In[ ]:=
```
SetDirectory[NotebookDirectory[]];
<< AdditiveDecomposition.m
```

AdditiveDecomposition.m is written by Hao Du and Elaine Wong, Austrian Academy of Sciences, RICAM, Version 0.0 (Feb. 5, 2020)
⟶ Type ?AdditiveDecomposition for help

In[ ]:=
```
? AdditiveDecomposition
```

Welcome to AdditiveDecomposition.m. This package accompanies the paper **An Additive Decomposition in S−Primitive Towers** by Hao Du, Jing Guo, Ziming Li and Elaine Wong.

The main objective of this package is an implementation of the algorithm for decomposing a function in an S−primitive tower into its integrable part and a remainder that is minimal in some sense,
both of which are in the same field. If the tower is also logarithmic, then we are able to determine an extension field such that the decomposition has a finer remainder. A function in an S−primitive
tower is integrable in the tower if and only if the remainder is equal to zero.

The following commands serve the above objectives: AddDecompInField and WellGenLogTower.

Common subtasks in the above algorithm are Hermite reduction and computing the matryoshka decomposition, and these are also provided in this implementation via HermiteReduceGeneral and
ProperDecomposition, respectively. Please refer to the example notebook for usage examples.

Some other functions that might be useful: ExtendedEuclidean, HeadTerm, MonomialIndicator, AddDecompLogTower. Please refer to the example notebook for usage examples.

Part 1: Examples from the Paper

Example 4.9

In[ ]:=
```
f = 1/(Log[x] LogIntegral[x]) + (-2 x Log[x] + LogIntegral[x])/Log[x]^2 + Log[Log[x]];
Print["f = ", f];
Print["Generators and derivatives:"];
genlog = {Log[x], LogIntegral[x], Log[Log[x]]}
derlog = D[#, x] & /@ genlog // Together
Print["Renaming generators and substituting into f:"];
genlist = Table[Symbol["t" <> ToString[i]], {i, Length[genlog]}]
subs = Thread[genlog → genlist];
derlist = derlog //. subs
fnew = f //. subs;
Print["f = ", fnew];
Print["Additive decomposition {integrable part, remainder}:"];
adif = AddDecompInField[fnew, x, genlist, derlist]
Print["Remainder in the base field:"];
subt = GenSubstitute[x, genlist, derlist];
r = (adif[[2]]) //. subt
```

f = $\text{Log}[\text{Log}[x]] + \dfrac{1}{\text{Log}[x]\ \text{LogIntegral}[x]} + \dfrac{-2\,x\,\text{Log}[x] + \text{LogIntegral}[x]}{\text{Log}[x]^2}$

Generators and derivatives:

Out[ ]= {Log[x], LogIntegral[x], Log[Log[x]]}

Out[ ]= $\left\{\dfrac{1}{x}, \dfrac{1}{\text{Log}[x]}, \dfrac{1}{x\,\text{Log}[x]}\right\}$

Renaming generators and substituting into f:

Out[ ]= {t1, t2, t3}

Out[ ]= $\left\{\dfrac{1}{x}, \dfrac{1}{t1}, \dfrac{1}{t1\,x}\right\}$

f = $\dfrac{1}{t1\,t2} + t3 + \dfrac{t2 - 2\,t1\,x}{t1^2}$

Additive decomposition {integrable part, remainder}:

Out[ ]= $\left\{-t2 + \dfrac{t2^2}{2} - \dfrac{t2\,x}{t1} + t3\,x - \dfrac{x^2}{t1}, \dfrac{1}{t1\,t2}\right\}$

Remainder in the base field:

