3*b^2*c^2*d*x + b^2*c^3 + (b^2*d^3*x^3*e^{(2*a)} + 3*b^2*c*d^2*x^2*e^{(2*a)} + 3*b^2*c^2*d*x*e^{(2*a)} + b^2*c^3*e^{(2*a)})*e^{(2*b*x)}), x)$$

**Fricas** [A]

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(sech(b*x+a)^3/(d*x+c),x, algorithm="fricas")`

[Out] `integral(sech(b*x + a)^3/(d*x + c), x)`

**Sympy** [A]

time = 0.00, size = 0, normalized size = 0.00

$$\int \frac{\operatorname{sech}^3(a + bx)}{c + dx} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(sech(b*x+a)**3/(d*x+c),x)`

[Out] `Integral(sech(a + b*x)**3/(c + d*x), x)`

**Giac** [A]

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

```
[In] integrate(sech(b*x+a)^3/(d*x+c),x, algorithm="giac")
```

```
[Out] integrate(sech(b*x + a)^3/(d*x + c), x)
```

**Mupad [A]**

time = 0.00, size = -1, normalized size = -0.05

$$\int \frac{1}{\cosh(a + bx)^3 (c + dx)} dx$$

Verification of antiderivative is not currently implemented for this CAS.

```
[In] int(1/(cosh(a + b*x)^3*(c + d*x)),x)
```

```
[Out] int(1/(cosh(a + b*x)^3*(c + d*x)), x)
```

$$3.13 \quad \int \left( \frac{x}{\operatorname{sech}^{\frac{3}{2}}(x)} - \frac{1}{3}x \sqrt{\operatorname{sech}(x)} \right) dx$$

Optimal. Leaf size=24

$$-\frac{4}{9\operatorname{sech}^{\frac{3}{2}}(x)} + \frac{2x \sinh(x)}{3\sqrt{\operatorname{sech}(x)}}$$

[Out]  $-4/9/\operatorname{sech}(x)^{(3/2)}+2/3*x*\sinh(x)/\operatorname{sech}(x)^{(1/2)}$

Rubi [A]

time = 0.06, antiderivative size = 24, normalized size of antiderivative = 1.00, number of steps used = 4, number of rules used = 2, integrand size = 20,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.100$ ,

Rules used = {4272, 4274}

$$\frac{2x \sinh(x)}{3\sqrt{\operatorname{sech}(x)}} - \frac{4}{9\operatorname{sech}^{\frac{3}{2}}(x)}$$

Antiderivative was successfully verified.

[In]  $\text{Int}[x/\text{Sech}[x]^{(3/2)} - (x*\text{Sqrt}[\text{Sech}[x]])/3,x]$

[Out]  $-4/(9*\text{Sech}[x]^{(3/2)}) + (2*x*\text{Sinh}[x])/(3*\text{Sqrt}[\text{Sech}[x]])$

Rule 4272

```
Int[(csc[(e_.) + (f_.)*(x_)]*(b_.))^(n_)*((c_.) + (d_.)*(x_)), x_Symbol] :>
  Simp[d*((b*Csc[e + f*x])^n/(f^2*n^2)), x] + (Dist[(n + 1)/(b^2*n), Int[(c
+ d*x)*(b*Csc[e + f*x])^(n + 2), x], x] + Simp[(c + d*x)*Cos[e + f*x]*((b*C
sc[e + f*x])^(n + 1)/(b*f*n)), x]) /; FreeQ[{b, c, d, e, f}, x] && LtQ[n, -
1]
```

Rule 4274

```
Int[(csc[(e_.) + (f_.)*(x_)]*(b_.))^(n_)*((c_.) + (d_.)*(x_))^(m_.), x_Symb
ol] :> Dist[(b*Sin[e + f*x])^n*(b*Csc[e + f*x])^n, Int[(c + d*x)^m/(b*Sin[e
+ f*x])^n, x], x] /; FreeQ[{b, c, d, e, f, m, n}, x] && !IntegerQ[n]
```

Rubi steps

$$\begin{aligned} \int \left( \frac{x}{\operatorname{sech}^{\frac{3}{2}}(x)} - \frac{1}{3}x \sqrt{\operatorname{sech}(x)} \right) dx &= -\left( \frac{1}{3} \int x \sqrt{\operatorname{sech}(x)} dx \right) + \int \frac{x}{\operatorname{sech}^{\frac{3}{2}}(x)} dx \\ &= -\frac{4}{9\operatorname{sech}^{\frac{3}{2}}(x)} + \frac{2x \sinh(x)}{3\sqrt{\operatorname{sech}(x)}} + \frac{1}{3} \int x \sqrt{\operatorname{sech}(x)} dx - \frac{1}{3} \left( \sqrt{\cosh(x)} \sqrt{\operatorname{sech}(x)} \right) \\ &= -\frac{4}{9\operatorname{sech}^{\frac{3}{2}}(x)} + \frac{2x \sinh(x)}{3\sqrt{\operatorname{sech}(x)}} \end{aligned}$$

