

HUMAN-DRIVEN GENETIC PROGRAMMING FOR PROGRAM SYNTHESIS: A PROTOTYPE

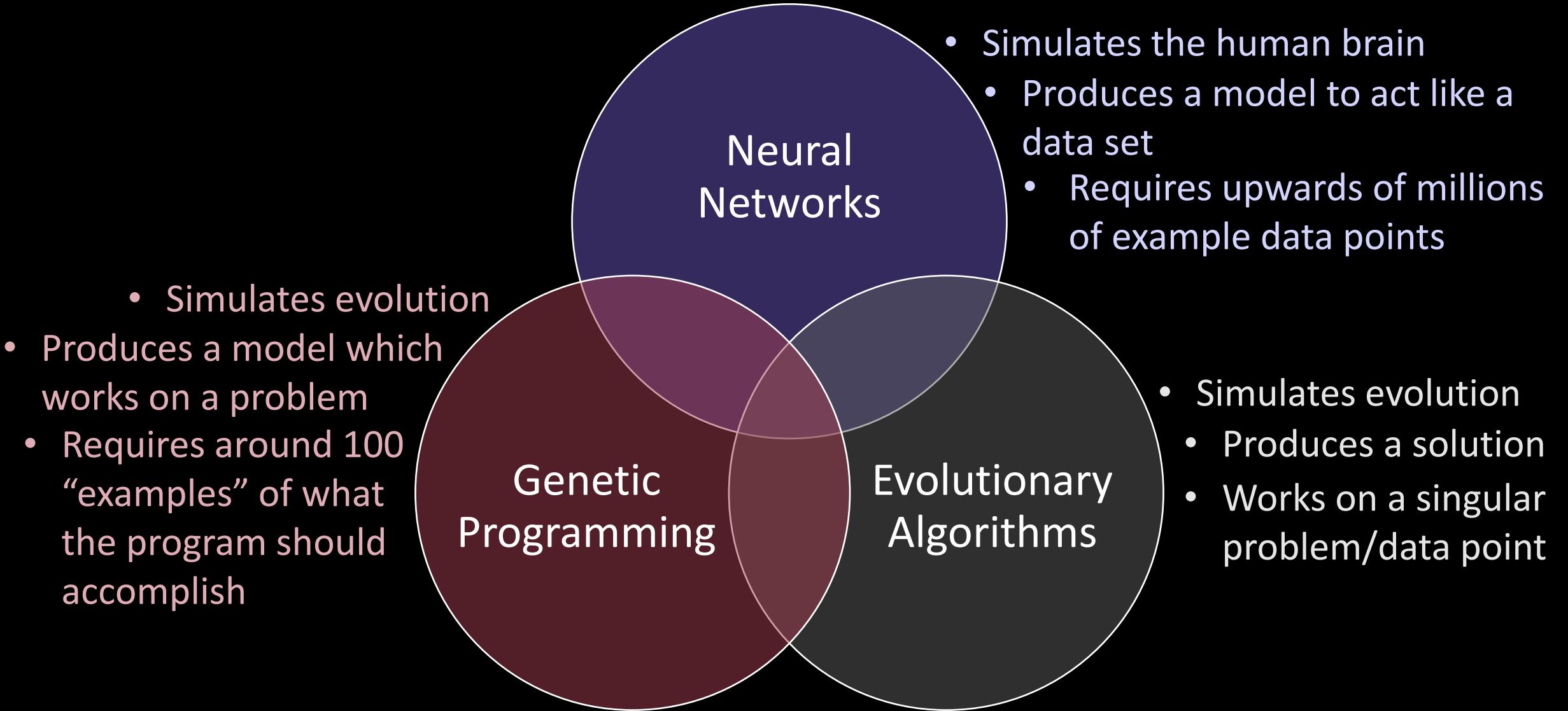


PROF. THOMAS HELMUTH, JAMES GUNDER FRAZIER, YUHAN
SHI, AHMED FARGHALI ABDELREHIM



11/04/2023

Hamilton



A COMPARISON OF GENETIC PROGRAMMING TO OTHER POPULAR
FORMS OF ARTIFICIAL INTELLIGENCE

Recent advances in genetic programming allow it to **generate** general purpose programs similar to those humans write, but **require specifications** in the form of extensive, labeled training data, a barrier to using it for user-driven synthesis, even for experts. Here we will describe the novel **prototype** of a *Human-Driven Genetic Programming* (HDGP) system which, by utilizing *Counterexample-Driven GP (CDGP)*, reduces the required number of initial training cases from one-hundred (100) to only five (5) and simplifies the initialization of genetic programming runs.

