Visualizing the Turing Tarpit

Jason Hemann Eric Holk
Indiana University

FARM ‘13

Epigrams on Programming

by

Alan J. Perlis

54. Beware of the Turing tar-pit in which everything is possible but nothing of interest is easy.

B

Y

I

λx.x

C

K

λxy.x

Ω

S

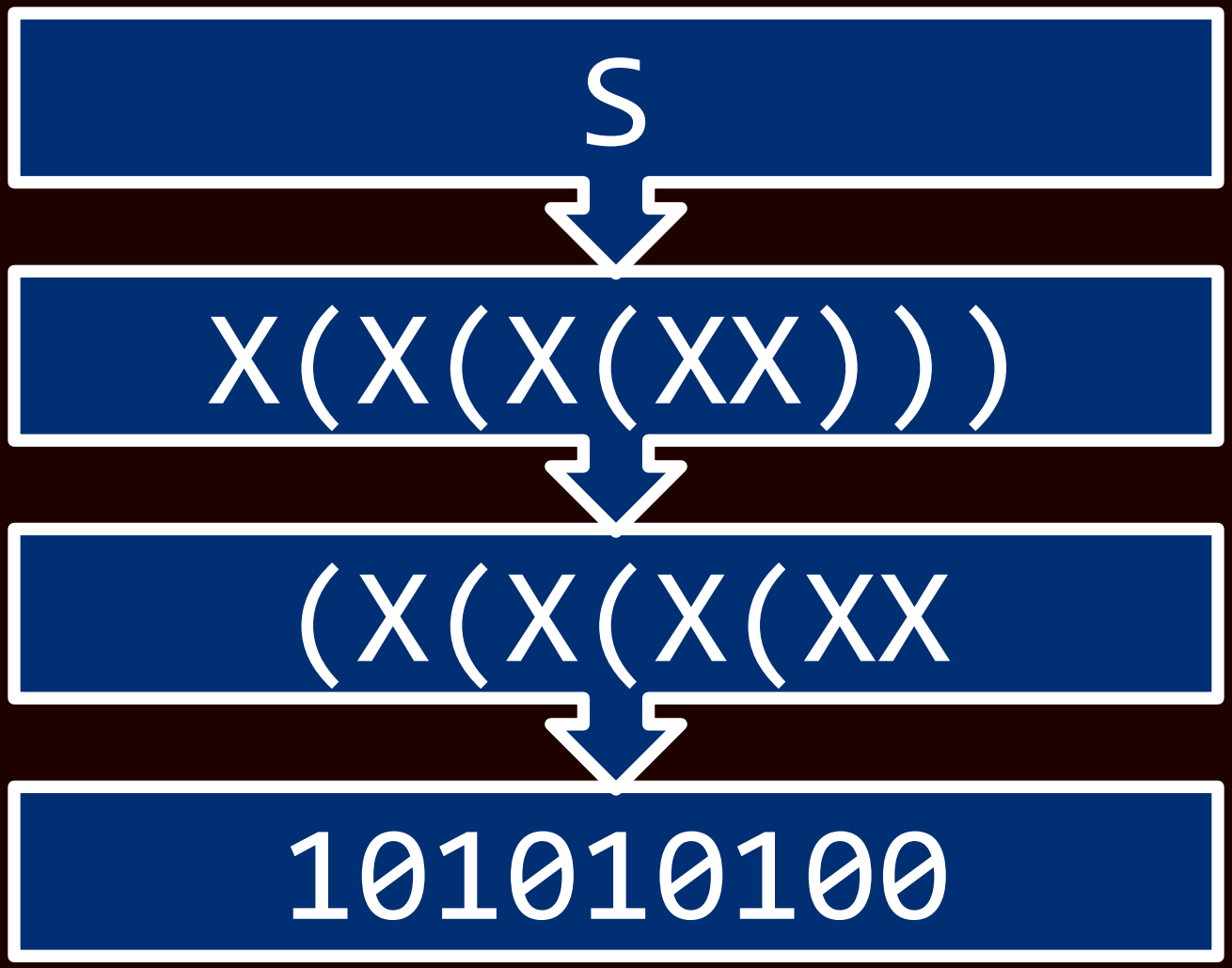
λxyz.xz(yz)