**Mathematica [A]**

time = 0.05, size = 17, normalized size = 0.71

$$\frac{2(-2 + 3x \tanh(x))}{9\operatorname{sech}^{\frac{3}{2}}(x)}$$

Antiderivative was successfully verified.

[In] Integrate[x/Sech[x]^(3/2) - (x\*Sqrt[Sech[x]])/3,x]

[Out] (2\*(-2 + 3\*x\*Tanh[x]))/(9\*Sech[x]^(3/2))

**Maple [F]**

time = 0.07, size = 0, normalized size = 0.00

$$\int \frac{x}{\operatorname{sech}(x)^{\frac{3}{2}}} - \frac{x \sqrt{\operatorname{sech}(x)}}{3} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int(x/sech(x)^(3/2)-1/3\*x\*sech(x)^(1/2),x)

[Out] int(x/sech(x)^(3/2)-1/3\*x\*sech(x)^(1/2),x)

**Maxima [F]**

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate(x/sech(x)^(3/2)-1/3\*x\*sech(x)^(1/2),x, algorithm="maxima")

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

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

time = 0.00, size = 0, normalized size = 0.00

Exception raised: TypeError

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate(x/sech(x)^(3/2)-1/3\*x\*sech(x)^(1/2),x, algorithm="fricas")

[Out] Exception raised: TypeError &gt;&gt; Error detected within library code: integrate: implementation incomplete (has polynomial part)

**Sympy [F]**

time = 0.00, size = 0, normalized size = 0.00

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

Verification of antiderivative is not currently implemented for this CAS.

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

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

**Giac** [F]

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate(x/sech(x)^(3/2)-1/3\*x\*sech(x)^(1/2),x, algorithm="giac")

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

**Mupad** [F]

time = 0.00, size = -1, normalized size = -0.04

$$-\int \frac{x \sqrt{\frac{1}{\cosh(x)}}}{3} - \frac{x}{\left(\frac{1}{\cosh(x)}\right)^{3/2}} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int(x/(1/cosh(x))^(3/2) - (x\*(1/cosh(x))^(1/2))/3,x)

[Out] -int((x\*(1/cosh(x))^(1/2))/3 - x/(1/cosh(x))^(3/2), x)



$$3.14 \quad \int \left( \frac{x}{\operatorname{sech}^{\frac{5}{2}}(x)} - \frac{3x}{5\sqrt{\operatorname{sech}(x)}} \right) dx$$

Optimal. Leaf size=24

$$-\frac{4}{25\operatorname{sech}^{\frac{5}{2}}(x)} + \frac{2x \sinh(x)}{5\operatorname{sech}^{\frac{3}{2}}(x)}$$

[Out]  $-4/25/\operatorname{sech}(x)^{(5/2)}+2/5*x*\sinh(x)/\operatorname{sech}(x)^{(3/2)}$

Rubi [A]

time = 0.06, antiderivative size = 24, normalized size of antiderivative = 1.00, number of steps used = 4, number of rules used = 2, integrand size = 20,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.100$ , Rules used = {4272, 4274}

$$\frac{2x \sinh(x)}{5\operatorname{sech}^{\frac{3}{2}}(x)} - \frac{4}{25\operatorname{sech}^{\frac{5}{2}}(x)}$$

Antiderivative was successfully verified.

[In]  $\text{Int}[x/\operatorname{Sech}[x]^{(5/2)} - (3*x)/(5*\text{Sqrt}[\operatorname{Sech}[x]]), x]$

[Out]  $-4/(25*\operatorname{Sech}[x]^{(5/2)}) + (2*x*\operatorname{Sinh}[x])/(5*\operatorname{Sech}[x]^{(3/2)})$