Out[ ]= $\dfrac{1}{\text{Log}[x]\ \text{LogIntegral}[x]}$

Example 5.1

In[∘]:=
```
f = Log[(1 + x) Log[x]] / (x Log[x]);
Print["f = ", f];
```

$$f = \frac{\text{Log}[(1+x)\,\text{Log}[x]]}{x\,\text{Log}[x]}$$

In[∘]:=
```
Print["Generators and derivatives for Tower (i):"];
genlog = {Log[x], Log[(x + 1) Log[x]]}
derlog = D[#, x] & /@ genlog // Together
Print["Renaming generators and substituting into f:"];
genlist = Table[Symbol["t" <> ToString[i]], {i, Length[genlog]}]
subs = Thread[genlog → genlist];
derlist = derlog //. subs
fnew = f //. subs;
Print["f = ", fnew];
Print["Additive decomposition {integrable part, remainder}:"];
adif = AddDecompInField[fnew, x, genlist, derlist]
Print["Remainder in the base field:"];
subt = GenSubstitute[x, genlist, derlist];
r = (adif[[2]]) //. subt
```

Generators and derivatives for Tower (i):

Out[∘]= {Log[x], Log[(1 + x) Log[x]]}

Out[∘]= $\left\{\frac{1}{x}, \frac{1+x+x\,\text{Log}[x]}{x\,(1+x)\,\text{Log}[x]}\right\}$

Renaming generators and substituting into f:

Out[∘]= {t1, t2}

Out[∘]= $\left\{\frac{1}{x}, \frac{1+x+t1\,x}{t1\,x\,(1+x)}\right\}$

$$f = \frac{t2}{t1\,x}$$

Additive decomposition {integrable part, remainder}:

Out[∘]= $\left\{0, \frac{t2}{t1\,x}\right\}$

Remainder in the base field:

Out[∘]= $\frac{\text{Log}[1+x]+\text{Log}[\text{Log}[x]]}{x\,\text{Log}[x]}$

In[∘]:=
```
f = (Log[1 + x] + Log[Log[x]])/(x Log[x]);
Print["f = ", f];
```

$$f = \frac{\text{Log}[1+x] + \text{Log}[\text{Log}[x]]}{x\,\text{Log}[x]}$$

In[∘]:=
```
Print["Generators and derivatives for Tower (ii):"];
genlog = {Log[x], Log[x + 1], Log[Log[x]]}
derlog = D[#, x] & /@ genlog // Together
Print["Renaming generators and substituting into f:"];
genlist = Table[Symbol["u" <> ToString[i]], {i, Length[genlog]}]
subs = Thread[genlog → genlist];
derlist = derlog //. subs
fnew = f //. subs;
Print["f = ", fnew];
Print["Additive decomposition {integrable part, remainder}:"];
adif = AddDecompInField[fnew, x, genlist, derlist]
Print["Remainder in the base field:"];
subt = GenSubstitute[x, genlist, derlist];
r = (adif[[2]]) //. subt
```

Generators and derivatives for Tower (ii):

Out[∘]= $\{\text{Log}[x], \text{Log}[1+x], \text{Log}[\text{Log}[x]]\}$

Out[∘]= $\left\{\frac{1}{x}, \frac{1}{1+x}, \frac{1}{x\,\text{Log}[x]}\right\}$

Renaming generators and substituting into f:

Out[∘]= $\{u1, u2, u3\}$

Out[∘]= $\left\{\frac{1}{x}, \frac{1}{1+x}, \frac{1}{u1\,x}\right\}$

$$f = \frac{u2 + u3}{u1\,x}$$

Additive decomposition {integrable part, remainder}:

Out[∘]= $\left\{\frac{u3^2}{2}, \frac{u2}{u1\,x}\right\}$

Remainder in the base field:

Out[∘]= $\frac{\text{Log}[1+x]}{x\,\text{Log}[x]}$

Example 5.7

In[•]:=
```
Print["Generators and derivatives:"];
genlog = {Log[x], Log[x Log[x]], Log[(x + 1) (Log[x] + 1) Log[x Log[x]]]};
derlog = D[#, x] & /@ genlog // Together
Print["Renaming generators:"];
genlist = Table[Symbol["t" <> ToString[i]], {i, Length[genlog]}]
subs = Thread[genlog → genlist];
derlist = derlog //. subs
Print["Associated matrix with respect to the matryoshka decomposition:"];
AssMat[genlist, derlist]
Print["We embed the previous tower into a well-generated one:"];
wglt = WellGenLogTower[x, genlist, derlist];
newgenlist = wglt[[1]];
newderlist = wglt[[3]];
subsu = Thread[genlist → newgenlist.wglt[[2]]]
wglt[[4]] // MatrixForm
```

