Pensieve Header: Tests for FAA.nb

(Alt) In[ ]:=
```
SetDirectory["C:\\drorbn\\AcademicPensieve\\People\\Kuno"];
<< FAA.m
```

```
This is Profile.m of http://www.drorbn.net/AcademicPensieve/Projects/Profile/.

This version: April 2020. Original version: July 1994.
```

(Alt) In[ ]:=
```
BeginProfile[]
```
(Alt) Out[ ]=
```
ProfileRoot
```

Cyclic Words and the Trace

(Alt) In[ ]:=
```
RotateToMinimal[{x, y, x, x, y}]
```
(Alt) Out[ ]=
```
{x, x, y, x, y}
```

(Alt) In[ ]:=
```
AW2[x, x, y, y, x] // tr2→2
```
(Alt) Out[ ]=
```
CW2[x, x, x, y, y]
```

Bases

(Alt) In[ ]:=
```
Basis3,{x,y}[AW1]
```
(Alt) Out[ ]=
```
{AW1[x, x, x], AW1[x, x, y], AW1[x, y, x], AW1[x, y, y],
 AW1[y, x, x], AW1[y, x, y], AW1[y, y, x], AW1[y, y, y]}
```

(Alt) In[ ]:=
```
Basis3,{x,y}[CW1]
```
(Alt) Out[ ]=
```
{CW1[x, x, x], CW1[x, x, y], CW1[x, y, y], CW1[y, y, y]}
```

(Alt) In[ ]:=
```
Table[Length@Basisd,{x,y}[AW1], {d, 10}] // FindSequenceFunction
```
(Alt) Out[ ]=
```
2^#1 &
```

(Alt) In[ ]:=
```
Table[Length@Basisd,{x,y}[CW1], {d, 10}] // FindSequenceFunction
```
(Alt) Out[ ]=
```
FindSequenceFunction[{2, 3, 4, 6, 8, 14, 20, 36, 60, 108}]
```

(Alt) In[•]:=
```
Basis[3,{x,y}][AW[1] AW[2]]
```
(Alt) Out[•]=
```
{AW[1][] AW[2][x, x, x], AW[1][] AW[2][x, x, y], AW[1][] AW[2][x, y, x], AW[1][] AW[2][x, y, y],
 AW[1][] AW[2][y, x, x], AW[1][] AW[2][y, x, y], AW[1][] AW[2][y, y, x], AW[1][] AW[2][y, y, y], AW[1][x] AW[2][x, x],
 AW[1][x] AW[2][x, y], AW[1][x] AW[2][y, x], AW[1][x] AW[2][y, y], AW[1][y] AW[2][x, x], AW[1][y] AW[2][x, y],
 AW[1][y] AW[2][y, x], AW[1][y] AW[2][y, y], AW[1][x, x] AW[2][x], AW[1][x, x] AW[2][y], AW[1][x, y] AW[2][x],
 AW[1][x, y] AW[2][y], AW[1][y, x] AW[2][x], AW[1][y, x] AW[2][y], AW[1][y, y] AW[2][x], AW[1][y, y] AW[2][y],
 AW[1][x, x, x] AW[2][], AW[1][x, x, y] AW[2][], AW[1][x, y, x] AW[2][], AW[1][x, y, y] AW[2][],
 AW[1][y, x, x] AW[2][], AW[1][y, x, y] AW[2][], AW[1][y, y, x] AW[2][], AW[1][y, y, y] AW[2][]}
```

(Alt) In[•]:=
```
Basis[3,{x,y}][AW[1] CW[2]]
```
(Alt) Out[•]=
```
{AW[1][] CW[2][x, x, x], AW[1][] CW[2][x, x, y], AW[1][] CW[2][x, y, y], AW[1][] CW[2][y, y, y],
 AW[1][x] CW[2][x, x], AW[1][x] CW[2][x, y], AW[1][x] CW[2][y, y], AW[1][y] CW[2][x, x],
 AW[1][y] CW[2][x, y], AW[1][y] CW[2][y, y], AW[1][x, x] CW[2][x], AW[1][x, x] CW[2][y], AW[1][x, y] CW[2][x],
 AW[1][x, y] CW[2][y], AW[1][y, x] CW[2][x], AW[1][y, x] CW[2][y], AW[1][y, y] CW[2][x], AW[1][y, y] CW[2][y],
 AW[1][x, x, x] CW[2][], AW[1][x, x, y] CW[2][], AW[1][x, y, x] CW[2][], AW[1][x, y, y] CW[2][],
 AW[1][y, x, x] CW[2][], AW[1][y, x, y] CW[2][], AW[1][y, y, x] CW[2][], AW[1][y, y, y] CW[2][]}
```