Rule 4272

$\text{Int}[(\operatorname{csc}[e_.] + (f_.)(x_.)]*(b_.))^{(n_.)}*((c_.) + (d_.)(x_.)), x\_Symbol] \rightarrow$   
 $\text{Simp}[d*((b*\operatorname{Csc}[e + f*x])^n/(f^2*n^2)), x] + (\text{Dist}[(n + 1)/(b^2*n), \text{Int}[(c + d*x)*(b*\operatorname{Csc}[e + f*x])^{(n + 2)}, x], x] + \text{Simp}[(c + d*x)*\operatorname{Cos}[e + f*x]*((b*\operatorname{Csc}[e + f*x])^{(n + 1)/(b*f*n)}), x]) /; \text{FreeQ}\{b, c, d, e, f\}, x] \&\& \text{LtQ}[n, -1]$

Rule 4274

$\text{Int}[(\operatorname{csc}[e_.] + (f_.)(x_.)]*(b_.))^{(n_.)}*((c_.) + (d_.)(x_.))^{(m_.)}, x\_Symbol] \rightarrow$   
 $\text{Dist}[(b*\operatorname{Sin}[e + f*x])^n*(b*\operatorname{Csc}[e + f*x])^n, \text{Int}[(c + d*x)^m/(b*\operatorname{Sin}[e + f*x])^n, x], x] /; \text{FreeQ}\{b, c, d, e, f, m, n\}, x] \&\& !\text{IntegerQ}[n]$

Rubi steps

$$\begin{aligned} \int \left( \frac{x}{\operatorname{sech}^{\frac{5}{2}}(x)} - \frac{3x}{5\sqrt{\operatorname{sech}(x)}} \right) dx &= -\left( \frac{3}{5} \int \frac{x}{\sqrt{\operatorname{sech}(x)}} dx \right) + \int \frac{x}{\operatorname{sech}^{\frac{5}{2}}(x)} dx \\ &= -\frac{4}{25\operatorname{sech}^{\frac{5}{2}}(x)} + \frac{2x \sinh(x)}{5\operatorname{sech}^{\frac{3}{2}}(x)} + \frac{3}{5} \int \frac{x}{\sqrt{\operatorname{sech}(x)}} dx - \frac{1}{5} \left( 3\sqrt{\cosh(x)} \sqrt{\operatorname{sech}(x)} \right) \\ &= -\frac{4}{25\operatorname{sech}^{\frac{5}{2}}(x)} + \frac{2x \sinh(x)}{5\operatorname{sech}^{\frac{3}{2}}(x)} \end{aligned}$$

**Mathematica [A]**

time = 0.07, size = 17, normalized size = 0.71

$$\frac{2(-2 + 5x \tanh(x))}{25 \operatorname{sech}^{\frac{5}{2}}(x)}$$

Antiderivative was successfully verified.

[In] Integrate[x/Sech[x]^(5/2) - (3\*x)/(5\*Sqrt[Sech[x]]),x]

[Out] (2\*(-2 + 5\*x\*Tanh[x]))/(25\*Sech[x]^(5/2))

**Maple [F]**

time = 0.07, size = 0, normalized size = 0.00

$$\int \frac{x}{\operatorname{sech}(x)^{\frac{5}{2}}} - \frac{3x}{5\sqrt{\operatorname{sech}(x)}} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] int(x/sech(x)^(5/2)-3/5\*x/sech(x)^(1/2),x)

[Out] int(x/sech(x)^(5/2)-3/5\*x/sech(x)^(1/2),x)

**Maxima [F]**

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate(x/sech(x)^(5/2)-3/5\*x/sech(x)^(1/2),x, algorithm="maxima")

[Out] integrate(-3/5\*x/sqrt(sech(x)) + x/sech(x)^(5/2), x)

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

time = 0.00, size = 0, normalized size = 0.00

Exception raised: TypeError

Verification of antiderivative is not currently implemented for this CAS.

[In] integrate(x/sech(x)^(5/2)-3/5\*x/sech(x)^(1/2),x, algorithm="fricas")

[Out] Exception raised: TypeError &gt;&gt; Error detected within library code: integrate: implementation incomplete (has polynomial part)

**Sympy [F]**

time = 0.00, size = 0, normalized size = 0.00

$$\frac{\int \left( -\frac{5x}{\operatorname{sech}^{\frac{5}{2}}(x)} \right) dx + \int \frac{3x}{\sqrt{\operatorname{sech}(x)}} dx}{5}$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(x/sech(x)**(5/2)-3/5*x/sech(x)**(1/2),x)`

[Out] `-(Integral(-5*x/sech(x)**(5/2), x) + Integral(3*x/sqrt(sech(x)), x))/5`

**Giac** [F]

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(x/sech(x)^(5/2)-3/5*x/sech(x)^(1/2),x, algorithm="giac")`