Generators and derivatives:

Out[•]= $$\left\{\frac{1}{x}, \frac{1+\text{Log}[x]}{x\,\text{Log}[x]}, \left(1+x+2\,\text{Log}[x]+2\,x\,\text{Log}[x]+\text{Log}[x]^2+x\,\text{Log}[x]^2+\text{Log}[x]\,\text{Log}[x\,\text{Log}[x]]+2\,x\,\text{Log}[x]\,\text{Log}[x\,\text{Log}[x]]+x\,\text{Log}[x]^2\,\text{Log}[x\,\text{Log}[x]]\right) \Big/ \left(x\,(1+x)\,\text{Log}[x]\,\left(1+\text{Log}[x]\right)\,\text{Log}[x\,\text{Log}[x]]\right)\right\}$$

Renaming generators:

Out[•]= $\{t1, t2, t3\}$

Out[•]= $$\left\{\frac{1}{x}, \frac{1+t1}{t1\,x}, \left(1+2\,t1+t1^2+t1\,t2+x+2\,t1\,x+t1^2\,x+2\,t1\,t2\,x+t1^2\,t2\,x\right) \Big/ \left(t1\,(1+t1)\,t2\,x\,(1+x)\right)\right\}$$

Associated matrix with respect to the matryoshka decomposition:

Out[•]//MatrixForm=
$$\begin{pmatrix} \frac{1}{x} & \frac{1}{x} & \frac{1}{1+x} \\ 0 & \frac{1}{t1\,x} & \frac{1}{(1+t1)\,x} \\ 0 & 0 & \frac{1+t1}{t1\,t2\,x} \end{pmatrix}$$

We embed the previous tower into a well-generated one:

Out[•]= $\{t1 \to u1, t2 \to u1 + u3, t3 \to u2 + u4 + u5\}$

Out[∘]//MatrixForm=

$$\begin{pmatrix} \frac{1}{x} & \frac{1}{1+x} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{u1\,x} & \frac{1}{(1+u1)\,x} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1+u1}{u1\,(u1+u3)\,x} \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

In[∘]:=

$$f1 = \frac{(1+t1)^2 + t1\,t2}{t1\,(1+t1)\,t2\,x};\ f2 = \frac{t3}{x};$$

```
Print["f1 = ", f1, " and f2 = ", f2];
phif1 = f1 /. subsu; phif2 = f2 /. subsu;
Print["ϕ(f1) = ", phif1, " and ϕ(f2) = ", phif2];
Print["Additive decompositions for all f1, ϕ(f1), f2, ϕ(f2), respectively:"];
adif1 = AddDecompInField[f1, x, genlist, derlist]
adwg1 = AddDecompInField[phif1, x, newgenlist, newderlist]
adif2 = AddDecompInField[f2, x, genlist, derlist]
adwg2 = AddDecompInField[phif2, x, newgenlist, newderlist]
Print["Projections of the remainders of f2 and ϕ(f2), respectively:"];
ProperDecomposition[adif2[[2]], genlist]
ProperDecomposition[adwg2[[2]], newgenlist]
```

$f1 = \frac{(1+t1)^2 + t1\,t2}{t1\,(1+t1)\,t2\,x}$ and $f2 = \frac{t3}{x}$

$\phi(f1) = \frac{(1+u1)^2 + u1\,(u1+u3)}{u1\,(1+u1)\,(u1+u3)\,x}$ and $\phi(f2) = \frac{u2+u4+u5}{x}$

Additive decompositions for all f1, ϕ(f1), f2, ϕ(f2), respectively:

