CAVERN: Constraints and Abstractions for program VERificatioN

**Constraint Programming**
Exploit relations (constraints) to infer new informations on objects that represent unknowns (variables)

- $U = R \times I$
- $\{2,3\} \times \{6\}$
- $\{1,2,3\} \times \{6,10\}$

**Abstractions**
Over-approximate the computation of relations to benefit from powerful solving techniques (e.g. Linear Programming)

WP4: Floating-point computations
Properties over floating-point data can be exploited to prune the search space.

**Program Verification**
Are properties P1a, P1b, ... verified by this implementation?

WP3: Integers and iterative computations
Fine-grained memory model of C and Bytecode Java

- PhD of Florence Charreteur (Mar. 2010)
- PhD of Mickael Delahaye (Oct. 2011)
- F. Charreteur, A. Gotlieb. Constraint-based test inputs generation for Java Bytecode. ISSRE 2010

**WP2: Memory models**
Fine-grained memory model of C and Bytecode Java

**WP1a**
`@behavior P1a : assumes flag = 0 & & n = 8 & & forall i in 1..8, ar[i] = cos(2*M_PI*i/n); ensures result == 0.`

**WP1b**
`@behavior P1b : assumes flag = 1 & & n = 8 & & forall i in 1..8, ar[i] = cos(2*M_PI*i/n); ensures result == 0;`

Contact: Arnaud.Gotlieb@inria.fr (INRIA Rennes, coordinator)

http://cavern.inria.fr