

```

> read "C:/Users/DELL/Desktop/ISSAC2025CR/src/AddDecomp.txt";
> read "C:/Users/DELL/Desktop/ISSAC2025CR/src/DField.txt";
> read "C:/Users/DELL/Desktop/ISSAC2025CR/src/LogPart.txt";
> read "C:/Users/DELL/Desktop/ISSAC2025CR/src/PrimitiveCR.txt";
> read "C:/Users/DELL/Desktop/ISSAC2025CR/src/PrimitiveInt.txt";
> read "C:/Users/DELL/Desktop/ISSAC2025CR/src/Rational.txt";
> read "C:/Users/DELL/Desktop/ISSAC2025CR/src/Reduction.txt";

```

```

> # load modules
> with(AddDecomp); with(DField); with(LogPart);with(PrimitiveTower);with
  (PrimitiveInt);with(Rational);with(Reduction);
                                     [SPrimitive]
[AlgebraicDerivative, AssociatedMatrix, Derivative,
  FromFunctionalToRational, FromRationalToFunctional, Tower]
[CompleteLogarithmicPart, IsACompleteLogarithmicPart, LogarithmicPart,
  VerifyLogPart]
[AuxiliaryReduction, Basis, CollectTowerInfo, CompleteReduction, Projection,
  Verification]
[AlgRaab, ConstantCoefficient, DecompositionOfGenerators,
  DecompositionOfRemainder, DifferentiateResidue, LocalResidue,
  LogarithmicPartI, PolynomialMatrix, PrimInt]
[BasisElement, CoeffLaurent, Coefficient, EEA, ExtendedEuclidean,
  ExtendedEuclideanOld, HdegAndHcoeff, HeadTerm, LaurentCoeffAndMonomial,
  LaurentMatryoshka, LinearReduction, Matryoshka, NormalizedBasis,
  NormalizedElement, NormallyProperAndLaurentPolynomialParts,
  NumerAndMonicDenom, PadicExpand, ProperAndPolynomialParts, TdegAndTcoeff]
[GCanonicalForm, GKernelAndShellReduction, GKernelReduction,
  GShellReduction, GShellReduction1, HermiteReduce, Indicator,
  LaurentPolynomialReduction, LaurentPolynomialReduction1,
  LaurentPolynomialReduction2, ResidualForm0]

```

(1)

▼ Example 3.12

```

> # set up a tower
> K313 := x, [t], [1/x];
                                      $K313 := x, [t], \left[\frac{1}{x}\right]$ 

```

(1.1)

```

> # associate pairs
> CollectTowerInfo(K313);
                                      $\left[\left[0, \frac{1}{x}, \frac{1}{x}, 1, \left[\left[t, \frac{1}{x}\right]\right]\right]\right]$ 

```

(1.2)

```
> # input an integrand
> f := ((x+1)*t^2+(x^2+2*x+2)*t+x+1)/(x*(t+1));
```

$$f := \frac{(x+1) t^2 + (x^2 + 2 x + 2) t + x + 1}{x (t + 1)} \quad (1.3)$$

```
> # initial reduction
> (g, s, p) := op(HermiteReduce(K313, f));
```

$$g, s, p := 0, -\frac{x}{t+1}, \frac{t x + x^2 + t + x + 1}{x} \quad (1.4)$$

```
> # auxiliary reduction
> (q,r) := AuxiliaryReduction(K313, p);
```

$$q, r := t x + \frac{1}{2} x^2, \frac{t}{x} + \frac{1}{x} \quad (1.5)$$

```
> # projection
> (u,v) := Projection(K313, r, _TowerInfo[1][3], _TowerInfo[1][4]);
```

$$u, v := \frac{1}{2} t^2 + t, 0 \quad (1.6)$$

```
> # verification
> normal(Derivative(K313, g+q+u)+ (s+v) - f);
```

$$0 \quad (1.7)$$

```
> # complete reduction
> (I_Part, NI_Part) := CompleteReduction(K313, f);
```