Out[∘]= $\left\{0,\ \frac{1+2\,t1+t1^2+t1\,t2}{t1\,(1+t1)\,t2\,x}\right\}$

Out[∘]= $\{u4+u5,\ 0\}$

Out[∘]= $\left\{-t1+t1\,t3,\ \left(-1-2\,t1-t1^2+t2-x-2\,t1\,x-t1^2\,x+t2\,x-t1\,t2\,x-t1^2\,t2\,x\right) \big/ \left((1+t1)\,t2\,x\,(1+x)\right)\right\}$

Out[∘]= $\left\{-u1+u1\,u2+u4+u1\,u4+u1\,u5,\ \frac{-1-u1-x-u1\,x-u1^2\,x-u1\,u3\,x}{(u1+u3)\,x\,(1+x)}\right\}$

Projections of the remainders of f2 and ϕ(f2), respectively:

Out[∘]= $\left\{\frac{t1}{-1-x},\ \frac{1}{(1+t1)\,x},\ \frac{-1-t1}{t2\,x},\ 0\right\}$

Out[∘]= $\left\{\frac{u1}{-1-x},\ 0,\ 0,\ \frac{-1-u1}{(u1+u3)\,x},\ 0,\ 0\right\}$

Part 2: Illustration of the Main Functions

AddDecompInField

In[∘]:= `? AddDecompInField`

AddDecompInField[f, var, gen, der] takes a function in an S–primitive tower and outputs the pair {integrable part of f, minimal remainder (0, if f is integrable)} from the same tower.

In[∘]:= `AddDecompInField[t4, x, {t1, t2, t3, t4}, {1 / x, 1 / t1, 1 / t2, 1 / t3}]`

Out[∘]= $\left\{t4\,x,\ -\frac{x}{t3}\right\}$

In[∘]:= `AddDecompInField[t4, x, {t1, t2, t3, t4},`
$\left\{\frac{1}{x}, \frac{1}{1+x}, \frac{1}{t1}, \left(-t1\,t2 - 2\,t1\,t3 + 2\,x + t1\,x - t1\,t2\,x - 2\,t1\,t3\,x + 2\,x^2 + 2\,t3\,x^2 + 2\,t3\,x^3\right) \big/ \left(t1\,x\,(1+x)\,\left(t2 + 2\,t3 + t3^2\,x\right)\right)\right\}]$

Out[∘]= $\left\{t4\,x,\ \left(t1\,t2 + 2\,t1\,t3 - 2\,x - t1\,x + t1\,t2\,x + 2\,t1\,t3\,x - 2\,x^2 - 2\,t3\,x^2 - 2\,t3\,x^3\right) \big/ \left(t1\,(1+x)\,\left(t2 + 2\,t3 + t3^2\,x\right)\right)\right\}$

In[∘]:= $\text{AddDecompInField}\left[t2 + t2\,t3 + \frac{t2 - 2\,t1\,x}{t1^2},\ x,\ \{t1, t2, t3\},\ \left\{\frac{1}{x}, \frac{1}{t1\,x}, \frac{1}{1+x}\right\}\right]$

Out[∘]= $\left\{-\frac{x}{t1} - \frac{t2\,x}{t1} + t2\,t3\,x,\ \frac{1 + t2 + t1\,t2 - t3 - x + t2\,x - t3\,x - 2\,x^2}{t1\,(1+x)}\right\}$

WellGenLogTower

In[∘]:= `? WellGenLogTower`

WellGenLogTower[var, gen, der] takes input from a primitive tower and outputs {new generators, the transfer matrix between old and new generators, new derivatives, associated matrix} of a well–generated tower.

