Experimentação em Engenharia de Software

On the Performance of the Python Language

PG 54232
PG 55972
PG 57539

Content

- Introduction
- Python Pool
  - Chosen criteria
  - Interpreted versions
  - Python compilers
- Object of Research
- Methodology
- Test Suitcase
  - Fibonacci
  - Primes
  - Sorting algorithms
  - Data querying
- Authors Notes
  - Threats to validity
  - Future work
  - Conclusion

Introduction

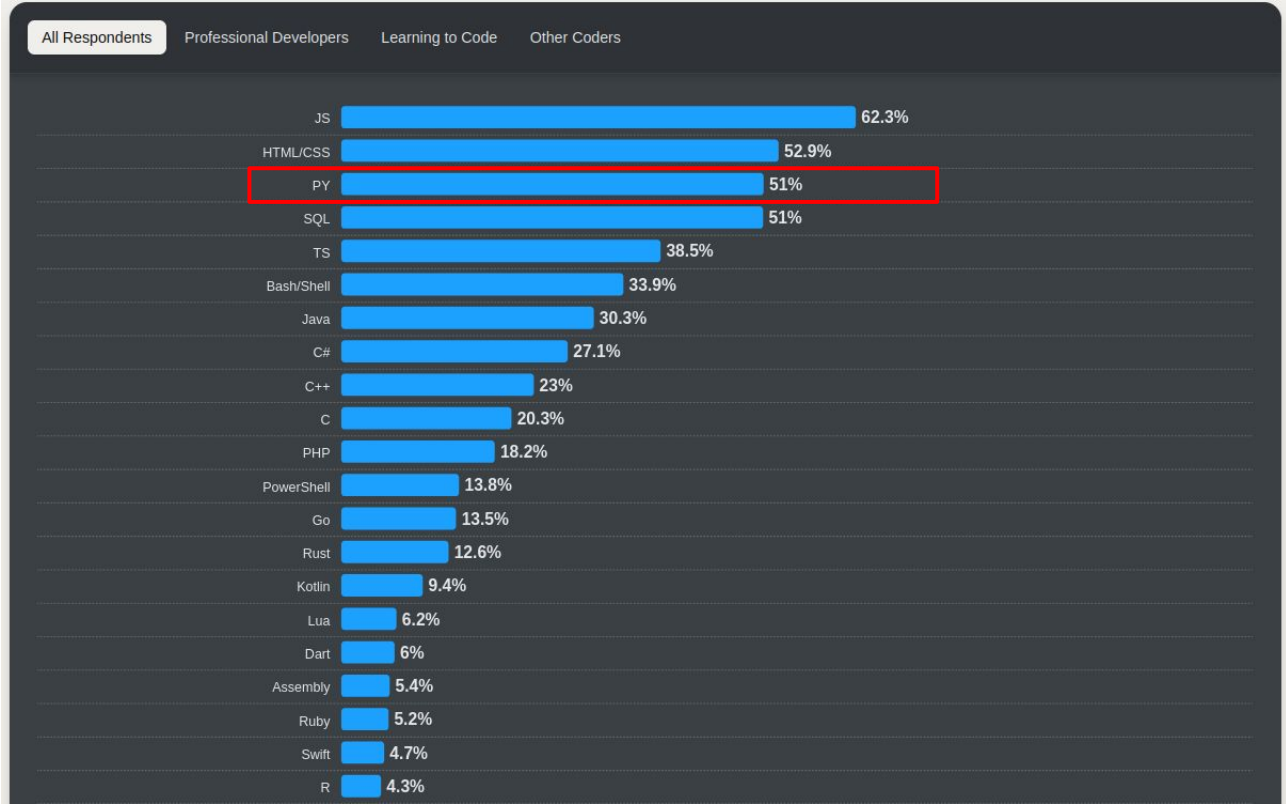
Python Performance

Total

	Energy		Time		Mb
(c) C	1.00	(c) C	1.00	(c) Pascal	1.00
(c) Rust	1.03	(c) Rust	1.04	(c) Go	1.05
(c) C++	1.34	(c) C++	1.56	(c) C	1.17
(c) Ada	1.70	(c) Ada	1.85	(c) Fortran	1.24
(v) Java	1.98	(v) Java	1.89	(c) C++	1.34
(c) Pascal	2.14	(c) Chapel	2.14	(c) Ada	1.47
(c) Chapel	2.18	(c) Go	2.83	(c) Rust	1.54
(v) Lisp	2.27	(c) Pascal	3.02	(v) Lisp	1.92
(c) Ocaml	2.40	(c) Ocaml	3.09	(c) Haskell	2.45
(c) Fortran	2.52	(v) C#	3.14	(i) PHP	2.57
(c) Swift	2.79	(v) Lisp	3.40	(c) Swift	2.71
(c) Haskell	3.10	(c) Haskell	3.55	(i) Python	2.80
(v) C#	3.14	(c) Swift	4.20	(c) Ocaml	2.82
(c) Go	3.23	(c) Fortran	4.20	(v) C#	2.85
(i) Dart	3.83	(v) F#	6.30	(i) Hack	3.34
(v) F#	4.13	(i) JavaScript	6.52	(v) Racket	3.52
(i) JavaScript	4.45	(i) Dart	6.67	(i) Ruby	3.97
(v) Racket	7.91	(v) Racket	11.27	(c) Chapel	4.00
(i) TypeScript	21.50	(i) Hack	26.99	(v) F#	4.25
(i) Hack	24.02	(i) PHP	27.64	(i) JavaScript	4.59
(i) PHP	29.30	(v) Erlang	36.71	(i) TypeScript	4.69
(v) Erlang	42.23	(i) Jruby	43.44	(v) Java	6.01
(i) Lua	45.98	(i) TypeScript	46.20	(i) Perl	6.62
(i) Jruby	46.54	(i) Ruby	59.34	(i) Lua	6.72
(i) Ruby	69.91	(i) Perl	65.79	(v) Erlang	7.20
(i) Python	75.88	(i) Python	71.90	(i) Dart	8.64
(i) Perl	79.58	(i) Lua	82.91	(i) Jruby	19.84