PROBLEM STATEMENT

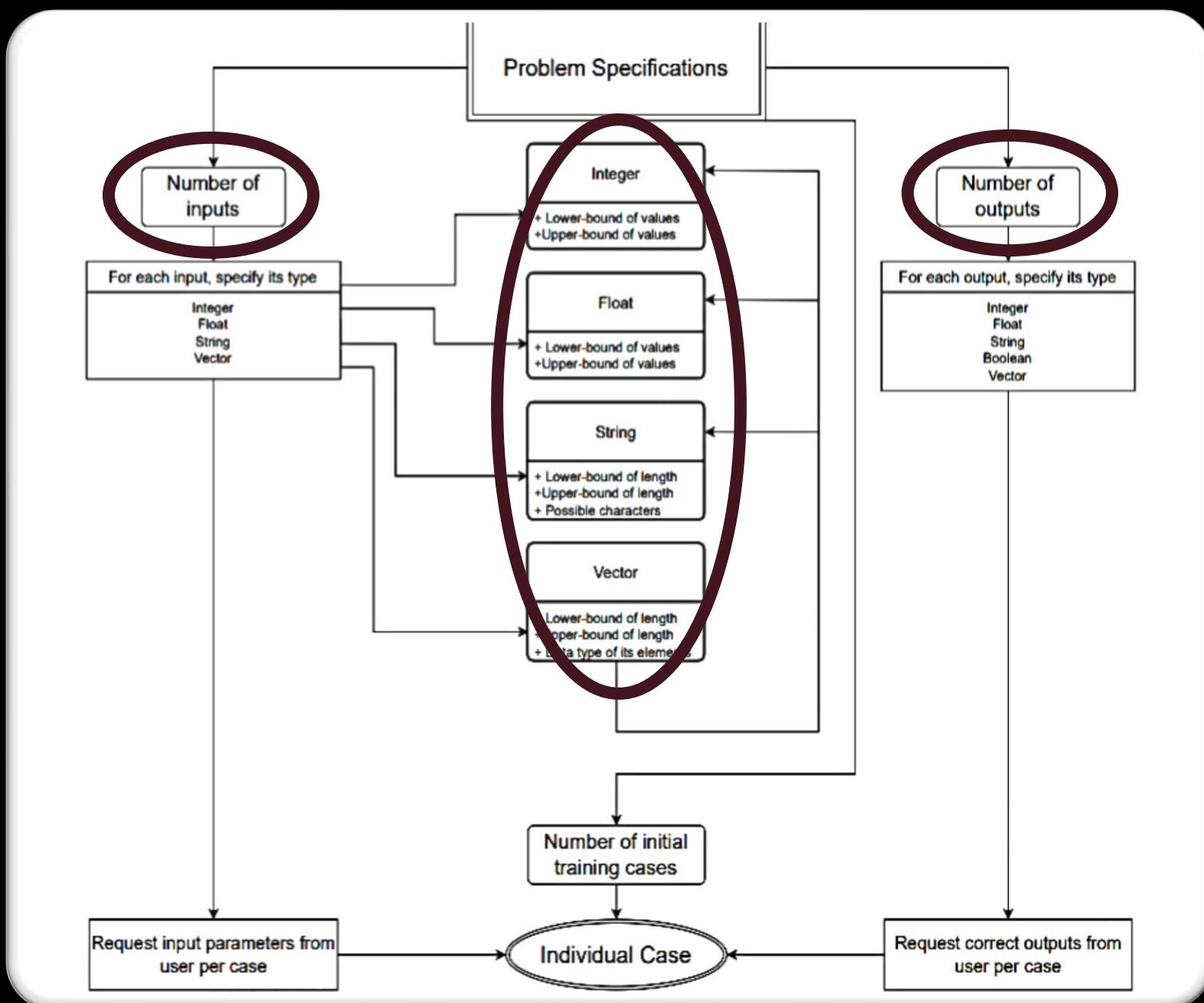
```
from-user
ow many outputs there can be and then prompts user to
le output
th each element as a Clojush friendly keyword that c
fied data type"
process-user-input "
e are: " :integer)]
t 1]
ount num-outputs)
outputs (acquire-output-type-from-user output-c
tput-count))
-----
; functions are responsible for generating
it Parameters (it is not responsible for ge
however.)
.
y integer parameter based off of boundr
map which is the parameter data type
```