X

λx.xSK

(X(X(XX))) ⇒ K

(XK) ⇒ S

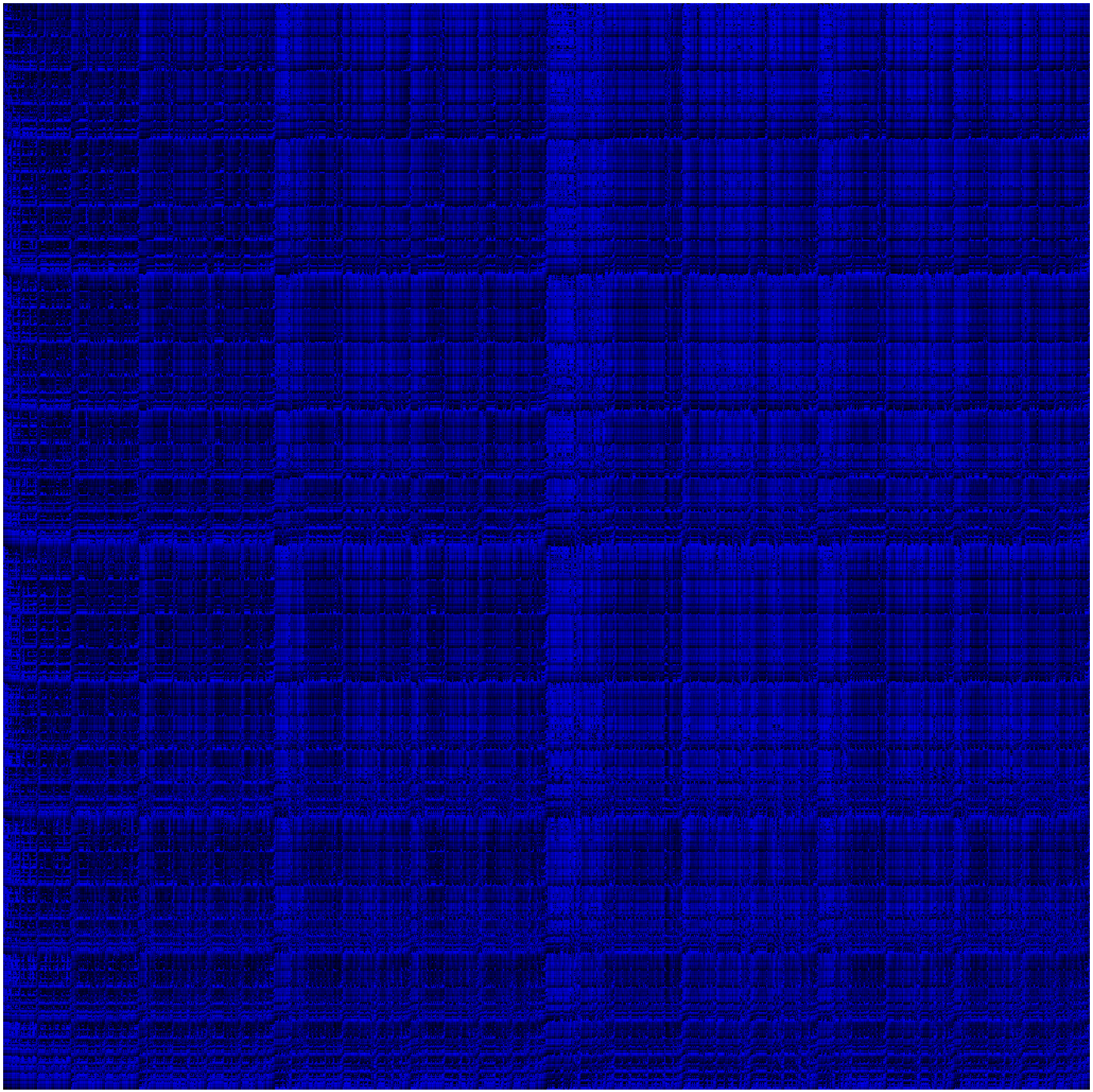

S
X(X(X(XX)))
(X(X(X(XX
101010100

Syntax	Semantics
[F] ⇒ ε	λx.x
⇒ [F]0	[F]SK
⇒ [F]1	λxy.[F](xy)

