

## CoCoALib - Feature #1470

### Get indexed indets from a polyring

25 Jun 2020 09:48 - John Abbott

<b>Status:</b>	New	<b>Start date:</b>	25 Jun 2020
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>	New Function	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	CoCoALib-0.99900	<b>Spent time:</b>	0.25 hour
<b>Description</b>			
Make a class so that indexed indets from a polyring can be used with their "natural" indexes (rather than, say, c++ vector indexes).			
<pre>PolyRing P = NewPolyRing(RingQQ(), SymbolRange("x",1,4)); IndexedIndet x(P, "x"); cout &lt;&lt; x(3) &lt;&lt; endl; // prints x[3]</pre>			
Note that square brackets can be overloaded, but only with a single arg. Round brackets can be overloaded with several args -- I suggest overloads for 1,2,3 long and vector<long>			
<b>Related issues:</b>			
Related to CoCoA-5 - Feature #1469: Get indexed indets from a polyring		<b>New</b>	<b>25 Jun 2020</b>

### History

#### #1 - 25 Jun 2020 09:48 - John Abbott

- Related to Feature #1469: Get indexed indets from a polyring added

#### #2 - 25 Jun 2020 09:51 - John Abbott

- Description updated

As usual, indexes should be of type long (no need for BigInt).

It seems reasonable to allow a "short-cut" for up to 3 indexes; certainly 1 and 2, but why not 3 as well?

Probably not worth overloading operator[]; it might be confusing to a C++ reader (who would think of a vector), and it works only for 1 index, while operator() can be made more flexible.

#### #3 - 07 Mar 2024 20:36 - John Abbott

- Target version changed from CoCoALib-0.99850 to CoCoALib-0.99900