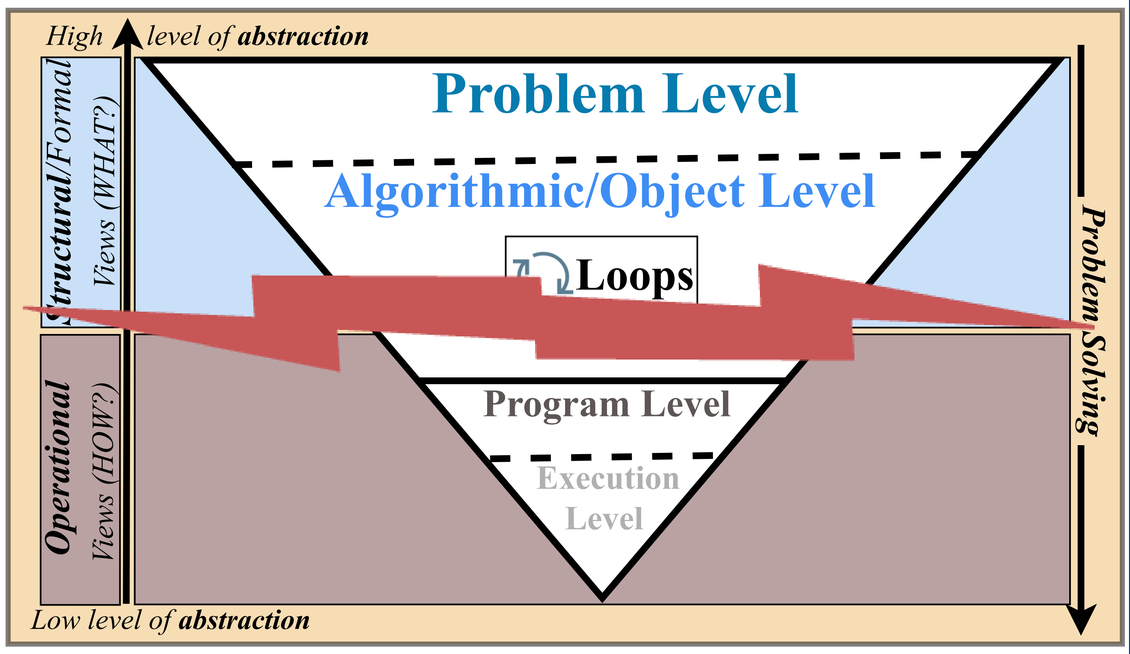
Introducing Formal Methods to Teach Loops in CS1: A Collaborative and Feedback-Driven Approach

Géraldine Brieven (gbrieven@uliege.be)
Université de Liège (Belgium)

1. Context and its three problematics

In our CS1 course, to solve problems, students first model their solution as modules that manipulate variables and objects *(formal view)*.

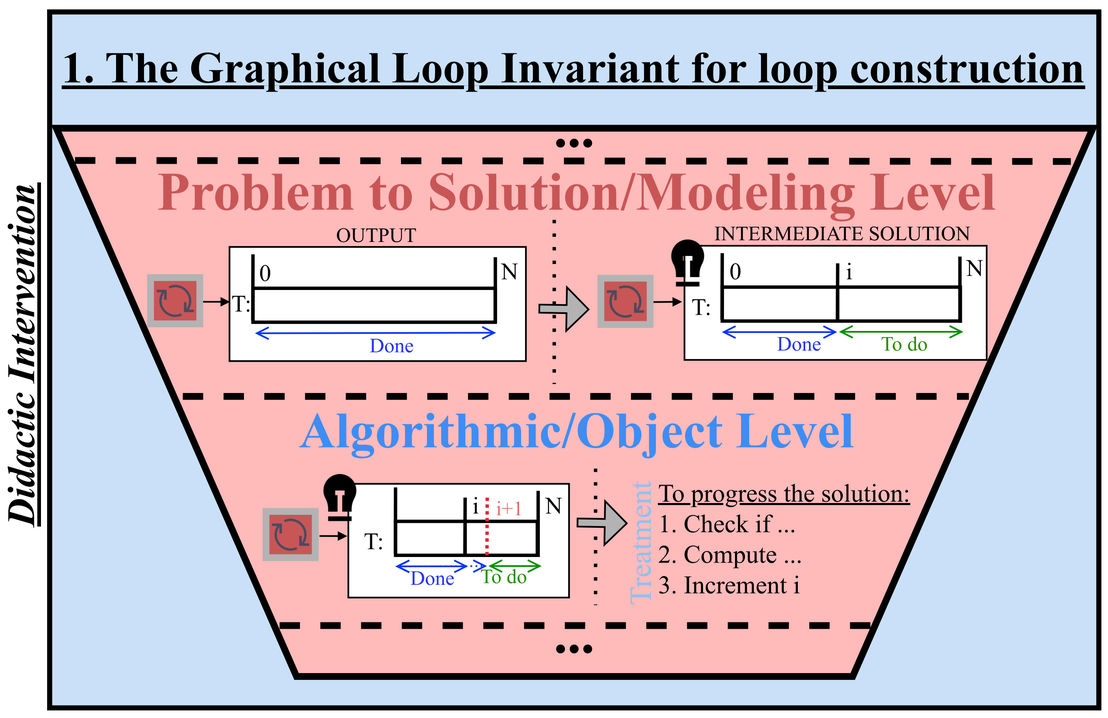


Problematics:

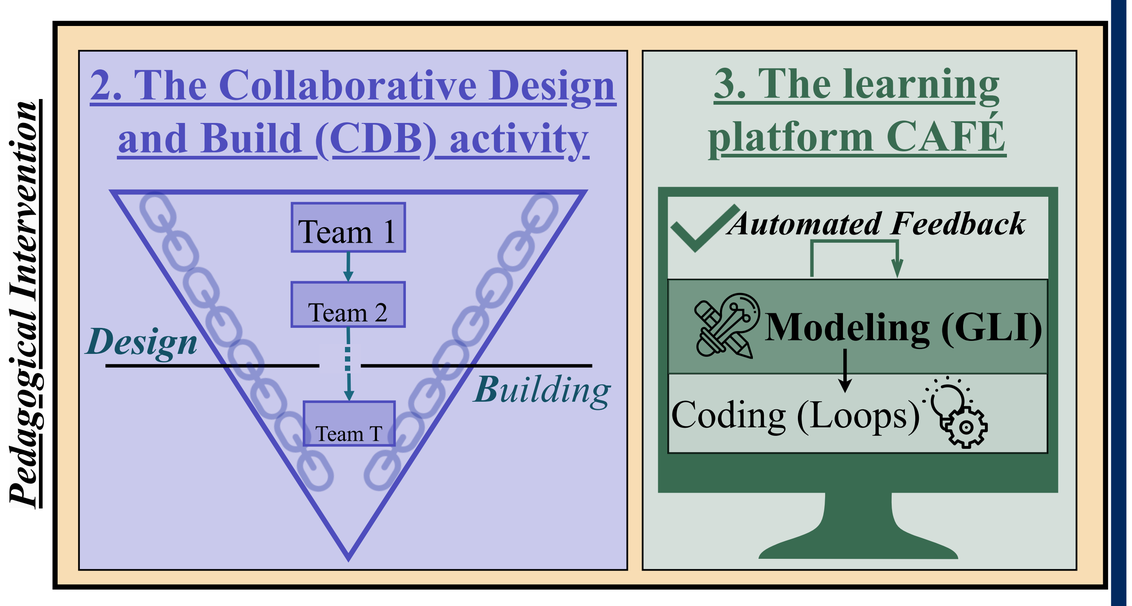
1. "Traditional" Formal Methods (FM) are not accessible to novices. → 2. Students miss interest in FM. → 3. Students miss feedback when they train on FM.

2. Three interventions

To bridge CS1 to formal methods, we introduce the **Graphical Loop Invariant (GLI)** as an additional problem-solving level.



We engage students through an **assembly-line activity** where they solve problems in teams. Students can also practice designing GLI and coding loops through regular homework assignments and they receive **personalized automated feedback** on their work.



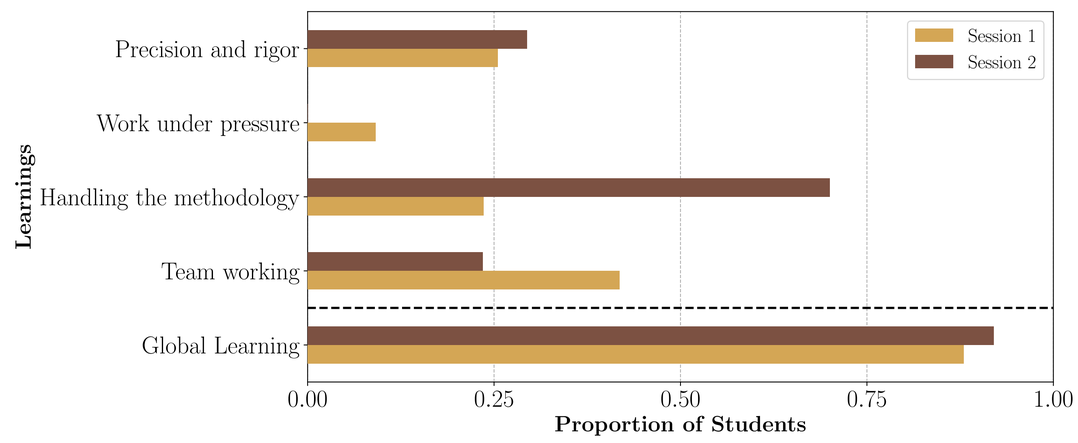
3. Four Research Questions

- **RQ1: Can the GLI approach be bridged to predicates?**
- **RQ2: Does the CDB activity motivate students on reasoning structurally (via the GLI)?**
- **RQ3: Does CAFÉ make students improve on the GLI?**
- **RQ4: Is the GLI efficient for students to write better code?**