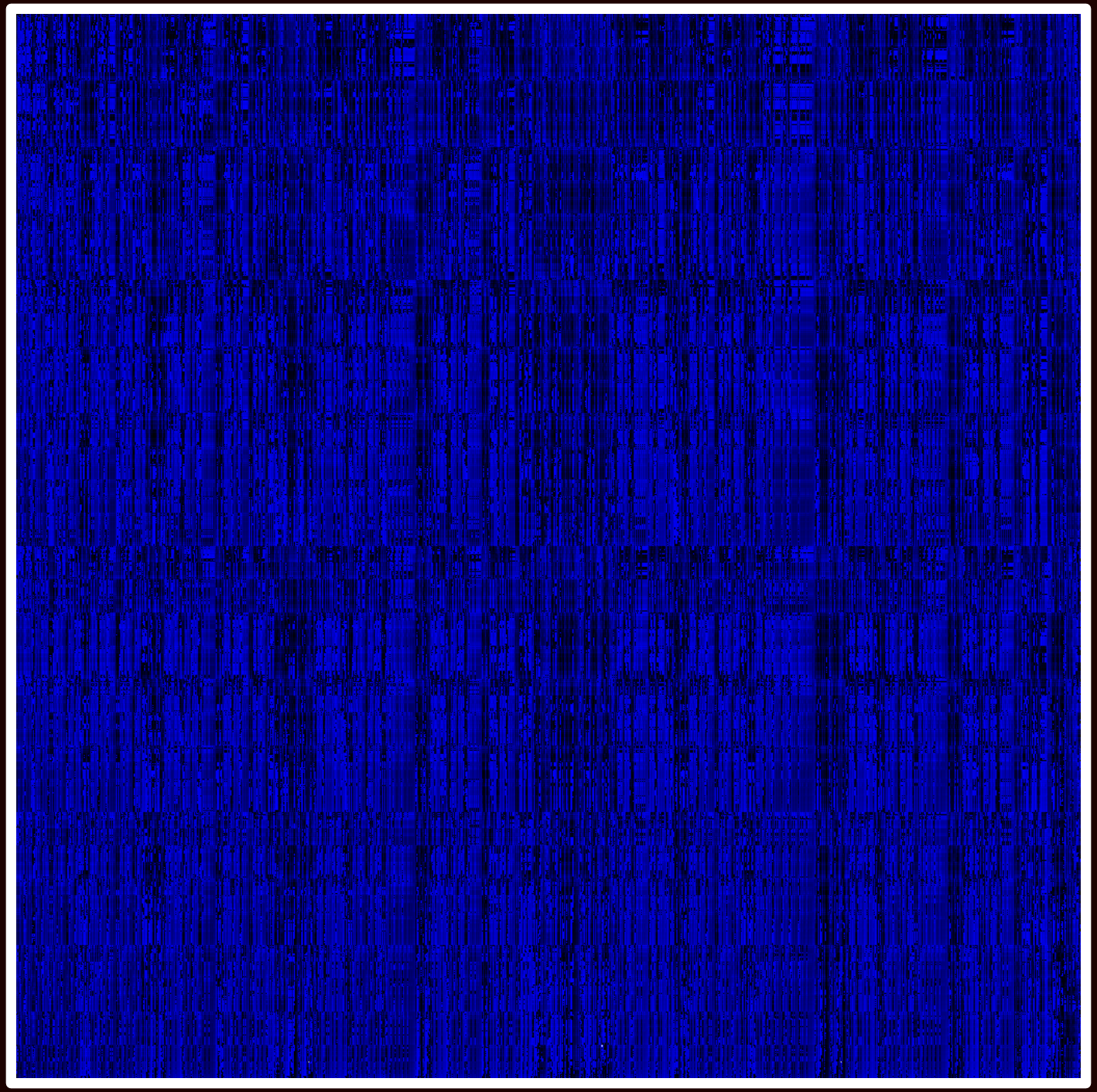
Every binary sequence is a valid program

