# CoCoA-5 - Feature #588

## Resume code for "gin" (generic initial ideal)

15 Jul 2014 15:42 - Anna Maria Bigatti

| Status: | Closed | | Start date: | 15 Jul 2014 |
|---|---|---|---|---|
| Priority: | Normal | | Due date: | |
| Assignee: | Anna Maria Bigatti | | % Done: | 100% |
| Category: | CoCoA-4 function to be added | | Estimated time: | 2.00 hours |
| Target version: | CoCoA-5.1.1 Seoul14 | | Spent time: | 2.25 hours |
| **Description** | | | | |
| Resurrect code for "gin", and implement it in cocoalib | | | | |
| **Related issues:** | | | | |
| Related to CoCoALib - Feature #589: Implement gin (generic initial ideal) in ... | | **New** | | **15 Jul 2014** |
| Related to CoCoA-5 - Feature #883: gin: return/print a suitable change of var... | | **Closed** | | **10 May 2016** |

---

**History**

#### #1 - 15 Jul 2014 18:14 - Anna Maria Bigatti

- *Status changed from In Progress to Feedback*

- *% Done changed from 10 to 90*

- *Estimated time set to 2.00 h*


Done in package.
Implemented only using twin-floats with range [-10^6, 10^6].
No point in offering more flexibility than this!


#### #2 - 27 Aug 2014 18:12 - Anna Maria Bigatti

- *Status changed from Feedback to Closed*

- *% Done changed from 90 to 100*


#### #3 - 28 Aug 2014 14:19 - John Abbott

I notice that the current definition will sometimes (almost never?) print out a message about trying again.  This is not good programming style :-(

I think that gin should just calculate the answer; maybe you could have a second function which returns a record containing the GIN and another field containing information about how hard it worked to get the answer (*e.g.* how many iterations it had to perform).  The function gin would then just call the second function, and simply return the value of field containing the answer (discarding the other information).


#### #4 - 02 Sep 2014 08:28 - Anna Maria Bigatti

John Abbott wrote:

> I notice that the current definition will sometimes (almost never?) print out a message about trying again.  This is not good programming style :-(


ok, I modified it adding an optional "verbose" flag

```
/**/  gin(ideal(y^7-x^4*z^3, x^5*z-y*z^5), "verbose");
-- trying with FloatPrecision 64
-- trying with FloatPrecision 64
ideal(x^6, x^5*y^2, x^4*y^4, x^3*y^6, x^2*y^8, x*y^10, y^12)
```

**#5 - 15 Sep 2014 18:28 - Anna Maria Bigatti**

*- Subject changed from Resurrect code for "gin" (generic initial ideal) to Resume code for "gin" (generic initial ideal)*


**#6 - 07 Apr 2017 16:31 - Anna Maria Bigatti**

*- Related to Feature #883: gin: return/print a suitable change of variables added*