[Out] `integrate(-3/5*x/sqrt(sech(x)) + x/sech(x)^(5/2), x)`

**Mupad** [F]

time = 0.00, size = -1, normalized size = -0.04

$$-\int \frac{3x}{5\sqrt{\frac{1}{\cosh(x)}}} - \frac{x}{\left(\frac{1}{\cosh(x)}\right)^{5/2}} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `int(x/(1/cosh(x))^(5/2) - (3*x)/(5*(1/cosh(x))^(1/2)),x)`

[Out] `-int((3*x)/(5*(1/cosh(x))^(1/2)) - x/(1/cosh(x))^(5/2), x)`

$$3.15 \quad \int \left( \frac{x}{\operatorname{sech}^{\frac{7}{2}}(x)} - \frac{5}{21} x \sqrt{\operatorname{sech}(x)} \right) dx$$

Optimal. Leaf size=47

$$-\frac{4}{49\operatorname{sech}^{\frac{7}{2}}(x)} - \frac{20}{63\operatorname{sech}^{\frac{3}{2}}(x)} + \frac{2x \sinh(x)}{7\operatorname{sech}^{\frac{5}{2}}(x)} + \frac{10x \sinh(x)}{21\sqrt{\operatorname{sech}(x)}}$$

[Out] -4/49/sech(x)^(7/2)-20/63/sech(x)^(3/2)+2/7\*x\*sinh(x)/sech(x)^(5/2)+10/21\*x\*sinh(x)/sech(x)^(1/2)

Rubi [A]

time = 0.07, antiderivative size = 47, normalized size of antiderivative = 1.00, number of steps used = 5, number of rules used = 2, integrand size = 20,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.100$ , Rules used = {4272, 4274}

$$-\frac{20}{63\operatorname{sech}^{\frac{3}{2}}(x)} - \frac{4}{49\operatorname{sech}^{\frac{7}{2}}(x)} + \frac{2x \sinh(x)}{7\operatorname{sech}^{\frac{5}{2}}(x)} + \frac{10x \sinh(x)}{21\sqrt{\operatorname{sech}(x)}}$$

Antiderivative was successfully verified.

[In] Int[x/Sech[x]^(7/2) - (5\*x\*Sqrt[Sech[x]])/21,x]

[Out] -4/(49\*Sech[x]^(7/2)) - 20/(63\*Sech[x]^(3/2)) + (2\*x\*Sinh[x])/(7\*Sech[x]^(5/2)) + (10\*x\*Sinh[x])/(21\*Sqrt[Sech[x]])

Rule 4272

```
Int[(csc[(e_.) + (f_.)*(x_)]*(b_.))^(n_)*((c_.) + (d_.)*(x_)), x_Symbol] :>
  Simp[d*((b*Csc[e + f*x])^n/(f^2*n^2)), x] + (Dist[(n + 1)/(b^2*n), Int[(c
+ d*x)*(b*Csc[e + f*x])^(n + 2), x], x] + Simp[(c + d*x)*Cos[e + f*x]*((b*C
sc[e + f*x])^(n + 1)/(b*f*n)), x]) /; FreeQ[{b, c, d, e, f}, x] && LtQ[n, -
1]
```

Rule 4274

```
Int[(csc[(e_.) + (f_.)*(x_)]*(b_.))^(n_)*((c_.) + (d_.)*(x_))^(m_.), x_Symb
ol] :> Dist[(b*Sin[e + f*x])^n*(b*Csc[e + f*x])^n, Int[(c + d*x)^m/(b*Sin[e
+ f*x])^n, x], x] /; FreeQ[{b, c, d, e, f, m, n}, x] && !IntegerQ[n]
```

Rubi steps

$$\begin{aligned}
\int \left( \frac{x}{\operatorname{sech}^{\frac{7}{2}}(x)} - \frac{5}{21} x \sqrt{\operatorname{sech}(x)} \right) dx &= - \left( \frac{5}{21} \int x \sqrt{\operatorname{sech}(x)} dx \right) + \int \frac{x}{\operatorname{sech}^{\frac{7}{2}}(x)} dx \\
&= -\frac{4}{49 \operatorname{sech}^{\frac{7}{2}}(x)} + \frac{2x \sinh(x)}{7 \operatorname{sech}^{\frac{5}{2}}(x)} + \frac{5}{7} \int \frac{x}{\operatorname{sech}^{\frac{3}{2}}(x)} dx - \frac{1}{21} \left( 5 \sqrt{\cosh(x)} \right) \\
&= -\frac{4}{49 \operatorname{sech}^{\frac{7}{2}}(x)} - \frac{20}{63 \operatorname{sech}^{\frac{3}{2}}(x)} + \frac{2x \sinh(x)}{7 \operatorname{sech}^{\frac{5}{2}}(x)} + \frac{10x \sinh(x)}{21 \sqrt{\operatorname{sech}(x)}} + \frac{5}{21} \int \\
&= -\frac{4}{49 \operatorname{sech}^{\frac{7}{2}}(x)} - \frac{20}{63 \operatorname{sech}^{\frac{3}{2}}(x)} + \frac{2x \sinh(x)}{7 \operatorname{sech}^{\frac{5}{2}}(x)} + \frac{10x \sinh(x)}{21 \sqrt{\operatorname{sech}(x)}}
\end{aligned}$$