A Gödel numbering of $CL_{\{SK\}}$ expressions

$CL_{\{SK\}}$		Jot
[S]	⇒	11111000
[K]	⇒	11100
[AB]	⇒	1[A][B]

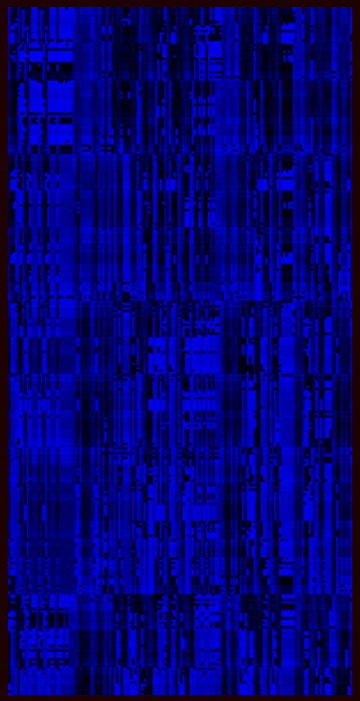


bv
to
WNF

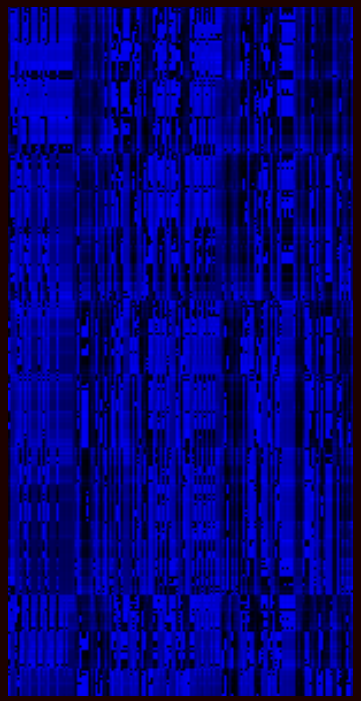


ao
to
NF

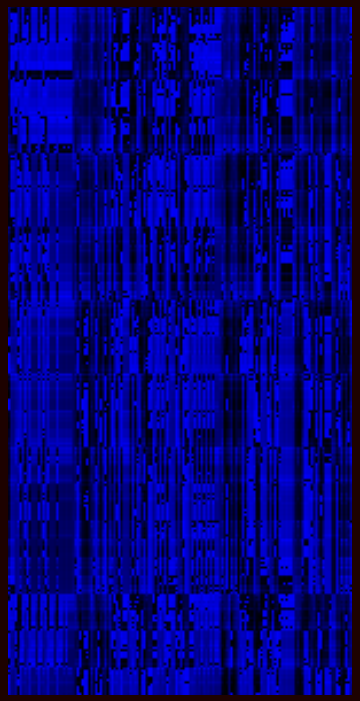
		Reduce under Abstraction	
		Yes	No
Reduce Args	Yes	Normal Form **ao**	Weak Normal Form **bv**
	No	Head Normal Form **he**	Weak Head Normal Form **bn**



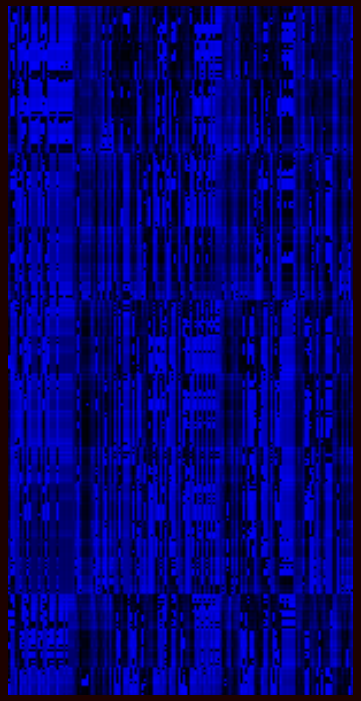
ao
NF
sec.



