Applying Genetic Programming to Bytecode and Assembly

Eric Collom

Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA

29 April '14,
UMM Senior Seminar

The big picture

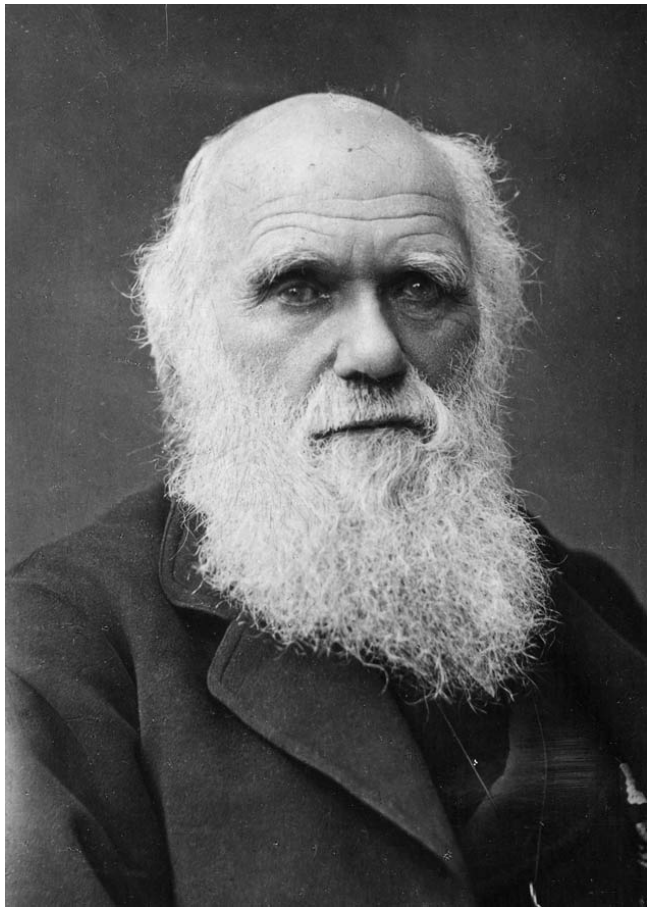
- Evolving whole programs is hard to do with source code.
- Evolving whole programs with bytecode and assembly is not as hard.

Outline

Outline

What is Evolutionary Computation?

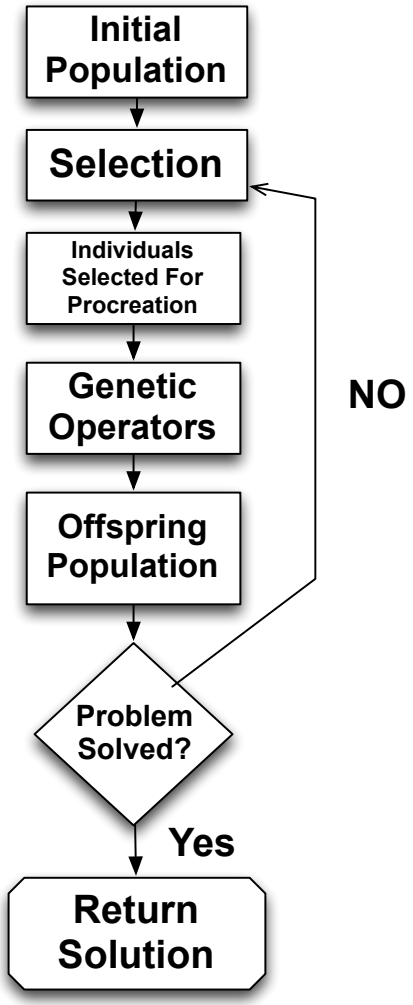
- EC is a a technique that is used to automate computer problem solving.
- Loosely emulates evolutionary biology.