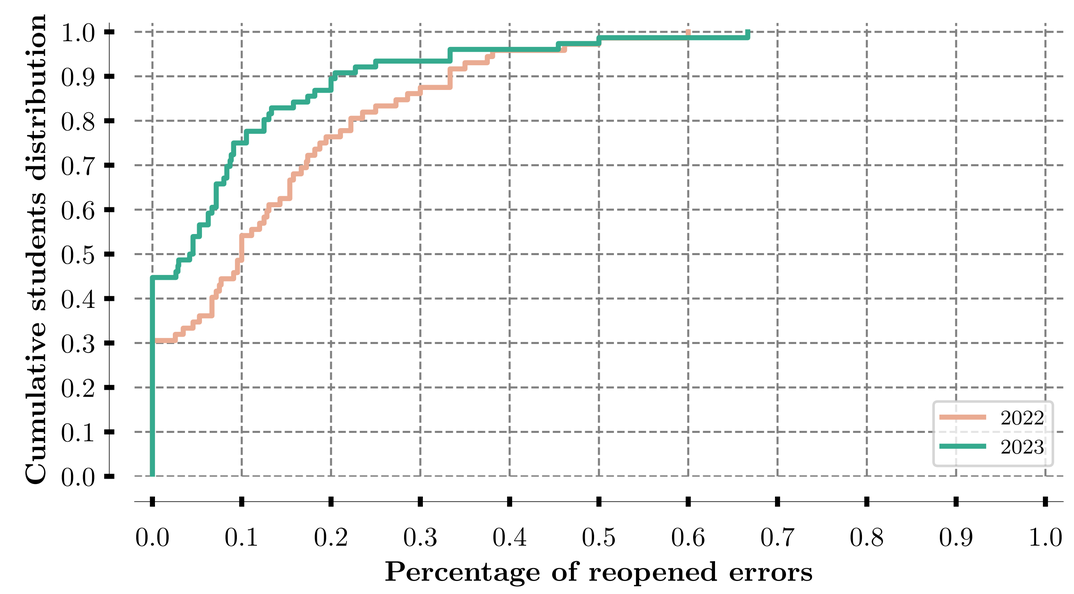
4. Experiments and results

- We translate students' **GLI and C code into Dafny code and annotations** using AST parsing and a sentence-BERT model. We then show that Dafny **successfully verifies consistency between the instructions and predicates.**

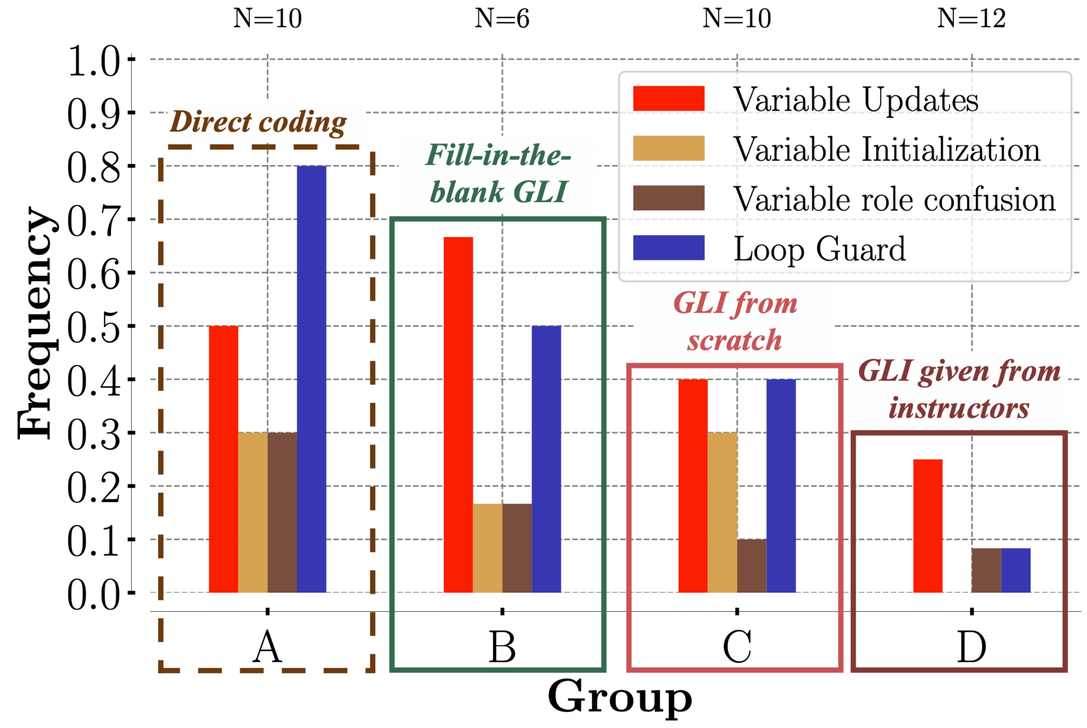
- We namely measure the **instant impact of the activity via a survey:**



- **We track typical errors students could fix in homework, but make again in the exam** (which indicates some remaining gaps):



- We conducted **A/B/C/D testing** where students solved a problem under different conditions:



Summer School on Informatics Education Research (SCHIER), September 8th, Louvain-La-Neuve (Belgium)