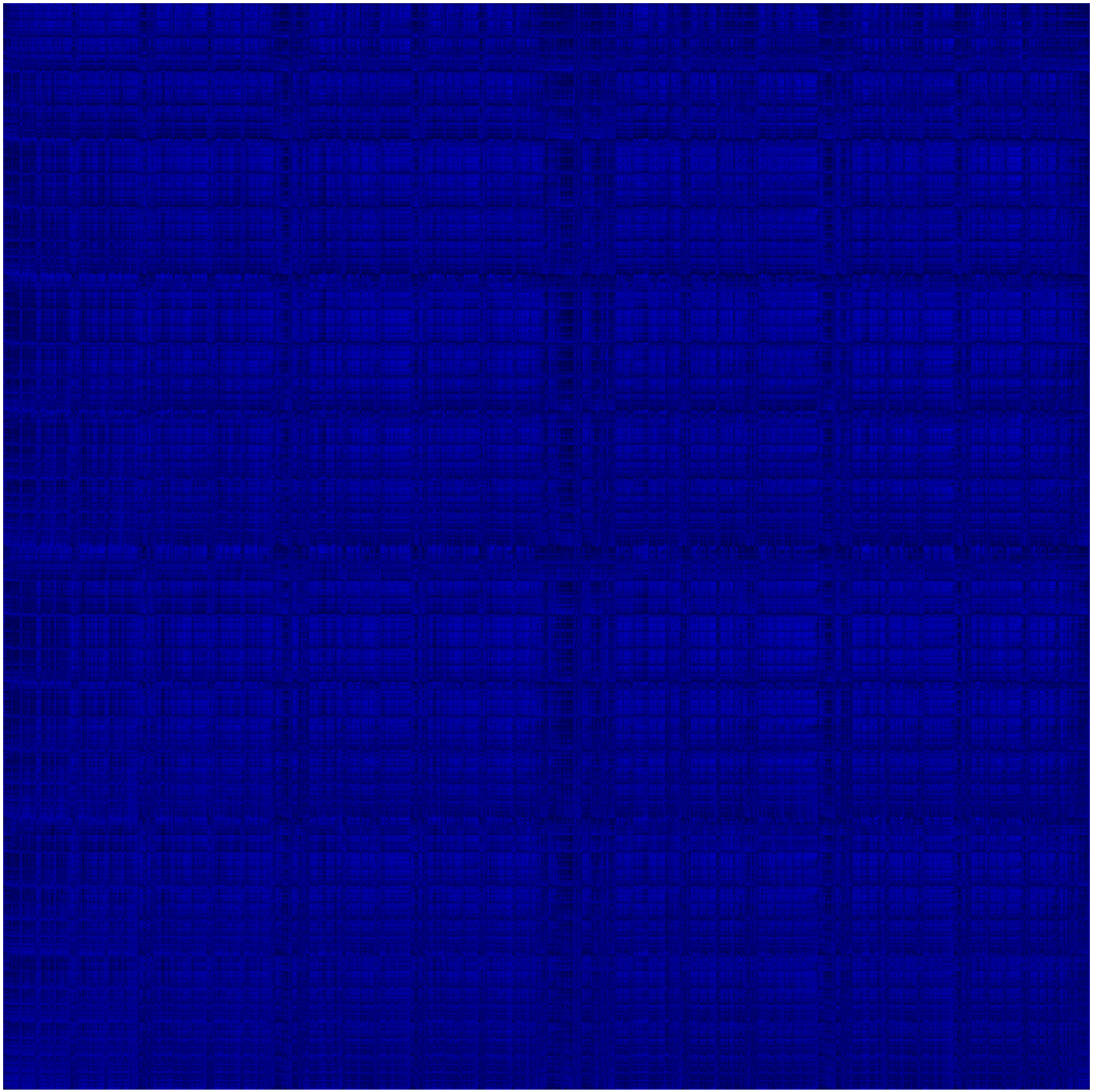
no
NF
sec.