ALGORITHM OVERVIEW

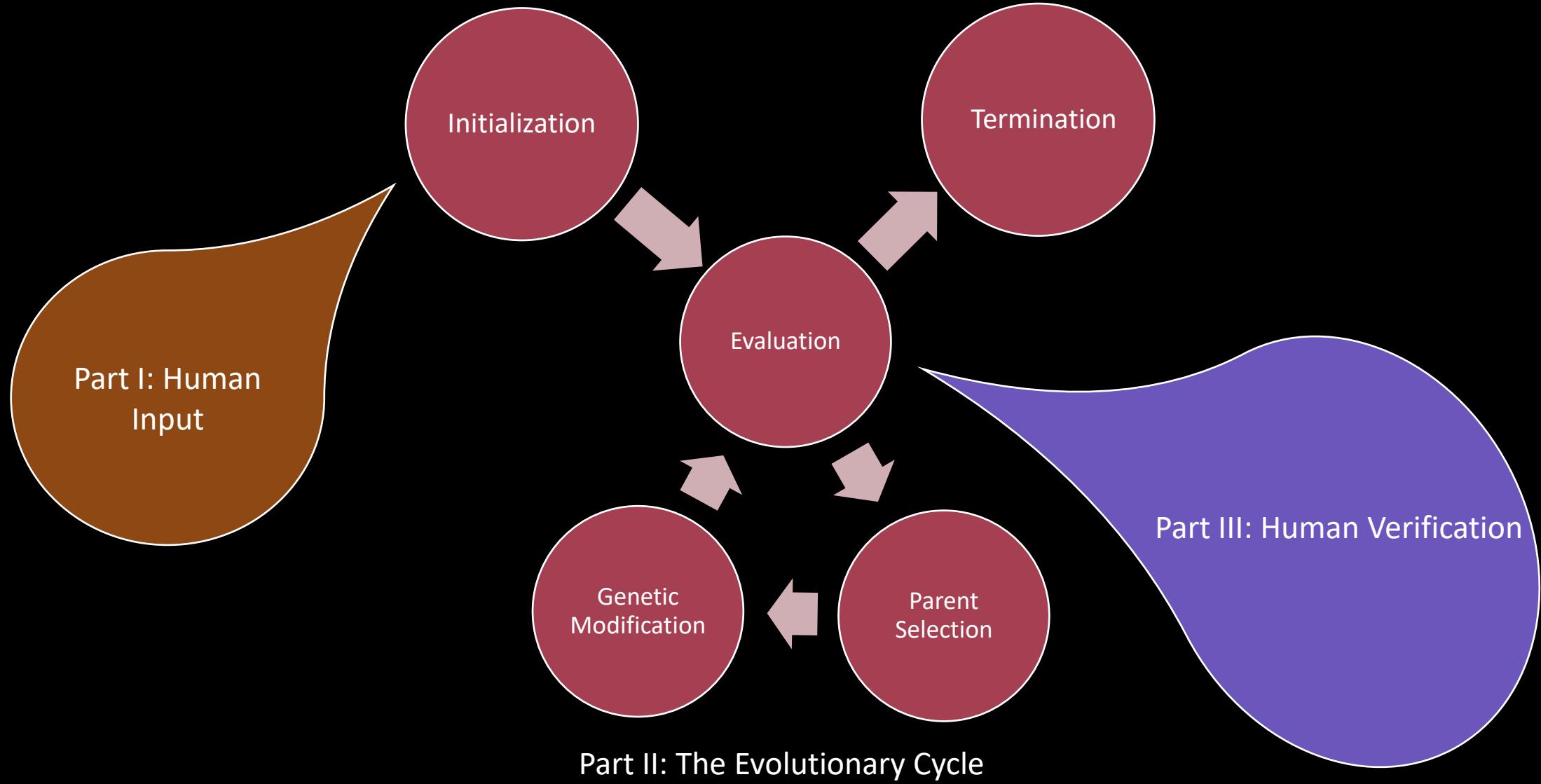
THREE PARTS TO MAKE THE WHOLE

- 1.Human Input
- 2.The Evolutionary Cycle
- 3.Human Verification

Image Source: <https://github.com/jgfrazie/Clojush>



(Source: Helmuth et al., 2023)



Case 0 : Input: [0] ; Program's output: ["0"]
Case 1 : Input: [10000] ; Program's output: ["10000"]
Case 2 : Input: [8142] ; Program's output: ["Fizz"]
Case 3 : Input: [5581] ; Program's output: ["5581"]
Case 4 : Input: [4580] ; Program's output: ["4580"]
Case 5 : Input: [2797] ; Program's output: ["2797"]
Case 6 : Input: [90] ; Program's output: ["Fizz"]
Case 7 : Input: [6589] ; Program's output: ["6589"]
Case 8 : Input: [7369] ; Program's output: ["7369"]
Case 9 : Input: [3652] ; Program's output: ["3652"]
Case 10 : Input: [6988] ; Program's output: ["6988"]

Caption: Outputs from a best program on the Fizz problem: a simplified version of the FizzBuzz problem

Case 2 : Input: [8142] ; Program's output: ["Fizz"]

EXPERIMENTS AND RESULTS

Problem	Initial Cases	Average Number of Human Verified Cases as Examples	Found a Solution
Subtract Two Integers	5	0	<input checked="" type="checkbox"/>
Spaces	5	26	<input checked="" type="checkbox"/>
Substring	5	8	<input checked="" type="checkbox"/>
Element Rank	5	0	
Fizz	5	17	<input checked="" type="checkbox"/>

```

cases if not a
-driven-case-generators counterexample
d counterexample-driven-fitness-threshold-for
oracle-function] :as argmap}]

that passes all of the training cases! ***")
check if the best program works on some new input

rate-counterexample-type % best argmap) ;; get
example-driven-case-generators))]

PROGRAM:" best)
