CoCoA-5 - Feature #502

New fn/object type: timer

30 Mar 2014 21:58 - John Abbott

Status:	Closed	**Start date:**	30 Mar 2014
Priority:	Normal	**Due date:**	
Assignee:	Anna Maria Bigatti	**% Done:**	100%
Category:	CoCoA-5 function: new	**Estimated time:**	3.00 hours
Target version:	CoCoA-5.1.0 Easter14	**Spent time:**	2.10 hours

Description

Here's an idea. Create a new fn called **timer** (or *NewTimer) which creates a TIMER object.
When you evaluate the object it simply return the amount of CPU time used since it was created.

Thus to time a computation you could do this:

```
T := timer();
XYZ := LongComputation();
PrintLn "Time taken: ", DecimalStr(T);
```

Is this a good idea? Is it worth it?
Does it give any real advantage over CpuTime()?

2014-04 conclusion: this solution is now implemented

```
T := CpuTime();
-- long computation --
TimeFrom(T);      -- to print it in decimal
CpuTimeFrom(T);  -- to have it as a usable number (as CpuTime())
```

History

#1 - 30 Mar 2014 22:01 - John Abbott

Once strange aspect of a TIMER object is that T=T may sometimes return false

I wonder whether the value produced by a timer should be a simple RAT or maybe a Record to help highlight that it is a not a "normal" value?

#2 - 30 Mar 2014 22:03 - John Abbott

What about a StrTimer which automatically returns a string?
That would avoid the tedious need to call DecimalStr.
Or if the value is a record, then one of the fields could be the DecimalStr of the time difference?

#3 - 31 Mar 2014 12:01 - Anna Maria Bigatti

- Status changed from New to In Progress

- Target version set to CoCoA-5.1.0 Easter14

- % Done changed from 0 to 10

you are right, it's annoying having to write **DecimalStr(...)**.
This is my trivial suggestion

```
Define CpuTimeFrom(T) return CpuTime()-T; EndDefine;
Define TimeFrom(T) return DecimalStr(CpuTimeFrom(T)); EndDefine;
```

so

```
T := CpuTime();
-- long computation --
TimeFrom(T);     -- to print it in decimal
CpuTimeFrom(T);  -- to have it as a usable number (as CpuTime())
```

Should I documented and publish it?

#4 - 31 Mar 2014 14:22 - John Abbott

I'm a little undecided.

I'm not so convinced by CpuTimeFrom(T); it's longer than CpuTime()-T and does not seem any clearer to me.

TimeFrom(T) is not bad; in some ways I'd like its name to contain Str somewhere, but that would make it longer...

#5 - 04 Apr 2014 16:34 - John Abbott

- % Done changed from 10 to 50

JAA suggests documenting and publishing TimeFrom but not CpuTimeFrom.
Really CpuTimeFrom seems superfluous to me; to me it offers no advantage over CpuTime -- anyone who needs access to the numerical value should have no trouble comprehending CpuTime. CpuTimeFrom just seems like pointless clutter... yet another function we shall have to maintain in the future.

#6 - 10 Apr 2014 01:08 - John Abbott

- % Done changed from 50 to 60

I have added an entry for TimeFrom to the C5 manual.

#7 - 15 Apr 2014 16:16 - Anna Maria Bigatti

- Status changed from In Progress to Feedback

- Assignee set to Anna Maria Bigatti

- % Done changed from 60 to 90

#8 - 02 May 2014 16:49 - John Abbott

- Status changed from Feedback to Closed

- % Done changed from 90 to 100

- Estimated time set to 3.00 h

TimeFrom is a satisfactory KISS solution. Still happy after 3 weeks, so closing.