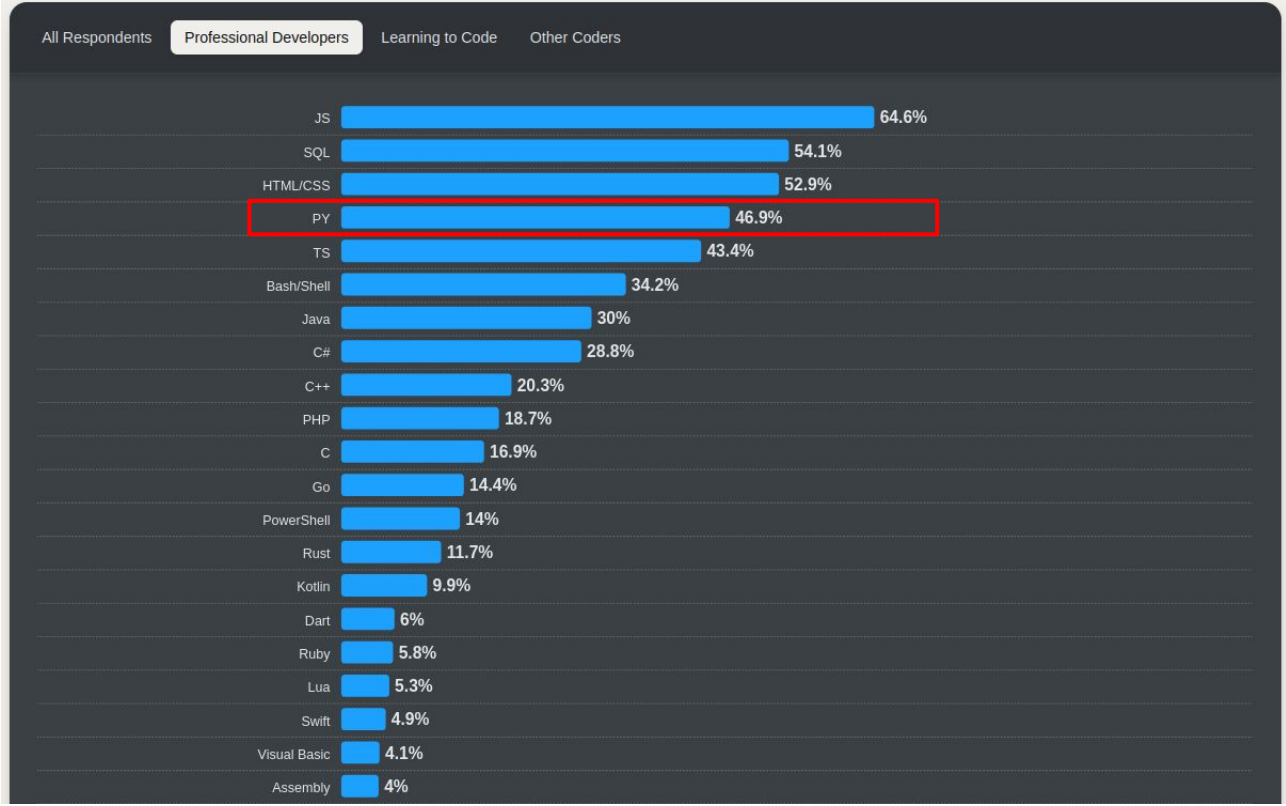
https://doi.org/10.1145/3136014.3136031

Python Amongst



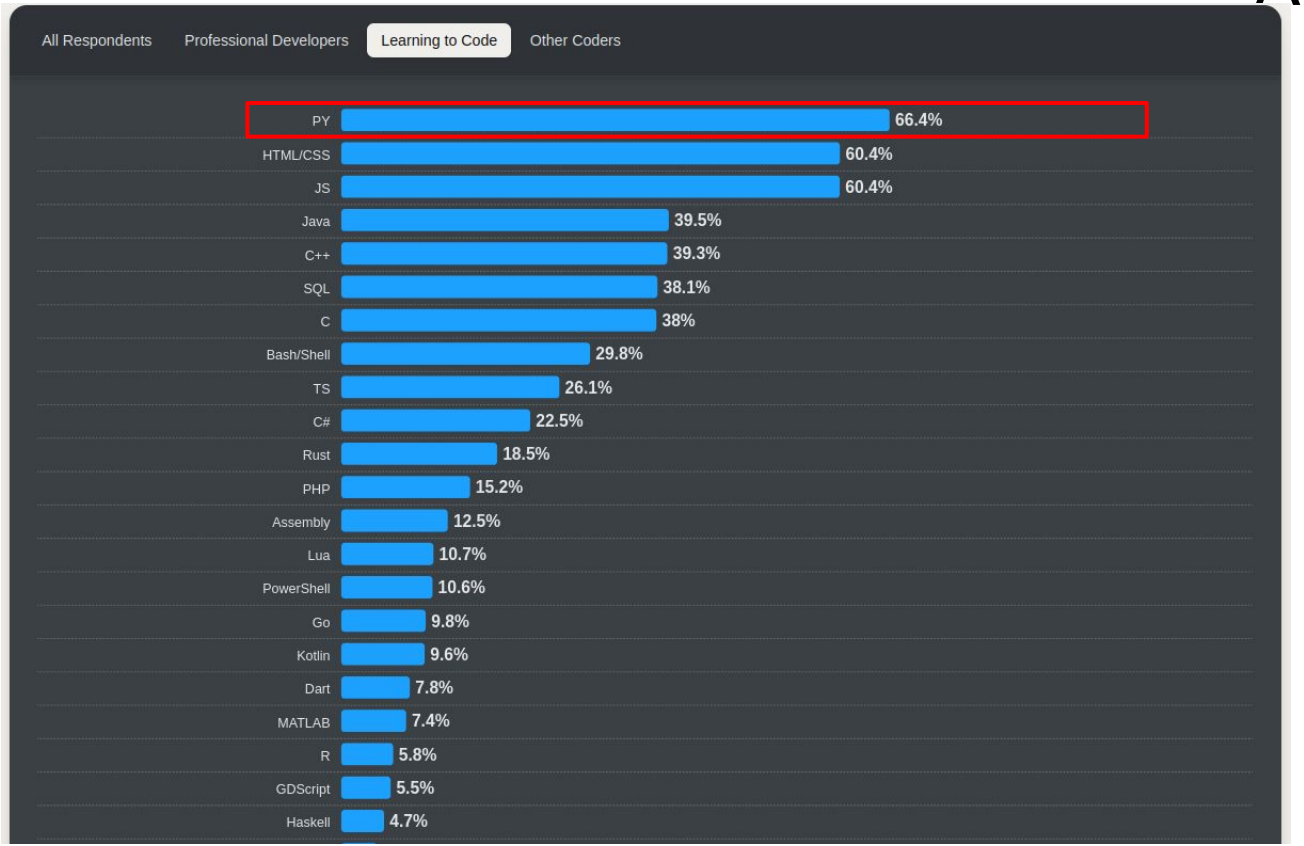
https://survey.stackoverflow.co/2024/technology#most-popular-technologies-language

Python Amongst



https://survey.stackoverflow.co/2024/technology#most-popular-technologies-language-prof

