

hifrog - Bug #13356

Bug in Hifrog LRA encoding - resulting assertion violation for a safe bench

26/09/2019 13:54 - Sepideh Asadi

Status:	Resolved	Start date:	26/09/2019
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:		Spent time:	0.00 hour
Description			
Benchmark: ex13-change-orig.c (attached)			
How to run: ./hifrog --logic qflra ex13-change-orig.c			
Error: LRA reports the bench as unsafe, while prop reports it as safe. The result of dump-query is SAT in OpenSMT and Z3, so the bug is in the encoding.			

History

#1 - 07/10/2019 00:11 - Sepideh Asadi

- Status changed from New to Resolved

This behaviour is Normal and expected in LRA, because LRA does not recognise int, all is Real number!

it is because of the following reason:

In C itself the correct behavior is $a1=a2$

but in LRA since in the encoding (cprover) we do optimization as

$x > c \rightarrow x \geq c+1$

for e.g., in the following c program contains integer

```
int x;
```

```
if (x>0)
```

```
assert(x>=1);
```

 in C itself, and Propositional logic it is safe

in LRA it is unsafe because of different nature of LRA that does not know integer, it considers it as Real: so $x=0.5$ cex is reported!

Files

ex13-change-orig.c	348 Bytes	26/09/2019	Sepideh Asadi
query_default-1.smt2	10.5 KB	26/09/2019	Sepideh Asadi