$$I_Part, NI_Part := t x + \frac{1}{2} x^2 + \frac{1}{2} t^2 + t, -\frac{x}{t+1} \quad (1.8)$$

```
> # verification
> Verification(K313, f, I_Part, NI_Part);
```

$$correct \quad (1.9)$$

▼ Example 4.4

```
> # set up a tower
> K44 := x, [t1, t2], [1/(x-1), -t1/x];
```

$$K44 := x, [t1, t2], \left[\frac{1}{x-1}, -\frac{t1}{x} \right] \quad (2.1)$$

```
> # associate pairs
> CollectTowerInfo(K44);
```

$$\left[\left[0, \frac{1}{x-1}, \frac{1}{x-1}, 1, \left[\left[t1, \frac{1}{x-1} \right], \left[\frac{1}{2} t1^2, \frac{t1}{x-1} \right] \right] \right], \left[0, -\frac{t1}{x}, \frac{t1}{x}, -1, \left[\left[t2, -\frac{t1}{x} \right] \right] \right] \right] \quad (2.2)$$

```
> # input an integrand
> f := ((x-1)^2*t1+x)*t2^3+x*(x-1)*t1)/(x^2*(x-1)*t2^2); ft := t2^2;
```

$$f := \frac{((x-1)^2 t1 + x) t2^3 + x (x-1) t1}{x^2 (x-1) t2^2}$$

$$ft := t2^2 \quad (2.3)$$

> # initial reduction

> (g, s, p) := op(HermiteReduce(K44,f)); (gt, st, pt) := op(HermiteReduce(K44,ft));

$$g, s, p := \frac{1}{t2}, 0, \frac{(t1 x^2 - 2 t1 x + t1 + x) t2}{x^2 (x-1)}$$

$$gt, st, pt := 0, 0, t2^2 \quad (2.4)$$

> # auxiliary reduction

> (q,r) := AuxiliaryReduction(K44, p); (qt,rt) := AuxiliaryReduction(K44, pt);

$$q, r := \frac{t1 t2}{x} - \frac{t1^2}{x} + t1^2, \frac{t1 t2}{x} - \frac{2 t1}{x}$$

$$qt, rt := x t2^2 + (2 t1 x - 2 t1 - 2 x) t2 + 2 x t1^2 - 2 t1^2 - 6 x t1 + 6 t1 + 6 x, -\frac{2 t1^2}{x} \quad (2.5)$$

> # projection

> (u,v) := Projection(K44, r, _TowerInfo[2][3], _TowerInfo[2][4]);
(ut,vt) := Projection(K44, rt, _TowerInfo[2][3], _TowerInfo[2][4]);

$$u, v := -\frac{1}{2} t2^2 + 2 t2, 0$$

$$ut, vt := 0, -\frac{2 t1^2}{x} \quad (2.6)$$

> # verification

> normal(Derivative(K44, g+q+u)+ (s+v) - f); normal(Derivative(K44, gt+qt+ut)+ (st+vt) - ft);

$$0$$

$$0 \quad (2.7)$$

> # complete reduction

> (I_Part, NI_Part) := CompleteReduction(K44, f); (I_Part_t, NI_Part_t) := CompleteReduction(K44, ft);

$$I_Part, NI_Part := \frac{1}{t2} + \frac{t1 t2}{x} - \frac{t1^2}{x} + t1^2 - \frac{1}{2} t2^2 + 2 t2, 0$$

$$I_Part_t, NI_Part_t := x t2^2 + (2 t1 x - 2 t1 - 2 x) t2 + 2 x t1^2 - 2 t1^2 - 6 x t1 + 6 t1 + 6 x, -\frac{2 t1^2}{x} \quad (2.8)$$

> # verification

> Verification(K44, f, I_Part, NI_Part); Verification(K44, ft, I_Part_t, NI_Part_t);

correct

L

correct

(2.9)