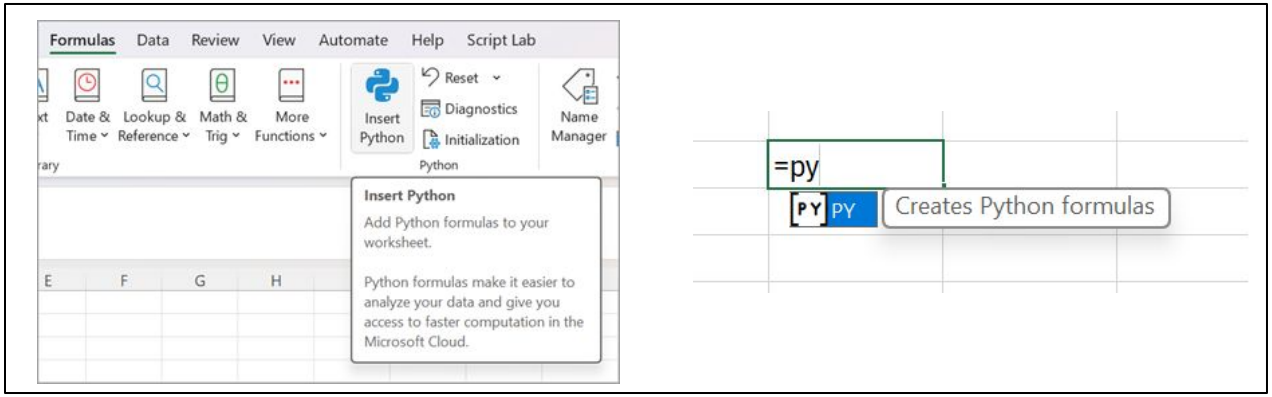
Python Amongst

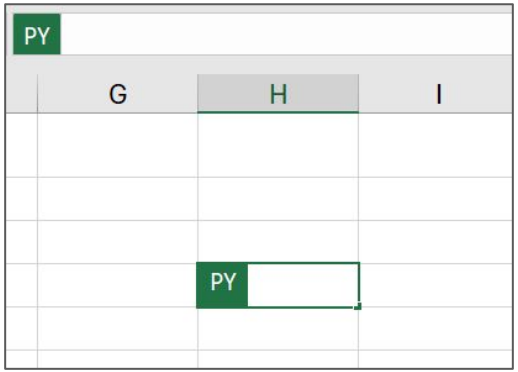


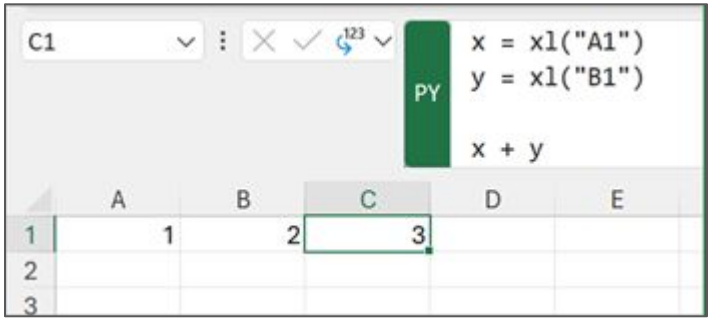
https://survey.stackoverflow.co/2024/technology#most-popular-technologies-language-learn

Python

Excel

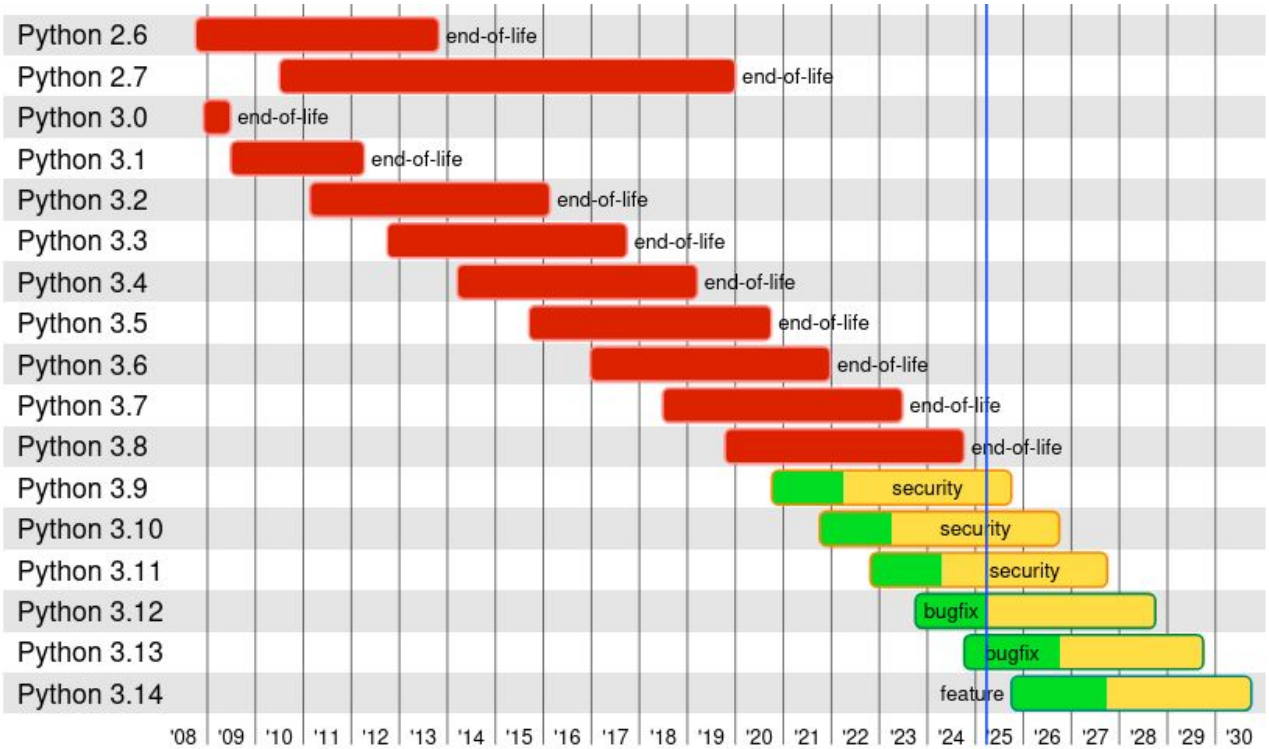






Python Pool

Python Pool



Chosen Criteria

https://devguide.python.org/versions/#versions

Python Pool

Interpreted Versions

Version	Release Date	Maintenance status
3.13.2	Feb. 4, 2025	Bugfix
3.13.1	Dec. 3, 2024	Bugfix
3.12.9	Feb. 4, 2025	Bugfix
3.12.8	Dec. 3, 2024	Bugfix
3.12.3	April 9, 2024	Bugfix
3.11.11	Dec. 3, 2024	Security
3.10.16	Dec. 3, 2024	Security
3.9.21	Dec. 3, 2024	Security
3.0.1	Feb. 13, 2009	End-of-life
2.7.18	April 20, 2020	End-of-life
2.0.1	June 22, 2001	End-of-life

https://www.python.org/downloads/

Python Pool

Python Compilers

Name	Version
Codon	0.18.2
Nuitka	0.4.1

https://www.reddit.com/r/Python/comments/13cbemn/list_of_python_compilers/?rdt=38151