**Mathematica [A]**

time = 0.07, size = 45, normalized size = 0.96

$$\sqrt{\operatorname{sech}(x)} \left( -\frac{167}{882} - \frac{88}{441} \cosh(2x) - \frac{1}{98} \cosh(4x) + \frac{13}{42} x \sinh(2x) + \frac{1}{28} x \sinh(4x) \right)$$

Antiderivative was successfully verified.

`[In] Integrate[x/Sech[x]^(7/2) - (5*x*Sqrt[Sech[x]])/21, x]``[Out] Sqrt[Sech[x]]*(-167/882 - (88*Cosh[2*x])/441 - Cosh[4*x]/98 + (13*x*Sinh[2*x])/42 + (x*Sinh[4*x])/28)`**Maple [F]**

time = 0.07, size = 0, normalized size = 0.00

$$\int \frac{x}{\operatorname{sech}(x)^{\frac{7}{2}}} - \frac{5x \sqrt{\operatorname{sech}(x)}}{21} dx$$

Verification of antiderivative is not currently implemented for this CAS.

`[In] int(x/sech(x)^(7/2)-5/21*x*sech(x)^(1/2), x)``[Out] int(x/sech(x)^(7/2)-5/21*x*sech(x)^(1/2), x)`**Maxima [F]**

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

`[In] integrate(x/sech(x)^(7/2)-5/21*x*sech(x)^(1/2), x, algorithm="maxima")``[Out] integrate(-5/21*x*sqrt(sech(x)) + x/sech(x)^(7/2), x)`

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

time = 0.00, size = 0, normalized size = 0.00

Exception raised: TypeError

Verification of antiderivative is not currently implemented for this CAS.

```
[In] integrate(x/sech(x)^(7/2)-5/21*x*sech(x)^(1/2),x, algorithm="fricas")
```

```
[Out] Exception raised: TypeError >> Error detected within library code: integ
rate: implementation incomplete (has polynomial part)
```

**Sympy [F]**

time = 0.00, size = 0, normalized size = 0.00

$$\frac{\int \left( -\frac{21x}{\operatorname{sech}^{\frac{7}{2}}(x)} \right) dx + \int 5x \sqrt{\operatorname{sech}(x)} dx}{21}$$

Verification of antiderivative is not currently implemented for this CAS.

```
[In] integrate(x/sech(x)**(7/2)-5/21*x*sech(x)**(1/2),x)
```

```
[Out] -(Integral(-21*x/sech(x)**(7/2), x) + Integral(5*x*sqrt(sech(x)), x))/21
```

**Giac [F]**

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

```
[In] integrate(x/sech(x)^(7/2)-5/21*x*sech(x)^(1/2),x, algorithm="giac")
```

```
[Out] integrate(-5/21*x*sqrt(sech(x)) + x/sech(x)^(7/2), x)
```

**Mupad [F]**

time = 0.00, size = -1, normalized size = -0.02

$$-\int \frac{5x \sqrt{\frac{1}{\cosh(x)}}}{21} - \frac{x}{\left(\frac{1}{\cosh(x)}\right)^{7/2}} dx$$

Verification of antiderivative is not currently implemented for this CAS.

```
[In] int(x/(1/cosh(x))^(7/2) - (5*x*(1/cosh(x))^(1/2))/21,x)
```

```
[Out] -int((5*x*(1/cosh(x))^(1/2))/21 - x/(1/cosh(x))^(7/2), x)
```

$$3.16 \quad \int \left( \frac{x^2}{\operatorname{sech}^{\frac{3}{2}}(x)} - \frac{1}{3}x^2 \sqrt{\operatorname{sech}(x)} \right) dx$$