In[∘]:= `(* This is an example of an S-primitive tower satisfying (CLI) and (MI) to be used for input. *)`
$\text{WellGenLogTower}\left[x, \{t1, t2, t3, t4\}, \left\{\frac{1}{x}, \frac{1}{1+x}, \frac{1}{t1} + \frac{1}{x}, \frac{1}{t1} + \frac{1}{t2} + \frac{1}{x}\right\}\right]$

Out[∘]= $\left\{\{u1, u2, u3, u4\}, \{\{1, 0, 1, 1\}, \{0, 1, 0, 0\}, \{0, 0, 1, 1\}, \{0, 0, 0, 1\}\},\right.$
$\left.\left\{\frac{1}{x}, \frac{1}{1+x}, \frac{1}{u1}, \frac{1}{u2}\right\}, \left\{\left\{\frac{1}{x}, \frac{1}{1+x}, 0, 0\right\}, \left\{0, 0, \frac{1}{u1}, 0\right\}, \left\{0, 0, 0, \frac{1}{u2}\right\}, \{0, 0, 0, 0\}\right\}\right\}$

In[∘]:=
```
(* This is an example of an S-
 primitive tower not satisfying (CLI) and (MI) to be used for input. *)
(* Note that u4' is u3-simple. *)
WellGenLogTower[x, {t1, t2, t3, t4},
 {1/x, 1/(1 + x), 1/t1, (2 + t1 + t1 t3^2 + 2 x + 2 t3 x + t1 t3^2 x + 2 t3 x^2)/(t1 (1 + x) (t2 + 2 t3 + t3^2 x))}]
```

Out[∘]=
```
{{u1, u2, u3, u4}, {{1, 0, 0, 1}, {0, 1, 0, 0}, {0, 0, 1, 0}, {0, 0, 0, 1}},
 {1/x, 1/(1 + x), 1/u1, (-u1 u2 - 2 u1 u3 + 2 x + u1 x - u1 u2 x - 2 u1 u3 x + 2 x^2 + 2 u3 x^2 + 2 u3 x^3) /
   (u1 x (1 + x) (u2 + 2 u3 + u3^2 x))}, {{1/x, 1/(1 + x), 0, 0}, {0, 0, 1/u1, 0}, {0, 0, 0, 0},
  {0, 0, 0, (-u1 u2 - 2 u1 u3 + 2 x + u1 x - u1 u2 x - 2 u1 u3 x + 2 x^2 + 2 u3 x^2 + 2 u3 x^3) /
    (u1 x (1 + x) (u2 + 2 u3 + u3^2 x))}}}
```

In[∘]:=
```
(* This is an example of a S-
 primitive tower not satisfying (CLI) and (MI) to be used for input. *)
(* Note that u4' is NOT u3-simple. *)
WellGenLogTower[x, {t1, t2, t3, t4},
 {1/x, 1/t1, 2/t1 + 1/(1 + x), (2 + t1 + t1 t2^2 + 2 x + 2 t2 x + t1 t2^2 x + 2 t2 x^2)/(t1 (1 + x) (t3 + t2^2 x))}]
```

Out[∘]=
```
{{u1, u2, u3, u4}, {{1, 0, 0, 1}, {0, 0, 1, 0}, {0, 1, 2, 0}, {0, 0, 0, 1}},
 {1/x, 1/(1 + x), 1/u1, (-u1 u2 - 2 u1 u3 + 2 x + u1 x - u1 u2 x - 2 u1 u3 x + 2 x^2 + 2 u3 x^2 + 2 u3 x^3) /
   (u1 x (1 + x) (u2 + 2 u3 + u3^2 x))}, {{1/x, 1/(1 + x), 0, 0}, {0, 0, 1/u1, 0}, {0, 0, 0, 0},
  {0, 0, 0, (-u1 u2 - 2 u1 u3 + 2 x + u1 x - u1 u2 x - 2 u1 u3 x + 2 x^2 + 2 u3 x^2 + 2 u3 x^3) /
    (u1 x (1 + x) (u2 + 2 u3 + u3^2 x))}}}
```

Part 3: Useful Functions

ExtendedEuclidean

In[∘]:=
```
? ExtendedEuclidean
```

ExtendedEuclidean[a, b, c, var] computes {r,s} such that
r a + s b = c and the degree of r with respect to var is lower than that of b.