Object of Research

Object of Research

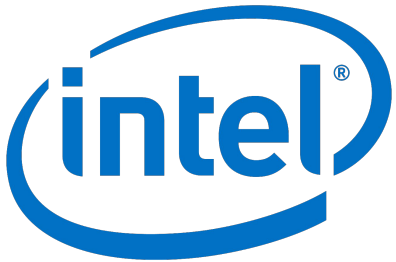
- *RQ1*
  - Which interpreter best handles **recursion**?
- *RQ2*
  - Which interpreter best handles **iteration**?
- *RQ3*
  - Which interpreter handles **memory management** best?
- *RQ4*
  - Which interpreter handles **data querying and processing** best?
- *RQ5*
  - What’s the most **green compiler**, and to which extent?

Methodology

Benchmark

RAPL

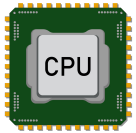
- Run 10x *via* RAPL
  - Delete min and max outlier
    - Mean

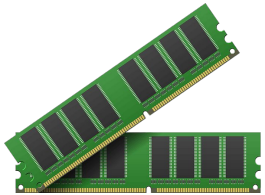


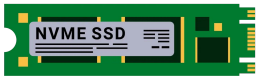
Environment

- CPU
  - Intel(R) Core(TM) i3-9100F CPU @ 3.60GHz
    - 4 Cores - 4 Threads
- RAM
  - DDR4 @ 2400 MHz
    - 8GB Dual Band (16 GB)
- ROM
  - M.2 2280 NVMe SSD
    - R 2,400 MB/s
- OS
  - Ubuntu 24.04.2 LTS
    - Linux 6.11.0-19-generic x86_64

Inspects







Environment

- GCC
  - V 13.3.0

C Language Compiler

Fibonacci

Fibonacci Recursive

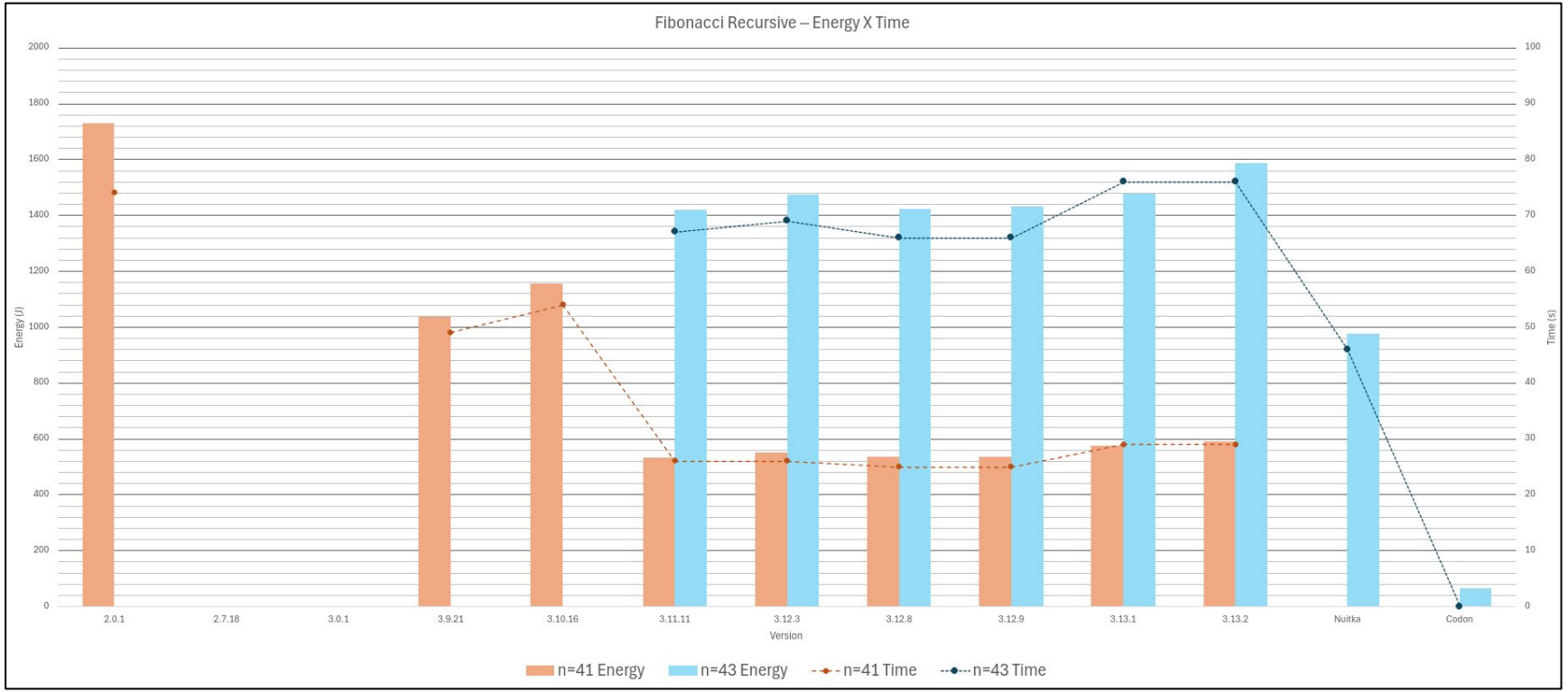
- Time Complexity
  - O(N)