**Optimal.** Leaf size=66

$$-\frac{8x}{9\operatorname{sech}^{\frac{3}{2}}(x)} - \frac{16}{27}i\sqrt{\cosh(x)} F\left(\frac{ix}{2} \middle| 2\right) \sqrt{\operatorname{sech}(x)} + \frac{16\sinh(x)}{27\sqrt{\operatorname{sech}(x)}} + \frac{2x^2\sinh(x)}{3\sqrt{\operatorname{sech}(x)}}$$

[Out]  $-8/9*x/\operatorname{sech}(x)^{(3/2)}+16/27*\sinh(x)/\operatorname{sech}(x)^{(1/2)}+2/3*x^2*\sinh(x)/\operatorname{sech}(x)^{(1/2)}-16/27*I*(\cosh(1/2*x)^2)^{(1/2)}/\cosh(1/2*x)*\operatorname{EllipticF}(I*\sinh(1/2*x),2^{(1/2)})*\cosh(x)^{(1/2)}*\operatorname{sech}(x)^{(1/2)}$

**Rubi [A]**

time = 0.12, antiderivative size = 66, normalized size of antiderivative = 1.00, number of steps used = 7, number of rules used = 5, integrand size = 24,  $\frac{\text{number of rules}}{\text{integrand size}} = 0.208$ , Rules used = {4273, 4274, 3854, 3856, 2720}

$$\frac{2x^2\sinh(x)}{3\sqrt{\operatorname{sech}(x)}} - \frac{8x}{9\operatorname{sech}^{\frac{3}{2}}(x)} + \frac{16\sinh(x)}{27\sqrt{\operatorname{sech}(x)}} - \frac{16}{27}i\sqrt{\cosh(x)} \sqrt{\operatorname{sech}(x)} F\left(\frac{ix}{2} \middle| 2\right)$$

Antiderivative was successfully verified.

[In]  $\operatorname{Int}[x^2/\operatorname{Sech}[x]^{(3/2)} - (x^2*\operatorname{Sqrt}[\operatorname{Sech}[x]])/3, x]$

[Out]  $(-8*x)/(9*\operatorname{Sech}[x]^{(3/2)}) - ((16*I)/27)*\operatorname{Sqrt}[\operatorname{Cosh}[x]]*\operatorname{EllipticF}[(I/2)*x, 2]*\operatorname{Sqrt}[\operatorname{Sech}[x]] + (16*\operatorname{Sinh}[x])/(27*\operatorname{Sqrt}[\operatorname{Sech}[x]]) + (2*x^2*\operatorname{Sinh}[x])/(3*\operatorname{Sqrt}[\operatorname{Sech}[x]])$

Rule 2720

$\operatorname{Int}[1/\operatorname{Sqrt}[\sin[(c_.) + (d_.)*(x_)]], x\_Symbol] \rightarrow \operatorname{Simp}[(2/d)*\operatorname{EllipticF}[(1/2)*(c - \operatorname{Pi}/2 + d*x), 2], x] /; \operatorname{FreeQ}\{c, d\}, x]$

Rule 3854

$\operatorname{Int}[(\operatorname{csc}[(c_.) + (d_.)*(x_)]*(b_.)^{(n_)}), x\_Symbol] \rightarrow \operatorname{Simp}[\operatorname{Cos}[c + d*x]*((b*\operatorname{Csc}[c + d*x])^{(n+1)}/(b*d^n)), x] + \operatorname{Dist}[(n+1)/(b^2*n), \operatorname{Int}[(b*\operatorname{Csc}[c + d*x])^{(n+2)}, x], x] /; \operatorname{FreeQ}\{b, c, d\}, x] \&\& \operatorname{LtQ}[n, -1] \&\& \operatorname{IntegerQ}[2*n]$

Rule 3856

$\operatorname{Int}[(\operatorname{csc}[(c_.) + (d_.)*(x_)]*(b_.)^{(n_)}), x\_Symbol] \rightarrow \operatorname{Dist}[(b*\operatorname{Csc}[c + d*x])^n*\operatorname{Sin}[c + d*x]^n, \operatorname{Int}[1/\operatorname{Sin}[c + d*x]^n, x], x] /; \operatorname{FreeQ}\{b, c, d\}, x] \&\& \operatorname{EqQ}[n^2, 1/4]$