▼ Example 4.5 (not available because there are not yet maple scripts for a complete reduction for algebraic functions in Maple)

▼ Example 5.4

```
> # set up a tower
> K54 := x, [t1, t2, t3], [1/(x-1), (1-t1)/x, 1/t1+1/x];
      K54 := x, [t1, t2, t3], [1/(x-1), (-t1+1)/x, 1/t1 + 1/x]
```

(4.1)

```
> # associate pairs
> CollectTowerInfo(K54);
[[[0, 1/(x-1), 1/(x-1), 1, [[t1, 1/(x-1)], [1/2 t1^2, (-t1+1)/x]]], [0, (-t1/x + 1/x), t1/x, -1,
[[t2, (-t1/x + 1/x)]]], [0, (x+t1)/(x t1), 1/x, 1, [[t3, (x+t1)/(x t1)]]]]]
```

(4.2)

```
> # input an integrand
> f := (x+(x-1)*t2)/((x-1)*t1)+(t2+(1-t1)*t3)/x;
      f := (x + (x-1) t2)/(x-1) t1 + (t2 + (-t1+1) t3)/x
```

(4.3)

```
> # determine elementary integrability
> g := PrimInt(K54,f);
      g := ln(t1) + t2 t3 - ln(x) + t3
```

(4.4)

```
> # verification
> F := subs({t1 = log(1-x), t2 = log(x)+polylog(2, x), t3 = ln(x)-Li(1-x)}, f); G := subs({t1 = log(1-x), t2 = log(x)+polylog(2, x), t3 = ln(x)-Li(1-x)}, g);
      F := (x + (x-1) (ln(x) + polylog(2, x)))/(x-1) ln(-x+1)
      + (ln(x) + polylog(2, x) + (-ln(-x+1) + 1) (ln(x) - Li(-x+1)))/x
      G := ln(ln(-x+1)) + (ln(x) + polylog(2, x)) (ln(x) - Li(-x+1)) - Li(-x+1)
```

(4.5)

```
> normal(diff(G,x)-F);
      0
```

(4.6)

```
> # try int
> int(F,x);
```

(4.7)

$$\left[\int \left(\frac{x + (x-1) (\ln(x) + \text{polylog}(2, x))}{(x-1) \ln(-x+1)} + \frac{\ln(x) + \text{polylog}(2, x) + (-\ln(-x+1) + 1) (\ln(x) - \text{Li}(-x+1))}{x} \right) dx \right] \quad (4.7)$$

▼ Example 5.4+ (removed from the submitted version due to lack of space)

```
> # set up a tower
> K := x, [t1, t2, t3], [1/x, 1/t1-1/(x+1), 1/(t1+1)+1/(x+2)];
      K := x, [t1, t2, t3], [1/x, 1/t1 - 1/(x+1), 1/(t1+1) + 1/(x+2)]
```

(5.1)

```
> # associate pairs
> CollectTowerInfo(K);
[[[0, 1/x, 1/x, 1, [[t1, 1/x]]], [0, 1/t1 - 1/(x+1), 1/(x+1), -1, [[t2, 1/t1 - 1/(x+1)]]],
 [0, (x+3+t1)/((t1+1)(x+2)), 1/(x+2), 1, [[t3, (x+3+t1)/((t1+1)(x+2)]]]]]
```

(5.2)

```
> # input an integrand
> f := (2*t1^2*t3*x+2*t1*t3*x^2+6*t1*t3*x+2*t1*x^2+4*t1*x+x^2+x-2)/((t1+1)*t1*x*(x+2));
      f := (2 t1^2 t3 x+2 t1 t3 x^2+6 t1 t3 x+2 t1 x^2+4 t1 x+x^2+x-2)/((t1+1) t1 x (x+2))
```

(5.3)

```
> # determine elementary integrability
> g := PrimInt(K,f);
      g := ln(t1+1) - ln(t1) + t3^2 + ln(x+1) - ln(x+2) + t2 + t3
```