Profile

- *Object of Research*
  - Recursion

Fibonacci

Energy X Time



Fibonacci

N	41		43	
Version	**Energy (J)**	**Time (s)**	**Energy (J)**	**Time (s)**
2.0.1	1 730	74	–	–
2.7.18	–	–	–	–
3.0.1	–	–	–	–
3.9.21	1 036	49	–	–
3.10.16	1 154	54	–	–
3.11.11	531	26	1 418	67
3.12.3	551	26	1 474	69
3.12.8	536	25	1 423	66
3.12.9	536	25	1 430	66
3.13.1	573	29	1 577	76
3.13.2	591	29	1 586	76

Energy X Time

N	43	
Version	**Energy (J)**	**Time (s)**
Nuitka	975	46
Codon	64	0.003

Fibonacci

N	41			43		
Version	**Power Up**	**Speed Up**	**Green Up**	**Power Up**	**Speed Up**	**Green Up**
2.0.1	1	1	1	–	–	–
2.7.18	–	–	–	–	–	–
3.0.1	–	–	—	–	–	–
3.9.21	0.91	1.53	1.67	–	–	–
3.10.16	0.92	1.38	1.50	–	–	–
3.11.11	0.89	2.91	3.26	1	1	1
3.12.3	0.90	2.82	3.14	1.01	0.97	0.96
3.12.8	0.91	2.94	3.22	1.02	1.02	1.00
3.12.9	0.91	2.93	3.22	1.02	1.01	0.99
3.13.1	0.84	2.55	3.02	0.98	0.88	0.90
3.13.2	0.86	2.52	2.92	0.98	0.88	0.89

Power Up X Speed Up X Green Up

N	43		
Version	**Power Up**	**Speed Up**	**Green Up**
3.11.11	1	1	1
Nuitka	1.00	1.46	1.45
Codon	0.97	21.30	22.01

Sieve of Eratosthenes

(Primes)

Sieve of Eratosthenes

- Time Complexity
  - O(N * log(log(N)))

Profile

- *Object of Research*
  - Iteration

Sieve of Eratosthenes

Energy X Time

Sieve of Eratosthenes

From 0 to	100 000		200 000 000		300 000 000	
Version	**Energy (J)**	**Time (s)**	**Energy (J)**	**Time (s)**	**Energy (J)**	**Time(s)**
2.0.1	1 247	52	–	–	–	–
2.7.18	1 218	52	–	–	–	–
3.0.1	1	0	1 460	69	–	–
3.9.21	–	–	1 407	67	–	–
3.10.16	–	–	1 488	71	–	–
3.11.11	–	–	1 145	56	–	–
3.12.3	–	–	913	44	1 203	57
3.12.8	–	–	882	42	1 122	53
3.12.9	–	–	899	43	1 148	54
3.13.1	–	–	898	43	1 138	55
3.13.2	–	–	909	44	1 169	56

Energy X Time

From 0 to	300 000 000	
Version	**Energy (J)**	**Time (s)**
Nuitka	1 415	67
Codon	33	2

Primes

Power Up X Speed Up X Green Up

From 0 to	100 000			200 000 000			300 000 000		
Version	**Power Up**	**Speed Up**	**Green Up**	**Power Up**	**Speed Up**	**Green Up**	**Power Up**	**Speed Up**	**Green Up**
2.0.1	1	1	1	–	–	–	–	–	–
2.7.18	0.98	1.00	1.02	–	–	–	–	–	–
3.0.1	0.68	928.30	1 370.67	1	1	1	–	–	–
3.9.21	–	–	–	1.00	1.04	1.04	–	–	–
3.10.16	–	–	–	0.99	0.97	0.98	–	–	–
3.11.11	–	–	–	1.00	1.27	1.27	–	–	–
3.12.3	–	–	–	0.99	1.58	1.60	1	1	1
3.12.8	–	–	–	1.00	1.66	1.65	1.0	1.08	1.07
3.12.9	–	–	–	1.00	1.63	1.62	1.01	1.06	1.05
3.13.1	–	–	–	0.99	1.61	1.63	0.99	1.05	1.06
3.13.2	–	–	–	0.99	1.58	1.61	0.99	1.02	1.03

From 0 to	300 000 000		
Version	**Power Up**	**Speed Up**	**Green Up**
3.12.3	1	1	1
Nuitka	1.01	0.86	0.85
Codon	1.02	37.16	36.60