Rule 4273

```
Int[(csc[(e_.) + (f_.)*(x_)]*(b_.))^(n_)*((c_.) + (d_.)*(x_))^(m_), x_Symbol]
:= Simp[d*m*(c + d*x)^(m - 1)*((b*Csc[e + f*x])^n/(f^2*n^2)), x] + (Dist
[(n + 1)/(b^2*n), Int[(c + d*x)^m*(b*Csc[e + f*x])^(n + 2), x], x] - Dist[d
^2*m*((m - 1)/(f^2*n^2)), Int[(c + d*x)^(m - 2)*(b*Csc[e + f*x])^n, x], x]
+ Simp[(c + d*x)^m*Cos[e + f*x]*((b*Csc[e + f*x])^(n + 1)/(b*f*n)), x]) /;
FreeQ[{b, c, d, e, f}, x] && LtQ[n, -1] && GtQ[m, 1]
```

Rule 4274

```
Int[(csc[(e_.) + (f_.)*(x_)]*(b_.))^(n_)*((c_.) + (d_.)*(x_))^(m_.), x_Symbol]
:= Dist[(b*Sine[e + f*x])^n*(b*Csc[e + f*x])^n, Int[(c + d*x)^m/(b*Sine[e
+ f*x])^n, x], x] /; FreeQ[{b, c, d, e, f, m, n}, x] && !IntegerQ[n]
```

Rubi steps

$$\begin{aligned}
\int \left( \frac{x^2}{\operatorname{sech}^{\frac{3}{2}}(x)} - \frac{1}{3}x^2 \sqrt{\operatorname{sech}(x)} \right) dx &= - \left( \frac{1}{3} \int x^2 \sqrt{\operatorname{sech}(x)} dx \right) + \int \frac{x^2}{\operatorname{sech}^{\frac{3}{2}}(x)} dx \\
&= -\frac{8x}{9\operatorname{sech}^{\frac{3}{2}}(x)} + \frac{2x^2 \sinh(x)}{3\sqrt{\operatorname{sech}(x)}} + \frac{1}{3} \int x^2 \sqrt{\operatorname{sech}(x)} dx + \frac{8}{9} \int \frac{1}{\operatorname{sech}^{\frac{3}{2}}(x)} dx \\
&= -\frac{8x}{9\operatorname{sech}^{\frac{3}{2}}(x)} + \frac{16 \sinh(x)}{27\sqrt{\operatorname{sech}(x)}} + \frac{2x^2 \sinh(x)}{3\sqrt{\operatorname{sech}(x)}} + \frac{8}{27} \int \sqrt{\operatorname{sech}(x)} dx \\
&= -\frac{8x}{9\operatorname{sech}^{\frac{3}{2}}(x)} + \frac{16 \sinh(x)}{27\sqrt{\operatorname{sech}(x)}} + \frac{2x^2 \sinh(x)}{3\sqrt{\operatorname{sech}(x)}} + \frac{1}{27} \left( 8\sqrt{\cosh(x)} \sqrt{\operatorname{sech}(x)} \right. \\
&= -\frac{8x}{9\operatorname{sech}^{\frac{3}{2}}(x)} - \frac{16}{27} i \sqrt{\cosh(x)} F\left(\frac{ix}{2} \middle| 2\right) \sqrt{\operatorname{sech}(x)} + \frac{16 \sinh(x)}{27\sqrt{\operatorname{sech}(x)}}
\end{aligned}$$

Mathematica [A]

time = 0.05, size = 55, normalized size = 0.83

$$\frac{1}{27} \sqrt{\operatorname{sech}(x)} \left( -12x - 12x \cosh(2x) - 16i \sqrt{\cosh(x)} F\left(\frac{ix}{2} \middle| 2\right) + 8 \sinh(2x) + 9x^2 \sinh(2x) \right)$$

Antiderivative was successfully verified.

```
[In] Integrate[x^2/Sech[x]^(3/2) - (x^2*Sqrt[Sech[x]])/3,x]
```

```
[Out] (Sqrt[Sech[x]]*(-12*x - 12*x*Cosh[2*x] - (16*I)*Sqrt[Cosh[x]]*EllipticF[(I/2)*x, 2] + 8*Sinh[2*x] + 9*x^2*Sinh[2*x]))/27
```



**Maple [F]**

time = 0.06, size = 0, normalized size = 0.00

$$\int \frac{x^2}{\operatorname{sech}(x)^{\frac{3}{2}}} - \frac{x^2 \sqrt{\operatorname{sech}(x)}}{3} dx$$

Verification of antiderivative is not currently implemented for this CAS.