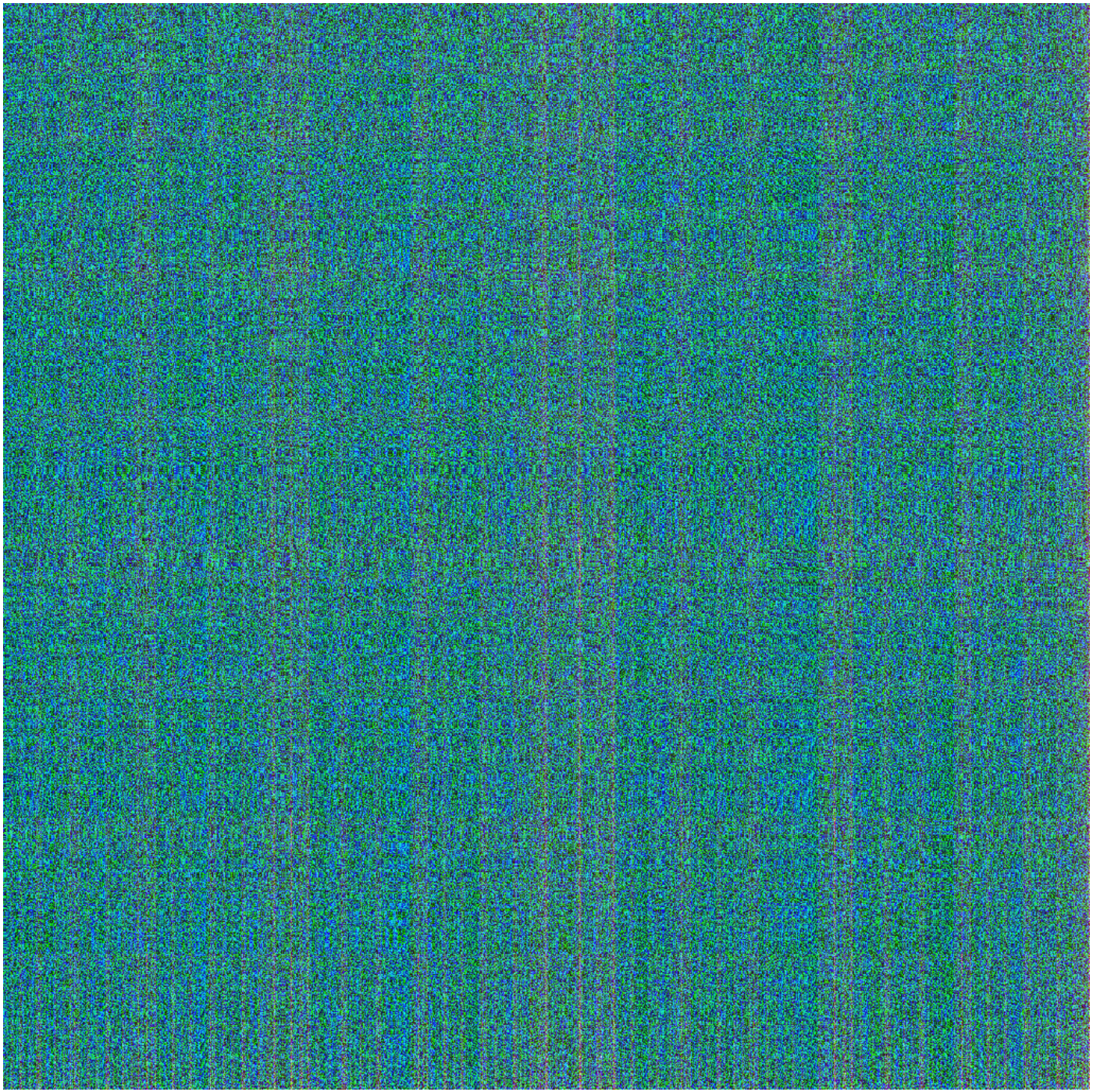
ha
NF
sec.