Bubble Sort

Bubble Sort

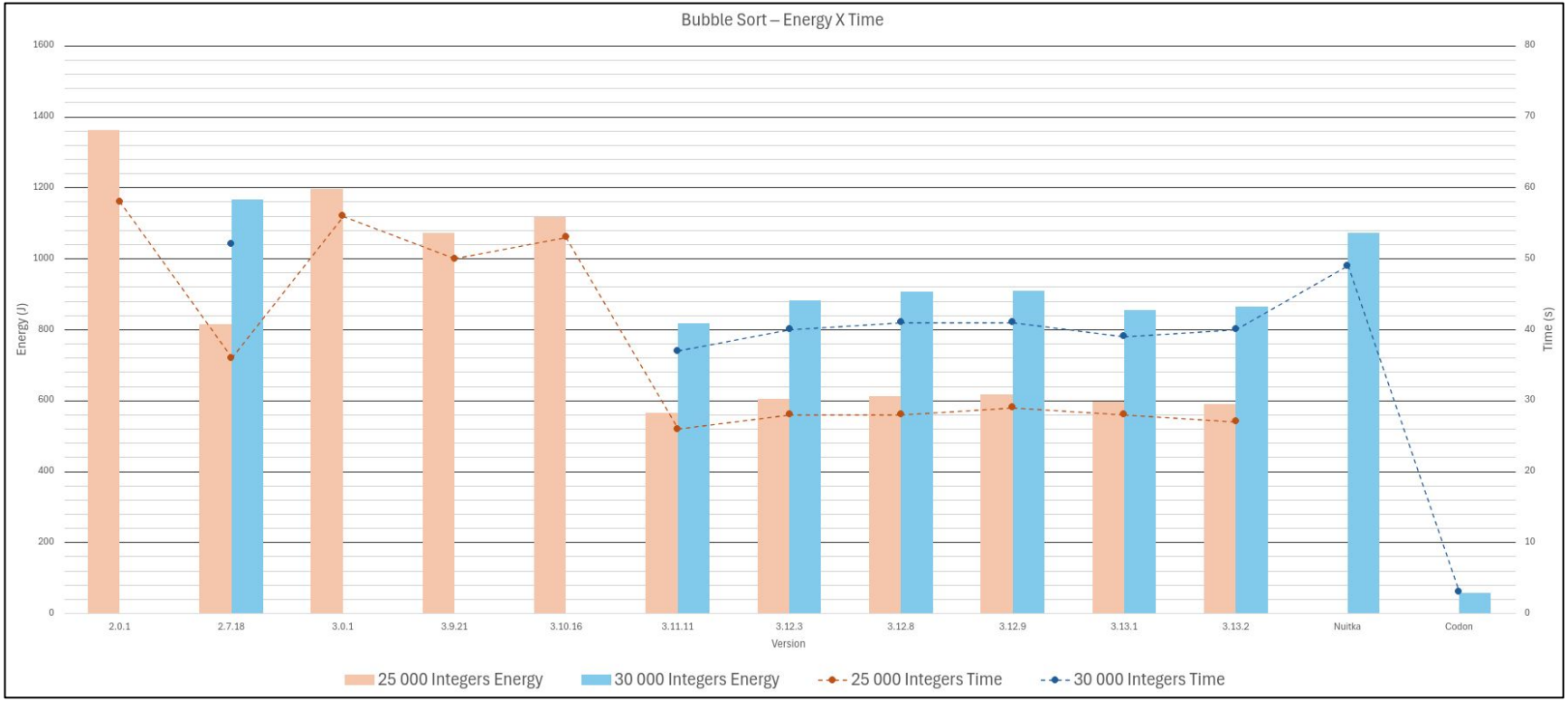
- Best case
  - O(N)
- Worst case
  - $O(N^2)$
- Average case
  - $O(N^2)$

Profile

- *Object of Research*
  - Memory management

Bubble Sort

Energy X Time



Bubble Sort

Integers	25 000		30 000	
Version	**Energy (J)**	**Time (s)**	**Energy (J)**	**Time (s)**
2.0.1	1 361	58	–	–
2.7.18	815	36	1 167	52
3.0.1	1 197	56	–	–
3.9.21	1 073	50	–	–
3.10.16	1 116	53	–	–
3.11.11	565	26	818	37
3.12.3	604	28	881	40
3.12.8	612	28	905	41
3.12.9	617	29	908	41
3.13.1	596	28	855	39
3.13.2	590	27	863	40

Energy X Time

Integers	30 000	
Version	**Energy (J)**	**Time (s)**
Nuitka	1 073	49
Codon	57	3

Bubble Sort

Power Up X Speed Up X Green Up

Integers	25 000			30 000		
Version	**Power Up**	**Speed Up**	**Green Up**	**Power Up**	**Speed Up**	**Green Up**
2.0.1	1	1	1	–	–	–
2.7.18	0.96	1.60	1.67	1	1	1
3.0.1	0.90	1.03	1.14	–	–	–
3.9.21	0.90	1.15	1.27	–	–	–
3.10.16	0.89	1.09	1.22	–	–	–
3.11.11	0.93	2.24	2.41	0.98	1.39	1.43
3.12.3	0.92	2.07	2.25	0.97	1.28	1.32
3.12.8	0.92	2.04	2.22	0.97	1.25	1.29
3.12.9	0.92	2.03	2.20	0.97	1.25	1.29
3.13.1	0.92	2.10	2.28	0.96	1.32	1.36
3.13.2	0.91	2.11	2.31	0.96	1.30	1.35

Integers	30 000		
Version	**Power Up**	**Speed Up**	**Green Up**
2.7.18	1	1	1
Nuitka	0.96	1.05	1.09
Codon	0.87	17.98	20.59