Multiplication

(Alt) In[•]:=
```
AW[7][x, y, x] AW[3][x, x, y] + AW[7][x, x] AW[3][y, y] // m[7,3→5]
```
(Alt) Out[•]=
```
AW[5][x, x, y, y] + AW[5][x, y, x, x, x, y]
```

(Alt) In[•]:=
```
AW[7][x, y, x] AW[3][] // m[7,3→5]
```
(Alt) Out[•]=
```
AW[5][x, y, x]
```

Testing associativity:

(Alt) In[•]:=
```
bas = Basis[3,{x,y}][AW1 AW2 AW3]
(bas // m1,2→1 // m1,3→1) == (bas // m2,3→2 // m1,2→1)
```
(Alt) Out[•]=
```
{AW1[] AW2[] AW3[x, x, x], AW1[] AW2[] AW3[x, x, y], AW1[] AW2[] AW3[x, y, x],
 AW1[] AW2[] AW3[x, y, y], AW1[] AW2[] AW3[y, x, x], AW1[] AW2[] AW3[y, x, y],
 AW1[] AW2[] AW3[y, y, x], AW1[] AW2[] AW3[y, y, y], AW1[] AW2[x] AW3[x, x],
 AW1[] AW2[x] AW3[x, y], AW1[] AW2[x] AW3[y, x], AW1[] AW2[x] AW3[y, y], AW1[] AW2[y] AW3[x, x],
 AW1[] AW2[y] AW3[x, y], AW1[] AW2[y] AW3[y, x], AW1[] AW2[y] AW3[y, y], AW1[] AW2[x, x] AW3[x],
 AW1[] AW2[x, x] AW3[y], AW1[] AW2[x, y] AW3[x], AW1[] AW2[x, y] AW3[y], AW1[] AW2[y, x] AW3[x],
 AW1[] AW2[y, x] AW3[y], AW1[] AW2[y, y] AW3[x], AW1[] AW2[y, y] AW3[y], AW1[] AW2[x, x, x] AW3[],
 AW1[] AW2[x, x, y] AW3[], AW1[] AW2[x, y, x] AW3[], AW1[] AW2[x, y, y] AW3[],
 AW1[] AW2[y, x, x] AW3[], AW1[] AW2[y, x, y] AW3[], AW1[] AW2[y, y, x] AW3[],
 AW1[] AW2[y, y, y] AW3[], AW1[x] AW2[] AW3[x, x], AW1[x] AW2[] AW3[x, y], AW1[x] AW2[] AW3[y, x],
 AW1[x] AW2[] AW3[y, y], AW1[x] AW2[x] AW3[x], AW1[x] AW2[x] AW3[y], AW1[x] AW2[y] AW3[x],
 AW1[x] AW2[y] AW3[y], AW1[x] AW2[x, x] AW3[], AW1[x] AW2[x, y] AW3[], AW1[x] AW2[y, x] AW3[],
 AW1[x] AW2[y, y] AW3[], AW1[y] AW2[] AW3[x, x], AW1[y] AW2[] AW3[x, y], AW1[y] AW2[] AW3[y, x],
 AW1[y] AW2[] AW3[y, y], AW1[y] AW2[x] AW3[x], AW1[y] AW2[x] AW3[y], AW1[y] AW2[y] AW3[x],
 AW1[y] AW2[y] AW3[y], AW1[y] AW2[x, x] AW3[], AW1[y] AW2[x, y] AW3[], AW1[y] AW2[y, x] AW3[],
 AW1[y] AW2[y, y] AW3[], AW1[x, x] AW2[] AW3[x], AW1[x, x] AW2[] AW3[y], AW1[x, x] AW2[x] AW3[],
 AW1[x, x] AW2[y] AW3[], AW1[x, y] AW2[] AW3[x], AW1[x, y] AW2[] AW3[y], AW1[x, y] AW2[x] AW3[],
 AW1[x, y] AW2[y] AW3[], AW1[y, x] AW2[] AW3[x], AW1[y, x] AW2[] AW3[y], AW1[y, x] AW2[x] AW3[],
 AW1[y, x] AW2[y] AW3[], AW1[y, y] AW2[] AW3[x], AW1[y, y] AW2[] AW3[y], AW1[y, y] AW2[x] AW3[],
 AW1[y, y] AW2[y] AW3[], AW1[x, x, x] AW2[] AW3[], AW1[x, x, y] AW2[] AW3[],
 AW1[x, y, x] AW2[] AW3[], AW1[x, y, y] AW2[] AW3[], AW1[y, x, x] AW2[] AW3[],
 AW1[y, x, y] AW2[] AW3[], AW1[y, y, x] AW2[] AW3[], AW1[y, y, y] AW2[] AW3[]}
```
(Alt) Out[•]=
```
True
```

Word “Cutting”

(Alt) In[•]:=
```
AW7[x, y, x] AW3[x, x, y] // D[x]7→1,2
```
(Alt) Out[•]=
```
x7→1,2[AW3[x, x, y] AW7[x, y, x]]
```