Charles Darwin
http://tinyurl.com/lqwj3wt

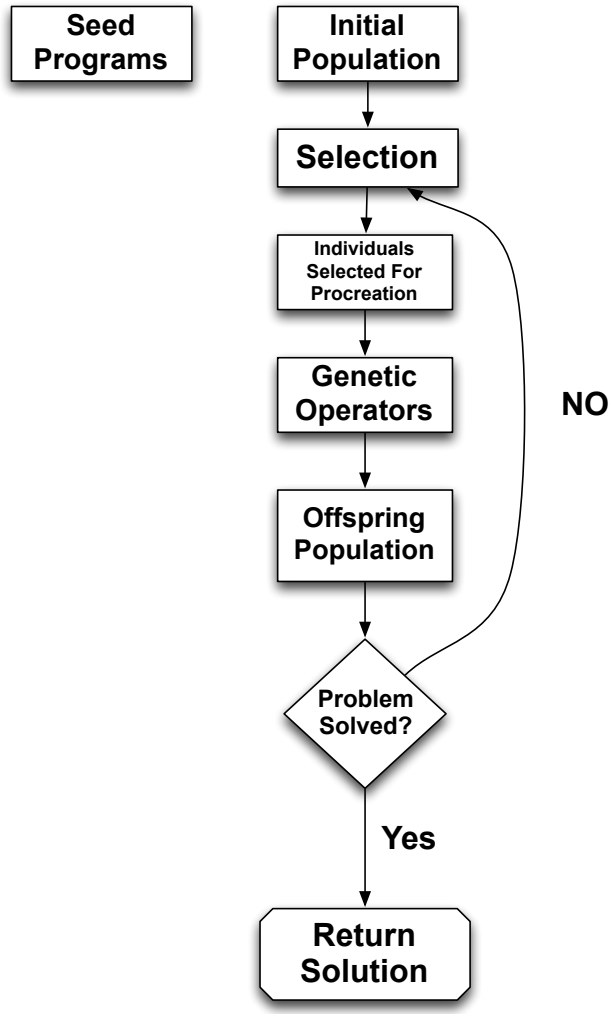
How does it work

- Continuous Optimization
- Selection is driven by the *fitness* of individuals
- Genetic Operators mimic sexual reproduction and mutation



