CoCoA-5 - Feature #1469

Get indexed indets from a polyring

25 Jun 2020 05:45 - John Abbott

Status:	New	**Start date:**	25 Jun 2020
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	CoCoA-5 function: new	**Estimated time:**	0.00 hour
Target version:	CoCoA-5.4.2	**Spent time:**	0.60 hour

Description

The using feature of CoCoA-4 has been removed in CoCoA-5. This can cause awkwardness when the ring to be used contains indets with indices.

The function indets(P,str) does not solve the matter nicely.

Make a function which make accessing indets with indexes easy. I am thinking of something like this:

```
P ::= QQ[x,y[1..3,1..3]];
y := IndetMap(P,"y");
y[2,2];  --> gives what you would expect
```

Another (contrived) example:

```
P ::= QQ[a[2],a[1],a[3]];
a := IndetMap(P,"a");
a[2]; --> gives 1st indet in P
```

Related issues:

Related to CoCoA-5 - Design #292: Rename IndetsCalled to indets	**Closed**	**17 Jan 2013**
Related to CoCoALib - Feature #1470: Get indexed indets from a polyring	**New**	**25 Jun 2020**
Related to CoCoA-5 - Feature #723: Referring to indets with multiple indices ...	**New**	**02 Jun 2015**

History

#1 - 25 Jun 2020 05:53 - John Abbott

Note that CoCoA-5 already has the (undocumented?!?) type INTMAP which does exactly what we want.

Just need to make it accessible via a function (and possibly lift any weird restrictions that might have been imposed?).
I wonder where the source code is... somewhere inside the interpreter (see near line 519).

JAA has just checked that an INTMAP may be assigned and passed as a parameter.

#2 - 25 Jun 2020 05:54 - John Abbott

There should probably be a similar fn available in CoCoALib; this implies that we'll need a type for INTMAP.

#3 - 25 Jun 2020 06:22 - John Abbott

- Related to Design #292: Rename IndetsCalled to indets added

#4 - 25 Jun 2020 09:48 - John Abbott

- Related to Feature #1470: Get indexed indets from a polyring added

#5 - 30 Oct 2020 12:29 - John Abbott

- Related to Feature #723: Referring to indets with multiple indices from a polyring added

#6 - 19 Jan 2021 12:39 - Julian Danner

Hi,
I would highly appreciate such a function, however I think it would be even better if one must not first initialize the variables, but could access them directly through the rings name P.
E.g. I would like to write

```
P::= QQ[x,y[1..3,1..3]];
P.y[2,2];  --> gives what you would expect
```

similar to what one could do with :: in CoCoA-4. (This could simplify the code without impact on readability.)

I wrote a function that follows this idea called inject and it can be found as part of the apcocoa-packages here:
https://git.fim.uni-passau.de/walsh/apcocoa-packages/-/blob/master/apcocoa/misc.cpkg5 with examples on usage in
https://git.fim.uni-passau.de/walsh/apcocoa-packages/-/blob/master/tests/misc.cocoa5. My function returns a record that stores the indets inside nested lists. The only limitation here is that it only works for strictly positive indices.
Using my function one can write the above example as follows:

```
P::= QQ[x,y[1..3,1..3]];
P:=inject(P);
P.y[2,2];  --> gives what you would expect
```

In particular it is not required to write any other lines to be able to access possible other indets. (And the original ring can be found in P.ring_.)

But in any case, just having this IndetMap would also be fine for me, since I could adapt my implementation to be correct in general (by replacing each field-entry with the corresponding INTMAP).