(Alt) In[•]:=
```
CW2[x, x, y, y, x] // D[x]2→3
```
(Alt) Out[•]=
```
x2→3[CW2[x, x, y, y, x]]
```

The Co-Product

(Alt) In[•]:=
```
AW7[x, y, x] AW3[x, x, y] // Δ3→5,6
```
(Alt) Out[•]=
```
AW5[x, x, y] AW6[] AW7[x, y, x] + 2 AW5[x, y] AW6[x] AW7[x, y, x] + AW5[x, x] AW6[y] AW7[x, y, x] +
 AW5[y] AW6[x, x] AW7[x, y, x] + 2 AW5[x] AW6[x, y] AW7[x, y, x] + AW5[] AW6[x, x, y] AW7[x, y, x]
```

(Alt) In[•]:=
```
AW7[x, y, x] CW3[x, y, x, y] // Δ3→5,6
```
(Alt) Out[•]=
```
AW7[x, y, x] CW5[x, y, x, y] CW6[] +
 2 AW7[x, y, x] CW5[x, y, y] CW6[x] + 2 AW7[x, y, x] CW5[x, x, y] CW6[y] +
 AW7[x, y, x] CW5[y, y] CW6[x, x] + 4 AW7[x, y, x] CW5[x, y] CW6[x, y] +
 AW7[x, y, x] CW5[x, x] CW6[y, y] + 2 AW7[x, y, x] CW5[y] CW6[x, x, y] +
 2 AW7[x, y, x] CW5[x] CW6[x, y, y] + AW7[x, y, x] CW5[] CW6[x, y, x, y]
```

The Antipode

(Alt) In[•]:=
```
AW7[x, y, x] AW3[x, y, x, y] // S3→4
```
(Alt) Out[•]=
```
AW4[y, x, y, x] AW7[x, y, x]
```

(Alt) In[•]:=
```
AW7[x, y, x] AW3[x, y, y] // S3→4
```
(Alt) Out[•]=
```
- AW4[y, y, x] AW7[x, y, x]
```

(Alt) In[•]:=
```
AW7[x, y, x] CW3[x, y, x, y] // S3→4
```
(Alt) Out[•]=
```
AW7[x, y, x] CW4[x, y, x, y]
```

(Alt) In[•]:=
```
AW7[x, y, x] CW3[x, y, y] // S3→4
```
(Alt) Out[•]=
```
- AW7[x, y, x] CW4[x, y, y]
```

Substitutions

(Alt) In[•]:=
```
AW7[x, y, x] AW3[x, y, x, y] // FA[x → z + w, y → x]
```
(Alt) Out[•]=
```
AW3[w, x, w, x] AW7[w, x, w] + AW3[w, x, z, x] AW7[w, x, w] +
 AW3[z, x, w, x] AW7[w, x, w] + AW3[z, x, z, x] AW7[w, x, w] +
 AW3[w, x, w, x] AW7[w, x, z] + AW3[w, x, z, x] AW7[w, x, z] + AW3[z, x, w, x] AW7[w, x, z] +
 AW3[z, x, z, x] AW7[w, x, z] + AW3[w, x, w, x] AW7[z, x, w] + AW3[w, x, z, x] AW7[z, x, w] +
 AW3[z, x, w, x] AW7[z, x, w] + AW3[z, x, z, x] AW7[z, x, w] + AW3[w, x, w, x] AW7[z, x, z] +
 AW3[w, x, z, x] AW7[z, x, z] + AW3[z, x, w, x] AW7[z, x, z] + AW3[z, x, z, x] AW7[z, x, z]
```

(Alt) In[•]:=
```
AW7[x, y, x] AW3[x, y, x, y] // FA[x → y, y → x]
```
(Alt) Out[•]=
```
AW3[y, x, y, x] AW7[y, x, y]
```

(Alt) In[•]:=
```
Sum[AW1 @@ Table[x, k] / k!, {k, 0, 4}]
```
(Alt) Out[•]=
$$\mathrm{AW}_1[\,] + \mathrm{AW}_1[x] + \frac{1}{2}\,\mathrm{AW}_1[x, x] + \frac{1}{6}\,\mathrm{AW}_1[x, x, x] + \frac{1}{24}\,\mathrm{AW}_1[x, x, x, x]$$