`[In] int(x^2/sech(x)^(3/2)-1/3*x^2*sech(x)^(1/2),x)``[Out] int(x^2/sech(x)^(3/2)-1/3*x^2*sech(x)^(1/2),x)`**Maxima [F]**

time = 0.00, size = 0, normalized size = 0.00

Failed to integrate

Verification of antiderivative is not currently implemented for this CAS.

`[In] integrate(x^2/sech(x)^(3/2)-1/3*x^2*sech(x)^(1/2),x, algorithm="maxima")``[Out] integrate(-1/3*x^2*sqrt(sech(x)) + x^2/sech(x)^(3/2), x)`**Fricas [F(-2)]**

time = 0.00, size = 0, normalized size = 0.00

Exception raised: TypeError

Verification of antiderivative is not currently implemented for this CAS.

`[In] integrate(x^2/sech(x)^(3/2)-1/3*x^2*sech(x)^(1/2),x, algorithm="fricas")``[Out] Exception raised: TypeError >> Error detected within library code: integrate: implementation incomplete (has polynomial part)`**Sympy [F]**

time = 0.00, size = 0, normalized size = 0.00

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

Verification of antiderivative is not currently implemented for this CAS.

`[In] integrate(x**2/sech(x)**(3/2)-1/3*x**2*sech(x)**(1/2),x)``[Out] -(Integral(-3*x**2/sech(x)**(3/2), x) + Integral(x**2*sqrt(sech(x)), x))/3`**Giac [F]**

time = 0.00, size = 0, normalized size = 0.00

could not integrate

Verification of antiderivative is not currently implemented for this CAS.

[In] `integrate(x^2/sech(x)^(3/2)-1/3*x^2*sech(x)^(1/2),x, algorithm="giac")`

[Out] `integrate(-1/3*x^2*sqrt(sech(x)) + x^2/sech(x)^(3/2), x)`

**Mupad [F]**

time = 0.00, size = -1, normalized size = -0.02

$$-\int \frac{x^2 \sqrt{\frac{1}{\cosh(x)}}}{3} - \frac{x^2}{\left(\frac{1}{\cosh(x)}\right)^{3/2}} dx$$

Verification of antiderivative is not currently implemented for this CAS.

[In] `int(x^2/(1/cosh(x))^(3/2) - (x^2*(1/cosh(x))^(1/2))/3,x)`

[Out] `-int((x^2*(1/cosh(x))^(1/2))/3 - x^2/(1/cosh(x))^(3/2), x)`

# Chapter 4

## Appendix

### Local contents

4.1	Download section . . . . .	92
4.2	Listing of Grading functions . . . . .	92

## 4.1 Download section

The following zip files contain the raw integrals used in this test.

**Mathematica format** Mathematica\_syntax.zip

**Maple and Mupad format** Maple\_syntax.zip

**Sympy format** SYMPY\_syntax.zip

**Sage math format** SAGE\_syntax.zip

## 4.2 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.

### 4.2.1 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,leafC
  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","none"}
          ,(*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","none"}
        ,(*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*)

```



```

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]

```

## 4.2.2 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 issues";
      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 r
                    convert(leaf_count_result,string)," vs. $2 (" ,
                    convert(leaf_count_optimal,string)," ) = ",convert(2*leaf_co
            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 o
                        convert(leaf_count_result,string)," $ vs. $2(",
                        convert(leaf_count_optimal,string),")=",convert(2*leaf_cou

    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:

```

### 4.2.3 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,
        fresnelc,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)),ExpnT
    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 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 = ""
                else:
                    grade = "B"
                    grade_annotation = "Leaf count of result is larger than twice the leaf count of optimal. "+str(leaf_count_result)
            else:
                grade = "C"
                grade_annotation = "Result contains higher order function than in optimal. Order "+str(ExpnType_result)

```

```
#print("Before returning. grade=",grade, " grade_annotation=",grade_annotation)

return grade, grade_annotation
```

#### 4.2.4 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: #instance(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','arccsch','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']

    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: #isinstance(expn,list):
        return max(map(expnType, expn)) #apply(max,map(ExpnType,expn))
    elif is_sqrt(expn):
        if type(expn.operands()[0])==Rational: #type(isinstance(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: #isinstance(expn,Pow)
        if type(expn.operands()[1])==Integer: #isinstance(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.op
elif expn.operator() == add_vararg or expn.operator() == mul_vararg: #isinstance(expn,Add) or instan
    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 t
        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_
else:
    grade = "C"
    grade_annotation = "Result contains higher order function than in optimal. Order "+str(expnType_resu

print("Before returning. grade=",grade, " grade_annotation=",grade_annotation)

return grade, grade_annotation

```