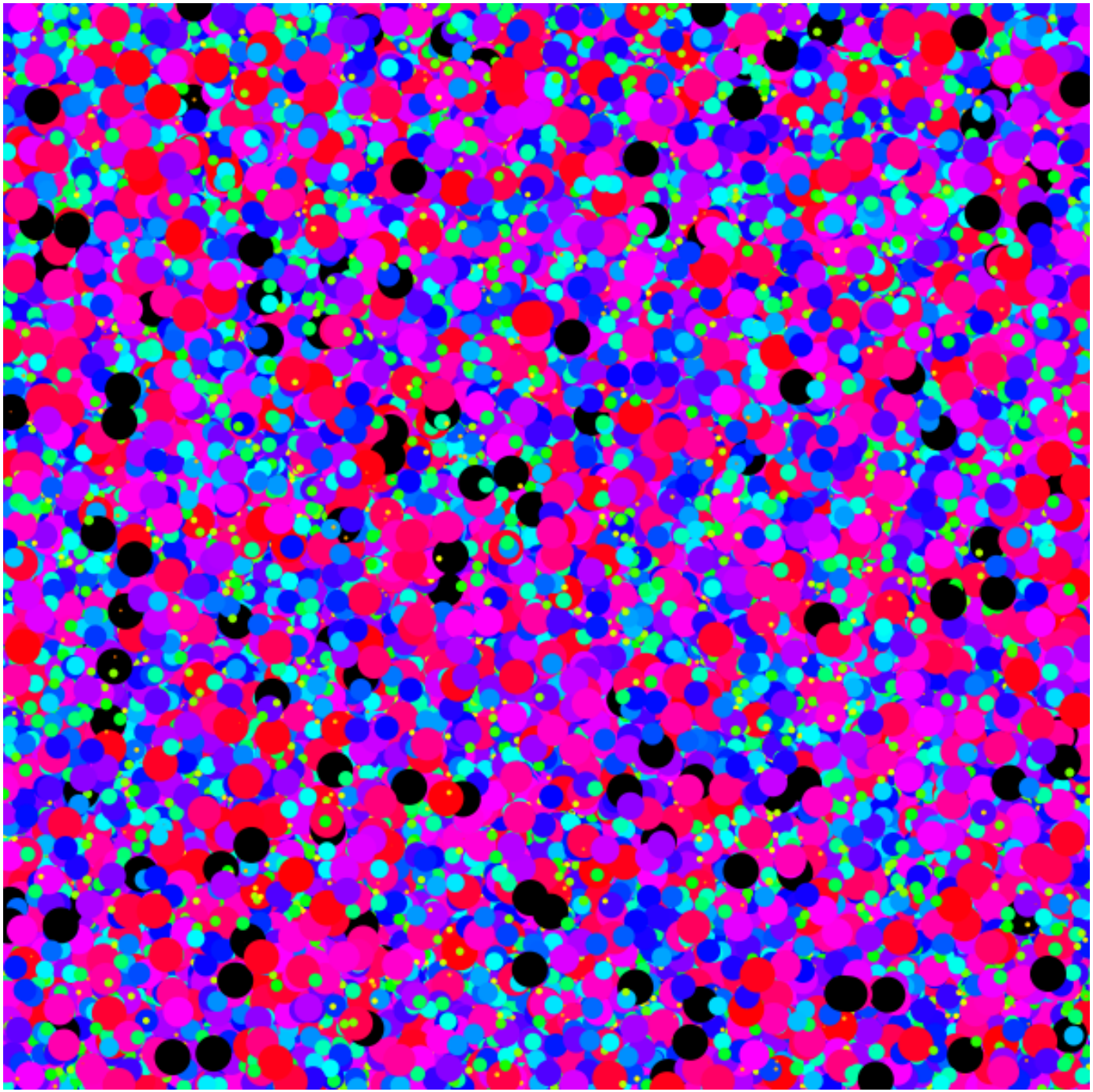
hn
NF
sec.



16M
bn
WHNF



bv
WNF
Adj.



bn
Rand
~1m

DEMO

- More programs
- More reductions
- Different methods
- More languages

Related Work

Barker, C. Iota and Jot: the simplest languages?. URL http://semarch.Linguistics.fas.nyu.edu/barker/Iota/

Sestoft, P. Demonstrating Lambda Calculus Reduction *The Essence of Computation*. pp. 420-435

Questions?

http://tarpit.github.io/TarpitGazer/

Code Gallery Paper