In[∘]:=
```
Catch[ExtendedEuclidean[x (1 + x) (1 + y)/(-1 + y), (-1 + x) (1 + y)/(-2 + y), 1 + x^2, x]]
(* Check: r a + s b = c *)
Simplify[%[[1]] * x (1 + x) (1 + y)/(-1 + y) + %[[2]] * (-1 + x) (1 + y)/(-2 + y) - (1 + x^2)]
```

Out[∘]= $\left\{\frac{-1+y}{1+y}, \frac{2-y}{1+y}\right\}$

Out[∘]= 0

HermiteReduceGeneral

In[∘]:= `? HermiteReduceGeneral`

HermiteReduceGeneral[f, var, gen, der] outputs {g, h, p}, such that f = g' + h + p, where h is simple and p is a polynomial with respect to the last generator in gen.

```
(* A hand made example *)
HermiteReduceGeneral[(t2 + x)/(-t1 + t2)^2, x, {t1, t2}, {1/x, 1/t1}]
```

Out[∘]= $\left\{\frac{-t1^2\,x - t1\,x^2}{(t1 - t2)\,(t1 - x)}, \frac{t1^3 + 2\,t1^2\,x - 2\,x^2 - t1\,x^2}{(t1 - t2)\,(t1 - x)^2}, 0\right\}$

```
(* Example from Bronstein's book Example 2.2.1 *)
HermiteReduceGeneral[
 (x^7 - 24 x^4 - 4 x^2 + 8 x - 8) / (x^8 + 6 x^6 + 12 x^4 + 8 x^2), x, {}, {}]
```

Out[∘]= $\left\{\frac{3}{2 + x^2} + \frac{4\,(1 + 2\,x^2)}{x\,(2 + x^2)^2}, \frac{1}{x}, 0\right\}$

ProperDecomposition

In[∘]:= `? ProperDecomposition`

ProperDecomposition[f, {t_1, ..., t_n}] outputs a list of functions {f_0, f_1, ..., f_n} such that f = f_0 + f_1 + ... + f_n, f_i in K(t_1, ..., t_i)[t_{i+1}, ...t_n] with t_i–proper coefficients for all i with i > 0 and f_0 in K[t_1, ..., t_n].

In[∘]:=
```
ProperDecomposition[(t1^2 t2 + t2^2 + x + t1 x)/((t1 - x) (-t1 + t2 + x)), {t1, t2}]
```

Out[∘]= $\left\{1 + t1 + x, \frac{t2}{t1 - x} + \frac{x^2}{t1 - x}, \frac{-t1^2 - t1^3 - x + t1\,x + t1^2\,x - x^2}{(t1 - x)\,(t1 - t2 - x)}\right\}$

In[∘]:= ProperDecomposition$\left[\frac{1+t1}{x}, \{t1\}\right]$

Out[∘]= $\left\{\frac{1}{x}+\frac{t1}{x}, 0\right\}$

In[∘]:= ProperDecomposition$\Big[$
$\left(t2^2+t1\,t2^2+t3+t1\,t3+2\,t2\,t3+2\,t1\,t2\,t3+2\,t2^2\,t3+t1\,t2^2\,t3+t3^2+t1\,t3^2+4\,t2\,t3^2+2\,t1\,t2\,t3^2+2\,t3^3+t1\,t3^3+t2^2\,x+t1\,t2^2\,x+t3\,x+t1\,t3\,x+2\,t2\,t3\,x+2\,t1\,t2\,t3\,x+t2^2\,t3\,x+t3^2\,x+t1\,t3^2\,x+2\,t2\,t3^2\,x+t3^3\,x\right)\Big/\left((1+t1)\,t3\,(t2+t3)^2\,(1+x)\right)$
,
{t1,
t2,
t3}$\Big]$

Out[∘]= $\left\{\frac{1}{1+x}, \frac{1}{1+t1}, 0, \frac{t2^2+t3+2\,t2\,t3+t3^2}{t3\,(t2+t3)^2}\right\}$

Head Term

In[∘]:= ? HeadTerm

HeadTerm[f, gen] outputs the highest term among all of the leading terms of each projection of f with respect to the proper decomposition in the form of {coefficient, monomial}.