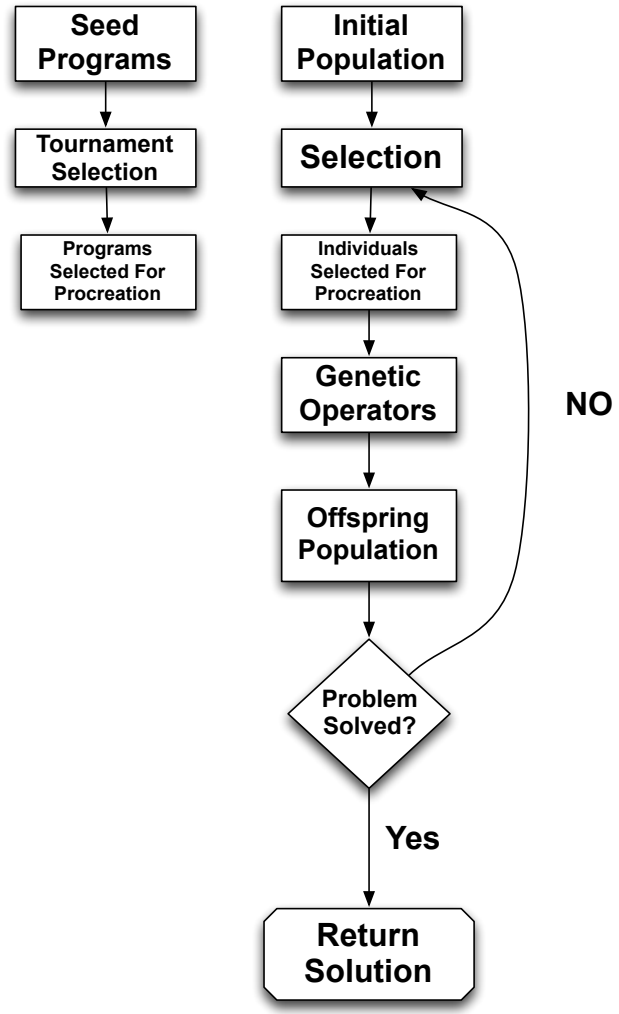
Genetic Programming

- Uses the EC technique to evolve programs
- The population is programs



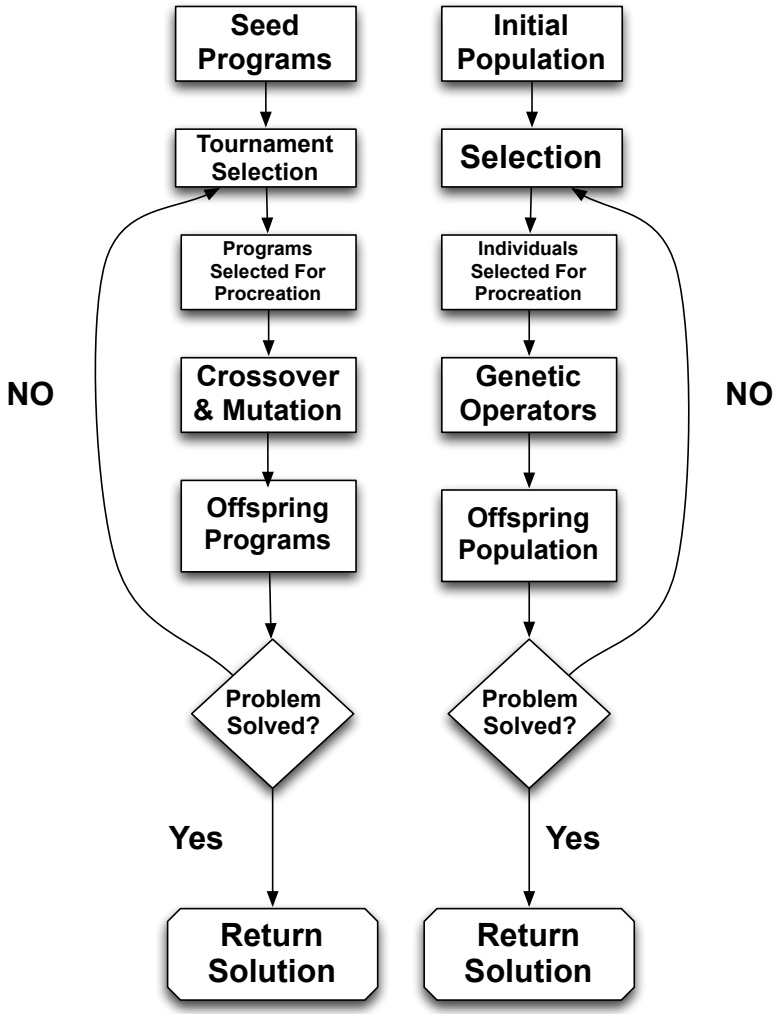
Genetic Programming

- Tournament Selection

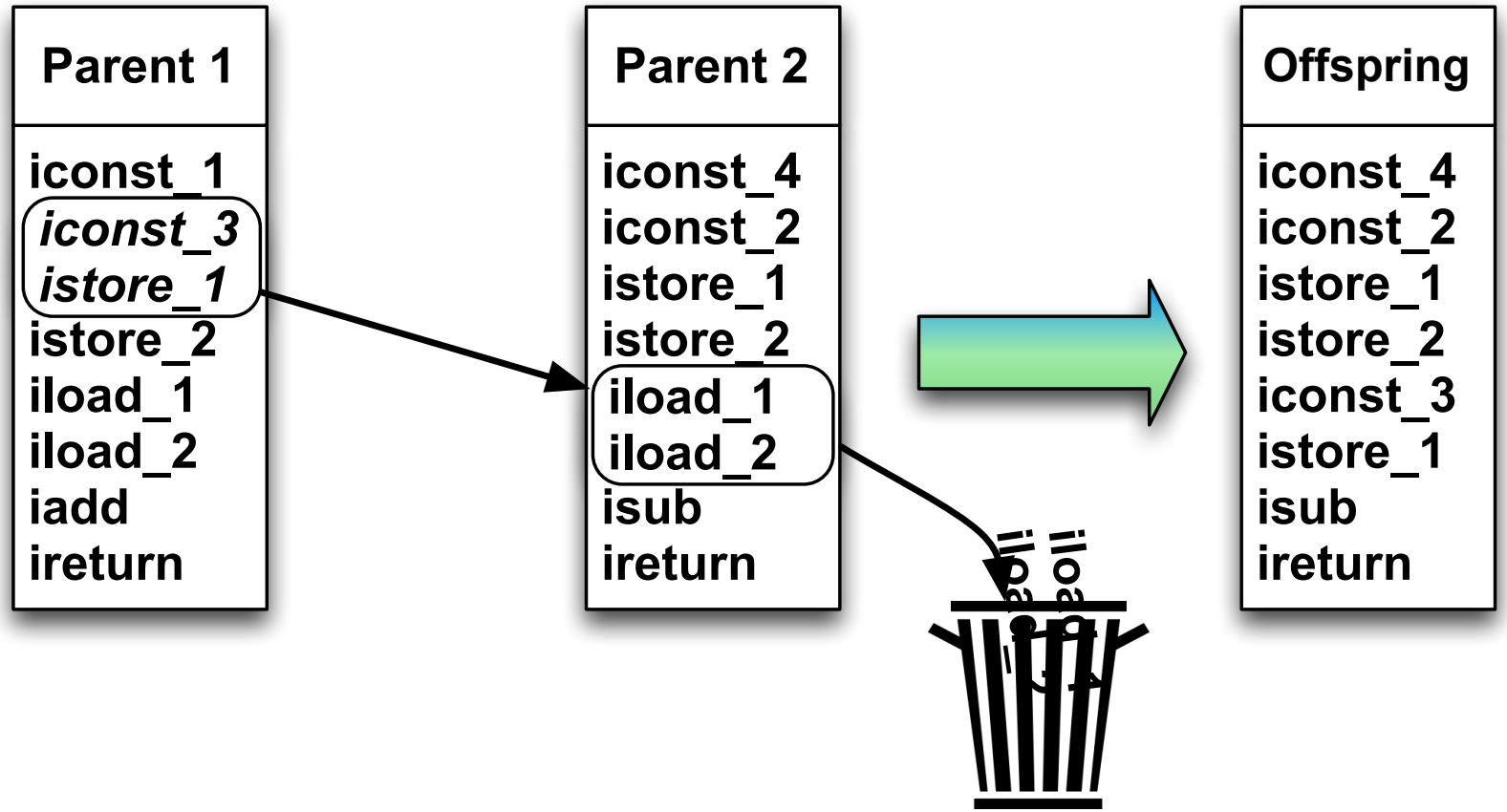


Genetic Programming

- Crossover
- Mutation

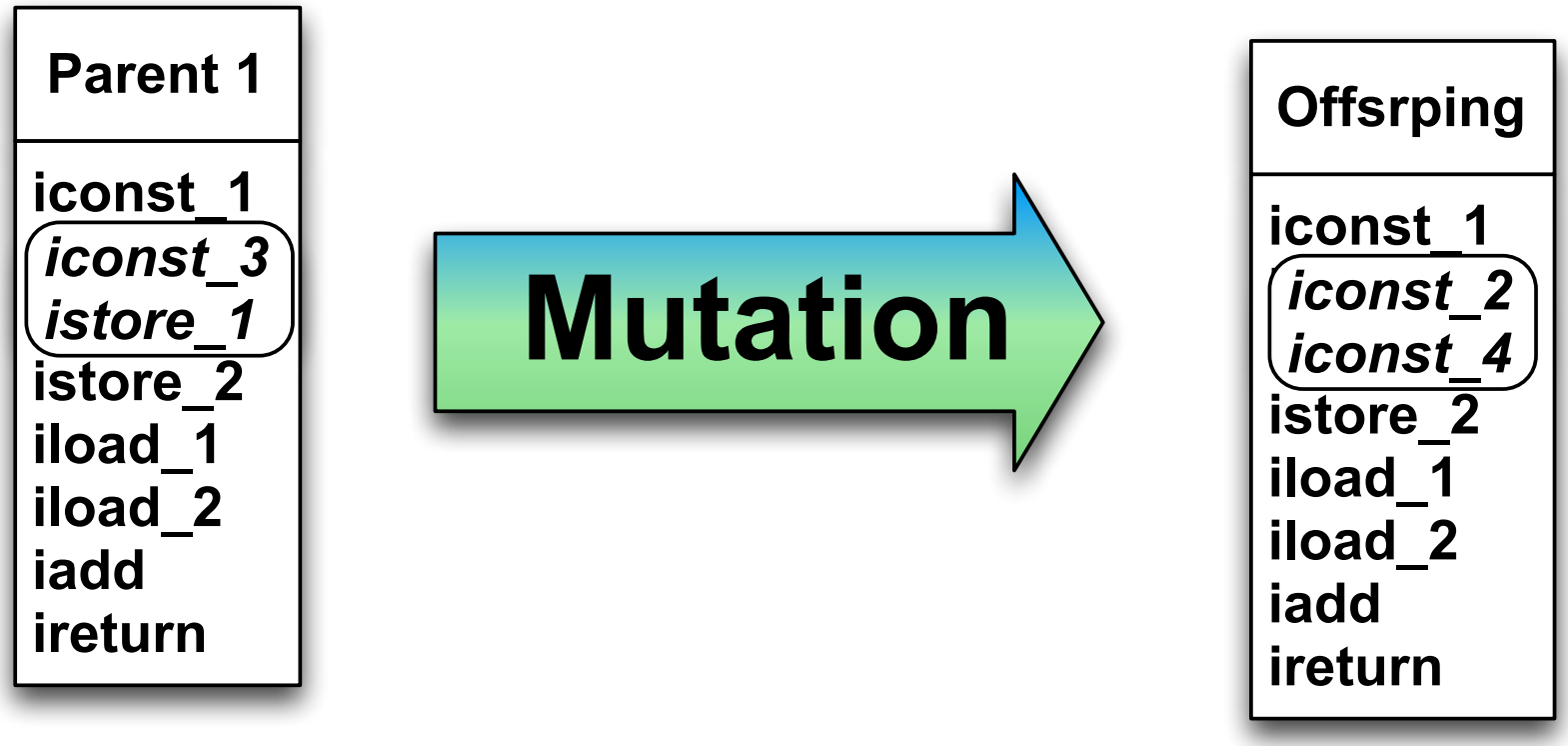