s (map first (interesting/run-best-on-all-cas
se counterexample-driven-case-checker
automatic (counterexample-check-results-autom
    all-cases best-results-on-all-case
human (counterexample-check-results-human
    all-cases best-results-on-all-cases ou
simulated-human (counterexample-check-results
    all-cases best-results-on-all-cases ou
(if (keyw

```

INSIGHTS AND FUTURE WORK

HUMANS IN THE LOOP

No system has used a human to drive the evolutionary cycle of a GP run before.

LESS SPECIFICATIONS AND CASES NEEDED

While others have eliminated specification needs, we have reduced specifications to purely user inputs and outputs. We also reduced the number of training cases needed to start a run from 100 to 5 by utilizing informal Counter-example Driven GP (iCDGP).

NEWFOUND LIMITATIONS

Issues never considered have been revealed through our prototype. For example, how can a user specify constraints for a problem beyond data type? (i.e., “This array must contain at least one zero”)

Image Source: <https://github.com/jgfrazie/Clojush>

CONTACT AND ACKNOWLEDGEMENTS



James G. Frazier

Computer Science | Senior Fellow
Hamilton College

✉ jgfrazie@hamilton.edu

🌐 <https://jgfrazie.github.io>

🌐 <https://mochiresearch.com/author/wchtr101/>

🌐 <https://lspector.github.io/push.html>



Prof. Thomas Helmuth

Assistant Professor of Computer Science
Hamilton College

✉ thelmuth@hamilton.edu

🌐 <https://cs.hamilton.edu/~thelmuth/>

🌐 <https://lspector.github.io/push.html>

Special thanks to the following:

- Prof. Sarah Morrison-Smith
- Yifan Wu
- Emily Weinstein



PUSH*lab
programs under selection + heredity