In[∘]:= HeadTerm$\left[\frac{t2}{t3}+\frac{t2^2\,t3}{t1}+t2\,t3^2, \{t1, t2, t3\}\right]$

HeadTerm$\Big[\frac{1}{(-1+t1)\,(1+t2)\,t3}\left(-t2+t1\,t2-t2^2+t1\,t2^2-t1^5\,t3^2+t1^6\,t3^2-2\,t2\,t3^2+2\,t1\,t2\,t3^2-t3^4+t1\,t3^4+t1\,t3^5+t1\,t2\,t3^5-t3\,x-t2\,t3\,x-t1\,t2\,t3\,x+t1^2\,t2\,t3\,x-t1\,t2^2\,t3\,x+t1^2\,t2^2\,t3\,x\right), \{t1, t2, t3\}\Big]$

Out[∘]= $\{1, t2\,t3^2\}$

Out[∘]= $\left\{1+\frac{1}{-1+t1}, t3^4\right\}$

MonomialIndicator

? MonomialIndicator

MonomialIndicator[monomial, gen] outputs the index of the lowest generator (with respect to the generator list, gen) appearing in the monomial. It outputs the length of gen if the monomial is 1.

In[•]:=
```
MonomialIndicator[x, {t1, t2, t3}]
MonomialIndicator[t2^3 t3^8, {t1, t2, t3}]
MonomialIndicator[t2^3 t3^8, {t3, t2, t1}]
MonomialIndicator[1, {t3, t2, t1}]
```

Out[•]= 3

Out[•]= 2

Out[•]= 1

Out[•]= 3

AddDecompLogTower

```
? AddDecompLogTower
? CheckAddDecompLogTower
```

AddDecompLogTower[f, var, gen, der] takes a function in a well-generated S-primitive tower and outputs the pair {integrable part of f, its minimal remainder (0 if f is integrable)} and all of the information needed to transform the old generators to the new ones in the form of a triple {new generator names, transfer matrix from old to new generators, new derivatives},

CheckAddDecompLogTower[f, var, gen, results] takes the results from AddDecompLogTower[f, var, gen, der] and outputs True if the decomposition is correct.

AddDecompLogTower$\left[t4, x, \{t1, t2, t3, t4\}, \left\{\frac{1}{x}, \frac{1}{t1}, \frac{2}{t1} + \frac{1}{1+x}, \frac{1}{t2} + \frac{1}{1+x}\right\}\right]$

```
CheckAddDecompLogTower @@ Append[{t4, x, {t1, t2, t3, t4}}, %]
```

Out[•]= $\left\{\left\{u2 - x + u2\, x + u4\, x, -\frac{x}{u3}\right\}, \{u1, u2, u3, u4\}, \{\{1, 0, 0, 0\}, \{0, 0, 1, 1\}, \{0, 1, 2, 0\}, \{0, 0, 0, 1\}\}, \left\{\frac{1}{x}, \frac{1}{1+x}, \frac{1}{u1}, \frac{1}{u3}\right\}\right\}$

Out[•]= True

```
AddDecompLogTower[t4, x, {t1, t2, t3, t4},
  {1/x, 1/t1, 2/t1 + 1/(1 + x), (2 + t1 + t1 t2^2 + 2 x + 2 t2 x + t1 t2^2 x + 2 t2 x^2)/(t1 (1 + x) (t3 + t2^2 x))}]
CheckAddDecompLogTower @@ Append[{t4, x, {t1, t2, t3, t4}}, %]
```