(5.4)

```
> # verification
> F := subs([t1 = log(x), t2 = -log(x+1)+Li(x), t3 = log(x+2)+int(1/(log(x)+1), x)],f); G:=subs([t1 = log(x), t2 = -log(x+1)+Li(x), t3 = log(x+2)+int(1/(log(x)+1), x)],g);
      F := 1/((ln(x)+1) ln(x) x (x+2)) (2 ln(x)^2 (ln(x+2) - e^-1 Ei(1, -ln(x) - 1)) x+2 ln(x) (ln(x+2) - e^-1 Ei(1, -ln(x) - 1)) x^2+6 ln(x) (ln(x+2) - e^-1 Ei(1, -ln(x) - 1)) x+2 ln(x) x^2+4 x ln(x) +x^2+x-2)
      G := ln(ln(x)+1) - ln(ln(x)) + (ln(x+2) - e^-1 Ei(1, -ln(x) - 1))^2 + Li(x) - e^-1 Ei(1, -ln(x) - 1)
```

(5.5)

```
> simplify(evala(normal(diff(G,x)-F)));
      0
```

(5.6)

```
> # try int
```

```
> int(F,x);
```

$$\int \frac{1}{(\ln(x)+1) \ln(x) x (x+2)} (2 \ln(x)^2 (\ln(x+2) - e^{-1} \text{Ei}(1, -\ln(x) - 1)) x + 2 \ln(x) (\ln(x+2) - e^{-1} \text{Ei}(1, -\ln(x) - 1)) x^2 + 6 \ln(x) (\ln(x+2) - e^{-1} \text{Ei}(1, -\ln(x) - 1)) x + 2 \ln(x) x^2 + 4 x \ln(x) + x^2 + x - 2) dx \quad (5.7)$$

Example 5.7

```
> # set up a tower
```

```
> K57 := y, [t], [1/(x+y)];
```

$$K57 := y, [t], \left[\frac{1}{x+y} \right] \quad (6.1)$$

```
> # associate pairs
```

```
> CollectTowerInfo(K57);
```

$$\left[\left[0, \frac{1}{x+y}, \frac{1}{x+y}, 1, \left[\left[t, \frac{1}{x+y} \right] \right] \right] \right] \quad (6.2)$$

```
> # input a function
```

```
> f := 2*x/(t^2*x+t^2*y-x^2-x*y);
```

$$f := \frac{2 x}{t^2 x + t^2 y - x^2 - x y} \quad (6.3)$$

```
> # complete reduction
```

```
> (g0, r0) := CompleteReduction(K57, f);
```

$$g0, r0 := 0, \frac{2 x}{(x+y) (t^2 - x)} \quad (6.4)$$

```
> # complete reduction
```

```
> (g1,r1) := CompleteReduction(K57, Derivative(x, [t], [1/(x+y)], f));
```

$$g1, r1 := \frac{\left(\frac{1}{2} x + \frac{1}{2} y \right) \left(-\frac{2 t}{x+y} + \frac{4 x}{(x+y)^2} \right)}{t^2 - x}, \frac{1}{(x+y) (t^2 - x)} \quad (6.5)$$

```
> # look for a C(x)-linear relation between r0 and r1
```

```
> normal(2*x*r1-r0);
```

$$0 \quad (6.6)$$

```
> # the minmal telescoper is 2*x*Dx - 1
```

```
> # find a certificate
```

```
> g := 2*x*g1-g0;
```

$$(6.7)$$

$$g := \frac{2 \left(\frac{1}{2} x + \frac{1}{2} y \right) \left(-\frac{2 t}{x+y} + \frac{4 x}{(x+y)^2} \right) x}{t^2 - x} \quad (6.7)$$

> # verification

> normal(2*x*Derivative(x, [t], [1/(x+y)], f) - f - Derivative(K57,g))
;

0 (6.8)