(Alt) In[•]:=
```
Sum[AW1 @@ Table[x, k] / k!, {k, 0, 4}] // FA[x → x + y]
```
(Alt) Out[•]=
$$\begin{aligned}&\mathrm{AW}_1[\,] + \mathrm{AW}_1[x] + \mathrm{AW}_1[y] + \frac{1}{2}\,\mathrm{AW}_1[x, x] + \frac{1}{2}\,\mathrm{AW}_1[x, y] + \frac{1}{2}\,\mathrm{AW}_1[y, x] + \\ &\frac{1}{2}\,\mathrm{AW}_1[y, y] + \frac{1}{6}\,\mathrm{AW}_1[x, x, x] + \frac{1}{6}\,\mathrm{AW}_1[x, x, y] + \frac{1}{6}\,\mathrm{AW}_1[x, y, x] + \frac{1}{6}\,\mathrm{AW}_1[x, y, y] + \\ &\frac{1}{6}\,\mathrm{AW}_1[y, x, x] + \frac{1}{6}\,\mathrm{AW}_1[y, x, y] + \frac{1}{6}\,\mathrm{AW}_1[y, y, x] + \frac{1}{6}\,\mathrm{AW}_1[y, y, y] + \\ &\frac{1}{24}\,\mathrm{AW}_1[x, x, x, x] + \frac{1}{24}\,\mathrm{AW}_1[x, x, x, y] + \frac{1}{24}\,\mathrm{AW}_1[x, x, y, x] + \frac{1}{24}\,\mathrm{AW}_1[x, x, y, y] + \\ &\frac{1}{24}\,\mathrm{AW}_1[x, y, x, x] + \frac{1}{24}\,\mathrm{AW}_1[x, y, x, y] + \frac{1}{24}\,\mathrm{AW}_1[x, y, y, x] + \frac{1}{24}\,\mathrm{AW}_1[x, y, y, y] + \\ &\frac{1}{24}\,\mathrm{AW}_1[y, x, x, x] + \frac{1}{24}\,\mathrm{AW}_1[y, x, x, y] + \frac{1}{24}\,\mathrm{AW}_1[y, x, y, x] + \frac{1}{24}\,\mathrm{AW}_1[y, x, y, y] + \\ &\frac{1}{24}\,\mathrm{AW}_1[y, y, x, x] + \frac{1}{24}\,\mathrm{AW}_1[y, y, x, y] + \frac{1}{24}\,\mathrm{AW}_1[y, y, y, x] + \frac{1}{24}\,\mathrm{AW}_1[y, y, y, y]\end{aligned}$$

Exterior Multiplication

(Alt) In[•]:=
```
EM4[
 Sum[AW1 @@ Table[x, k] / k!, {k, 0, 4}],
 Sum[AW2 @@ Table[x, k] / k!, {k, 0, 4}]
]
```
(Alt) Out[•]=
$$\begin{aligned}&\mathrm{AW}_1[\,]\,\mathrm{AW}_2[\,] + \mathrm{AW}_1[x]\,\mathrm{AW}_2[\,] + \frac{1}{2}\,\mathrm{AW}_1[x, x]\,\mathrm{AW}_2[\,] + \frac{1}{6}\,\mathrm{AW}_1[x, x, x]\,\mathrm{AW}_2[\,] + \\ &\frac{1}{24}\,\mathrm{AW}_1[x, x, x, x]\,\mathrm{AW}_2[\,] + \mathrm{AW}_1[\,]\,\mathrm{AW}_2[x] + \mathrm{AW}_1[x]\,\mathrm{AW}_2[x] + \frac{1}{2}\,\mathrm{AW}_1[x, x]\,\mathrm{AW}_2[x] + \\ &\frac{1}{6}\,\mathrm{AW}_1[x, x, x]\,\mathrm{AW}_2[x] + \frac{1}{2}\,\mathrm{AW}_1[\,]\,\mathrm{AW}_2[x, x] + \frac{1}{2}\,\mathrm{AW}_1[x]\,\mathrm{AW}_2[x, x] + \frac{1}{4}\,\mathrm{AW}_1[x, x]\,\mathrm{AW}_2[x, x] + \\ &\frac{1}{6}\,\mathrm{AW}_1[\,]\,\mathrm{AW}_2[x, x, x] + \frac{1}{6}\,\mathrm{AW}_1[x]\,\mathrm{AW}_2[x, x, x] + \frac{1}{24}\,\mathrm{AW}_1[\,]\,\mathrm{AW}_2[x, x, x, x]\end{aligned}$$

(Alt) In[•]:=

```
PrintProfile[]
```

PaddedForm: Formatting specification {Indeterminate, 3} should be a positive integer or a pair of positive integers.

PaddedForm: Formatting specification {Indeterminate, 3} should be a positive integer or a pair of positive integers.

PaddedForm: Formatting specification {Indeterminate, 3} should be a positive integer or a pair of positive integers.

General: Further output of PaddedForm::iprf will be suppressed during this calculation.

(Alt) Out[•]=

```
ProfileRoot is root. Profiled time: 0.
  ( 6)  0/ 0 above FAAm
FAAm: called 6 times, time in 0./0.
  ( 6)  0/ 0 under ProfileRoot
```