Out[∘]= $\{\{-x + u1\,x + u4\,x,\ (u1\,u2 + 2\,u1\,u3 - 2\,x - u1\,x + u1\,u2\,x + 2\,u1\,u3\,x - 2\,x^2 - 2\,u3\,x^2 - 2\,u3\,x^3) / (u1\,(1 + x)\,(u2 + 2\,u3 + u3^2\,x))\},$
$\{\{u1, u2, u3, u4\}, \{\{1, 0, 0, 1\}, \{0, 0, 1, 0\}, \{0, 1, 2, 0\}, \{0, 0, 0, 1\}\},$
$\{\frac{1}{x}, \frac{1}{1 + x}, \frac{1}{u1}, (-u1\,u2 - 2\,u1\,u3 + 2\,x + u1\,x - u1\,u2\,x - 2\,u1\,u3\,x + 2\,x^2 + 2\,u3\,x^2 + 2\,u3\,x^3) / (u1\,x\,(1 + x)\,(u2 + 2\,u3 + u3^2\,x))\}\}\}$

Out[∘]= True

In[∘]:=
```
(* The following shows that Mathematica is unable
 to integrate the integrable part (but we can!). *)
inttemp = Log[x + 1] - x + Log[x + 1] x + x Integrate[1 / LogIntegral[x], x]
ftemp = Log[x + 1] + Integrate[1 / LogIntegral[x], x]
remtemp = -x / LogIntegral[x]
Simplify[D[inttemp, x] + remtemp - ftemp]
Integrate[Together[ftemp - remtemp], x]
```

Out[∘]= $-x + x \int \frac{1}{\mathrm{LogIntegral}[x]}\,dx + \mathrm{Log}[1 + x] + x\,\mathrm{Log}[1 + x]$

Out[∘]= $\int \frac{1}{\mathrm{LogIntegral}[x]}\,dx + \mathrm{Log}[1 + x]$

Out[∘]= $-\frac{x}{\mathrm{LogIntegral}[x]}$

Out[∘]= 0

Out[∘]= $\int \left(\left(x + \left(\int \frac{1}{\mathrm{LogIntegral}[x]}\,dx\right) \mathrm{LogIntegral}[x] + \mathrm{Log}[1 + x]\,\mathrm{LogIntegral}[x]\right) \Big/ \mathrm{LogIntegral}[x]\right) dx$

Part 4: Testsuite

In[∘]:=
```
(* This collection of examples were used for
 timings. Examples provided by Christoph Koutschan. *)
christoph1 = Import["decomp_examples_collection1.m"];
```

```
(* Testing Christoph's examples that take longer than a second to compute. *)
list = {22, 52, 53, 54, 57, 58, 76, 81, 92, 93, 94, 95, 96, 98};
Do[
 Print[{list[[i]], Timing[adwgtemp = AddDecompLogTower @@ christoph1[[list[[i]]]];],
   CheckAddDecompLogTower @@
    Append[christoph1[[list[[i]]]][[1 ;; 3]], adwgtemp]}], {i, 1, Length[list]}]
```

{22, {0.868701, Null}, True}

{52, {6.58974, Null}, True}

{53, {1.86005, Null}, True}

{54, {9.83859, Null}, True}

{57, {11.7119, Null}, True}

{58, {16.3598, Null}, True}

{76, {1.79769, Null}, True}

{81, {17.6016, Null}, True}

{92, {1.41359, Null}, True}

{93, {1.90118, Null}, True}

{94, {1.41863, Null}, True}

{95, {1.90065, Null}, True}

{96, {1.53078, Null}, True}

{98, {1.04365, Null}, True}

```
(* 82 and 90 takes a long time, but at least it finishes! *)
```

```
Do[Print[{i, Timing[adwgtemp = AddDecompLogTower @@ christoph1[[i]];],
   CheckAddDecompLogTower @@
    Append[christoph1[[i]][[1 ;; 3]], adwgtemp]}], {i, 82, 82}]
```

{82, {7104.54, Null}, True}

```
Do[Print[{i, Timing[adwgtemp = AddDecompLogTower @@ christoph1[[i]];],
   CheckAddDecompLogTower @@
    Append[christoph1[[i]][[1 ;; 3]], adwgtemp]}], {i, 90, 90}]
```

{90, {3575.02, Null}, True}