Crossover



Crossover with Java Bytecode

Mutation



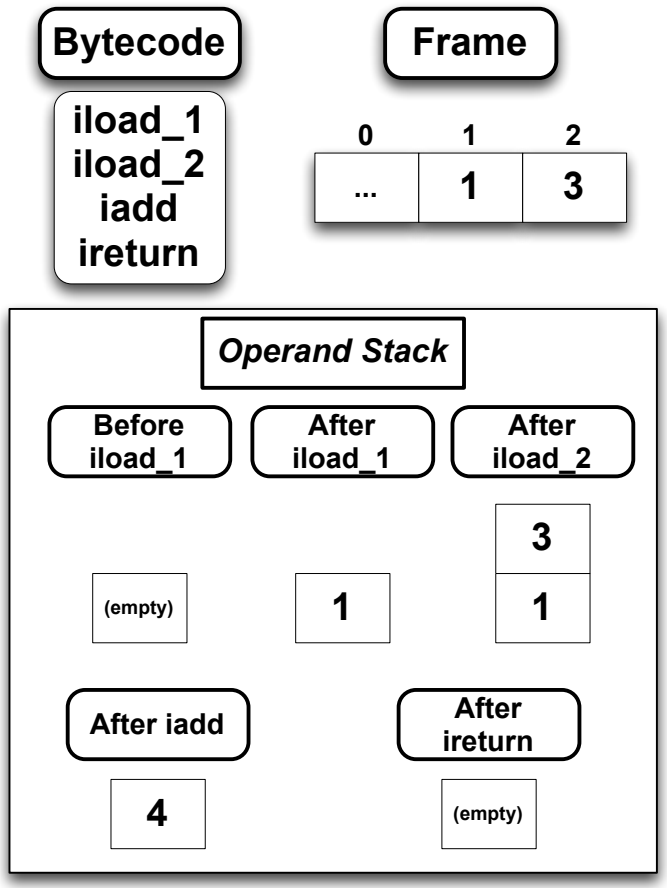
Crossover with Java Bytecode

Java Virtual Machine

- Frames
- Array of local variables
- Operand Stack

Java Bytcode and Frames

- Opcodes
- Prefix indicates type



Outline

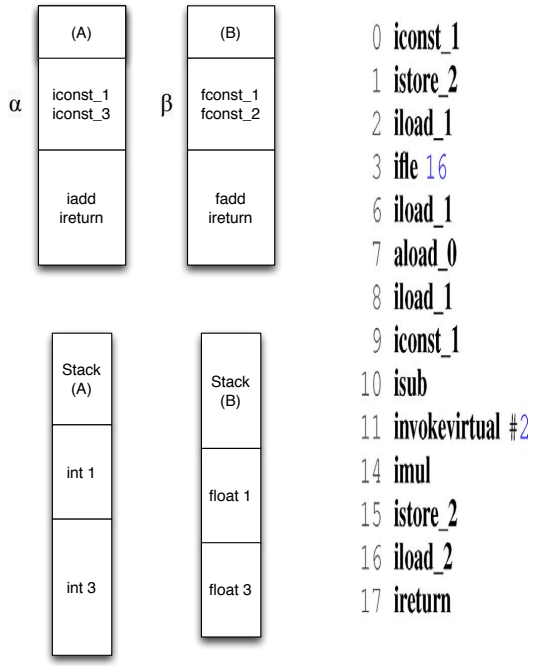
Source Code Constraints

Flexibility

Outline

Selecting Offspring

- There is still a chance to produce non-compilable code
- Solution: Add restrictions to code selection.
- Stack and Frame Depth
- Variable Types
- Control Flow



Crossover

Non-Halting Offspring

Outline

Selecting Offspring

Genetic Operators

Non-Halting Offspring

Outline

Conclusions

References