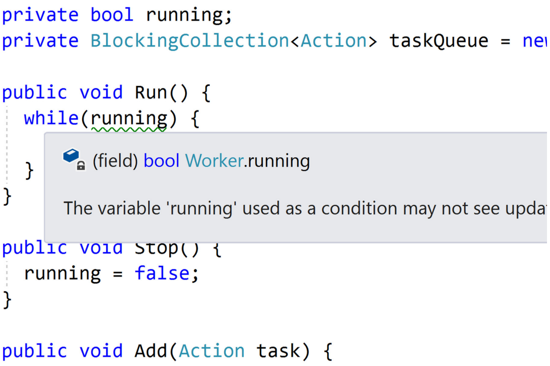
Christoph Josef Amrein

Students	Christoph Josef Amrein
Lecturers	Prof. Dr. Luc Bläser
Advisors	- -
Topic	Software and Systems

Concurrency Bug Finder

Roslyn-Based Static Code Analysis for C#



```
private bool running;
private BlockingCollection<Action> taskQueue = new

public void Run() {
  while(running) {

  }
}

public void Stop() {
  running = false;
}

public void Add(Action task) {
```

(field) bool Worker.running

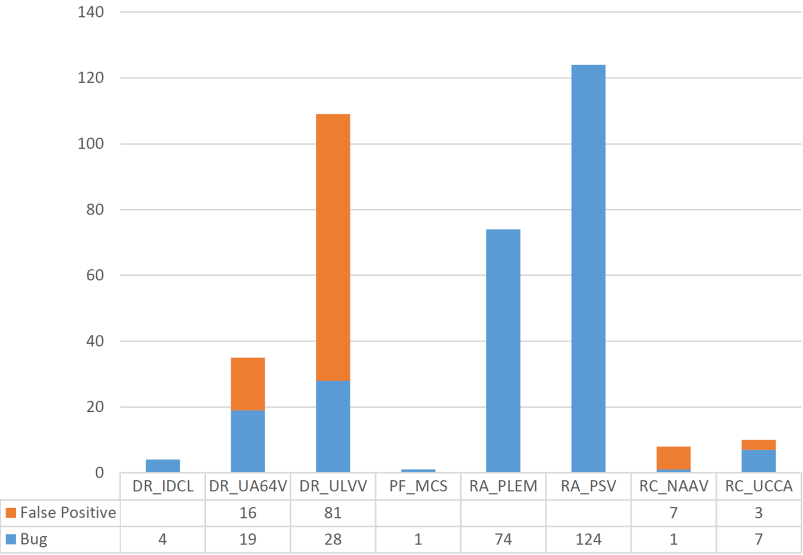
The variable 'running' used as a condition may not see upda

Just-in-Time Analysis in Visual Studio

This thesis presents a catalog of concurrency-related bug patterns. In contrast to design patterns, bug patterns describe specific code constellations that are error-prone. Each pattern is introduced with a brief description of the possible concurrency issues that may occur and what changes are necessary to avoid them. Moreover, the patterns have a short explanation of how their presence can be observed.

Concurrency Bug Finder is a Roslyn-based static code analysis tool for the introduced bug patterns. The underlying core idea of the analyses is: «Code made to run concurrently will run concurrently.» This presumption removes the necessity to identify if a code is truly executed concurrently. Instead, the concurrent execution of the code is assumed if synchronization primitives are used.

The experimental evaluation verified the reliability of the different bug pattern analyses by scanning various projects with Concurrency Bug Finder and manually assessing the correctness of the reported issues. It revealed that more than 36% of the 365 findings are malign and only 29% are false positives. The remaining 35% indicate code locations that could be improved with regard to the design.



	DR_IDCL	DR_UA64V	DR_ULVV	PF_MCS	RA_PLEM	RA_PSV	RC_NAAV	RC_UCCA
🟧 False Positive		16	81				7	3
🟦 Bug	4	19	28	1	74	124	1	7

Results of the Experimental Evaluation per Pattern