

Code Safety: C and Java programs

Description	Prove or disprove code is safe. Three types of safety are considered: memory, thread, and type safety. Examples are provided. Cryptol specs are created to prove safety or find a counterexample which reveals inputs that lead to unsafe operation.
Purpose	Safety is extremely important as attackers will try to exploit unsafe code. Tools for proving safety or finding unsafe code are extremely important.
Audience	This module is intended for: <ol style="list-style-type: none">1 The general public2 K-12 and college classes on cyber defense3 preparation for proficiency in the use of tools and a computing environment suitable for the study of cyber defense
Objectives	After completing the module: <ol style="list-style-type: none">1 Learners will understand how code can be written to be unsafe2 Learners will be able to spot unsafe code in several common cases3 Learners will use SAW to identify safe and unsafe code
Keywords	Memory safety, thread safety, type safety, SAW, Cryptol
Category	cybersecurity > education
Delivery	java applets and written documentation in pdf format
Team	John Franco and Ethan Link
Assessment	The applets provide the means for experimentation. Questions are asked in the documentation that help with the set up of experiments. The ideas that learners come up with is evidence that the module was successful.
Workflow	No particular schedule was established
Environment	All materials are contained in a single jar file. The jar file can be run on any computer where java version 14 or higher and some pdf reader such as acroread or evince are available. The jar file may be executed in the cyber range or learners may download the jar file (which is considered to be an executable file) and run it on their personal computers.