



DISSERTATION DEFENSE

Madeline Endres



Three Lenses on Improving Programmer Productivity: From Anecdote to Evidence

Tuesday, July 16, 2024

3:30pm – 5:30pm

4901 Beyster / Hybrid – [Zoom](#)

ABSTRACT: "Can we use neurosymbolic insights to help programmers write more correct code faster? Can reading technical material make students better at programming? Do professional developers think cannabis use leads to more creative programs?" Leveraging core software engineering techniques with insights from psychology and medicine, this talk presents a series of algorithms and theoretically-grounded interventions that enhance programmer productivity. By combining large-scale exploratory empirical investigations and controlled human-focused experimental design, we both build mathematical models of the impact of understudied features on programmer productivity and also provide actionable and evidence-backed interventions that improve productivity in practice.

This talk presents findings from three primary lenses: developing efficient and usable bug-fixing tools for non-traditional novices, designing effective programming training informed by objective measures of programming cognition, and understanding the impact of external factors, such as psychoactive substance use. Regarding developing efficient and usable programming tools, this dissertation proposes two novel methods of bug-fixing support targeting parse-errors and input-related bugs. Both are error types that we identify as commonly-encountered by non-traditional novice programmers (e.g., those learning without the support of the traditional classroom) but are overlooked by existing program-repair tools. Next, to help novice programmers become more like experts faster, we propose a model of novice programming expertise using neuroimaging (fNIRS). We leverage our cognitive findings to design and evaluation a novel supplemental reading training that improves programming outcomes. Finally, we argue that external factors also impact software productivity, including those anecdotally-reported but understudied by the scientific literature. In this dissertation, we study the impact of one such factor: psychoactive substance use. This talk presents the first survey of the prevalence of such substances in software and also proposes a mathematical model of the true impact of one such substance, cannabis, on programming ability.

CHAIR: Prof. Westley Weimer