Some Computations in $A^{n,l,m}$ using VasCalc

First we load up the definitions, and initialize the program: (see download and setup instructions at http://katlas.math.toronto.edu/drorbn/index.php?title=VasCalc)

In[1]:=
```
<< C:/drorbn/projects/06VasCalc/trunk/CDinterface.m
SetVasCalcPath["C:/drorbn/projects/06VasCalc/trunk"]
```

The spaces of (rational linear combinations of) chord diagrams modulo the 4-T relations are accessed by the objects CDSpace[# of lines, # of circles, # of chords]. Constructing these spaces may take some time, but only needs to be done once - the result is stored on disc for future use.
Here we find the dimension of the space with five chords on a skeleton of one line...

In[3]:=
```
GetDimension[CDSpace[1, 0, 5]]
```

Out[3]=
```
10
```

...which is isomorphic to closing the line to a circle:

In[4]:=
```
GetDimension[CDSpace[0, 1, 5]]
```

Out[4]=
```
10
```

We can also find a basis for such a space. The notation to represent a chord diagram is as follows: number all the chords, and for each line (or circle), write down the numbers of the chords that one passes as one walks up the line (around the circle).
For example, here's the basis for $A^{1,1,3}$

In[5]:=
```
GetCDBasis[CDSpace[1, 1, 3]]
```

Out[5]=
```
{CD[Line[1, 1], Circle[2, 2, 3, 3]],
 CD[Line[1, 1], Circle[2, 3, 2, 3]], CD[Line[1], Circle[1, 2, 3, 2, 3]],
 CD[Line[1], Circle[1, 2, 3, 3, 2]], CD[Line[], Circle[1, 1, 2, 3, 3, 2]],
 CD[Line[1, 2, 1, 2], Circle[3, 3]], CD[Line[], Circle[1, 2, 1, 3, 2, 3]],
 CD[Line[1, 2], Circle[1, 3, 3, 2]], CD[Line[1, 2, 2, 1], Circle[3, 3]],
 CD[Line[1, 2, 2], Circle[1, 3, 3]], CD[Line[], Circle[1, 2, 3, 1, 2, 3]],
 CD[Line[1, 2, 3], Circle[1, 3, 2]], CD[Line[1, 2, 3, 2, 1, 3], Circle[]],
 CD[Line[1, 2, 3, 2], Circle[1, 3]], CD[Line[1, 2, 3, 3], Circle[1, 2]],
 CD[Line[1, 2, 3, 2, 3, 1], Circle[]], CD[Line[1, 2, 3, 2, 3], Circle[1]],
 CD[Line[1, 2, 3, 3, 2, 1], Circle[]], CD[Line[1, 2, 3, 3, 2], Circle[1]]}
```

Some operations (addition, scalar multiplication, vector multiplication, and a few operations that alter the skeleton) have also been implemented: the algebraic operations actually take place in the graded algebra $A^{l,m}$on a skeleton of l lines and m circles. For example, let's demonstrate the commutativity of

$A^{1,0}$.

Define two chord diagrams as follows - the "ASeries" denotes a linear combination of chord diagrams that lives in $A^{l,m}$:

```
In[6]:= cd1 = ASeries[{1}, 0, CD[Line[1, 2, 2, 1]], 5]

Out[6]= ASeries[{1}, 0,
  {« JavaObject[vectorSpace.Coefficient] », « JavaObject[vectorSpace.Coefficient] »,
   « JavaObject[ChordVector] », « JavaObject[vectorSpace.Coefficient] »,
   « JavaObject[vectorSpace.Coefficient] », « JavaObject[vectorSpace.Coefficient] »}]

In[7]:= cd2 = ASeries[{1}, 0, CD[Line[1, 2, 3, 1, 2, 3]], 5]

Out[7]= ASeries[{1}, 0,
  {« JavaObject[vectorSpace.Coefficient] », « JavaObject[vectorSpace.Coefficient] »,
   « JavaObject[vectorSpace.Coefficient] », « JavaObject[ChordVector] »,
   « JavaObject[vectorSpace.Coefficient] », « JavaObject[vectorSpace.Coefficient] »}]
```

The operator "." returns the concatenation of the two diagrams. Also, note that the internal representation of these objects is in Java, so we use the command "// CD" to convert back to human-readable notation:

```
In[8]:= cd1.cd2 // CD

Out[8]= CD[Line[1, 2, 2, 1, 3, 4, 5, 3, 4, 5]]

In[9]:= cd2.cd1 // CD

Out[9]= CD[Line[1, 2, 3, 1, 2, 3, 4, 5, 5, 4]]
```

These certainly are different diagrams, but in the quotient obtained from the 4T relations, they are equivalent:

```
In[10]:= Reduce[cd1.cd2 - cd2.cd1] // CD

Out[10]= 0
```

In fact, every basis element of (say) $A^{1,0,5}$ commutes with every other:

```
In[11]:= basis = ASeries[{1}, 0, #, 5] & /@ GetCDBasis[CDSpace[1, 0, 5]];

In[12]:= Flatten[ Table[
   CD[Reduce[basis[[i]].basis[[j]] - basis[[j]].basis[[i]]]], {i, 10}, {j, 10}]]

Out[12]= {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}

In[14]:= GetDimension[CDSpace[1, 0, #]] & /@ Range[5]

Out[14]= {1, 2, 3, 6, 10}
```