Insertion Sort

Insertion Sort

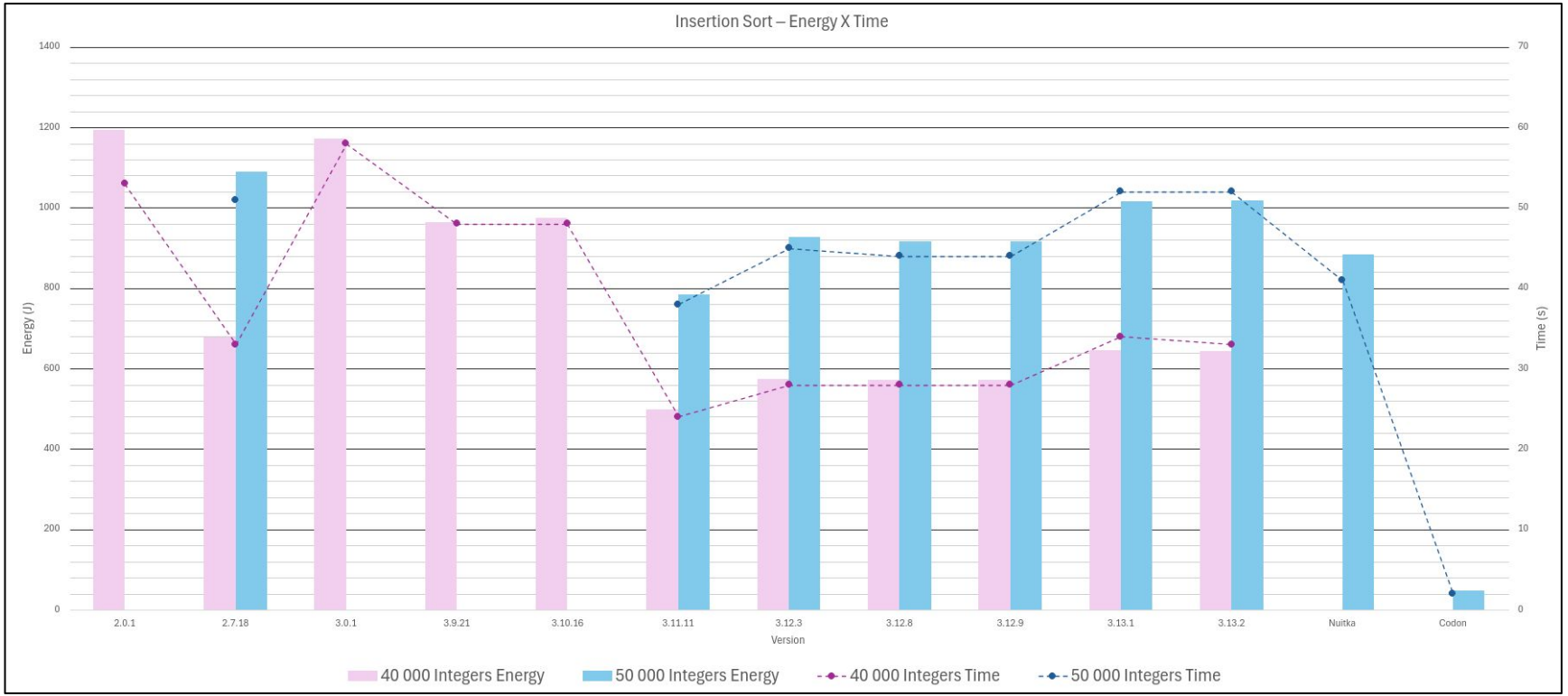
- Best case
  - O(N)
- Worst case
  - $O(N^2)$
- Average case
  - $O(N^2)$

Profile

- *Object of Research*
  - Memory management

Insertion Sort

Energy X Time



Insertion Sort

Integers	40 000		50 000	
Version	**Energy (J)**	**Time (s)**	**Energy (J)**	**Time (s)**
2.0.1	1 194	53	–	–
2.7.18	678	33	1 090	51
3.0.1	1 171	58	–	–
3.9.21	964	48	–	–
3.10.16	974	48	–	–
3.11.11	498	24	784	38
3.12.3	575	28	927	45
3.12.8	573	28	916	44
3.12.9	573	28	916	44
3.13.1	645	34	1 017	52
3.13.2	644	33	1 019	52

Energy X Time

Integers	50 000	
Version	**Energy (J)**	**Time (s)**
Nuitka	883	41
Codon	47	2

Insertion Sort

Power Up X Speed Up X Green Up

Integers	40 000			50 000		
Version	**Power Up**	**Speed Up**	**Green Up**	**Power Up**	**Speed Up**	**Green Up**
2.0.1	1	1	1	–	–	–
2.7.18	0.92	1.62	1.76	1	1	1
3.0.1	0.89	0.91	1.02	–	–	–
3.9.21	0.90	1.11	1.24	–	–	–
3.10.16	0.90	1.10	1.23	–	–	–
3.11.11	0.91	2.17	2.40	0.97	1.35	1.39
3.12.3	0.90	1.87	2.07	0.97	1.14	1.18
3.12.8	0.90	1.88	2.08	0.98	1.16	1.19
3.12.9	0.91	1.90	2.08	0.98	1.16	1.19
3.13.1	0.85	1.58	1.85	0.92	0.99	1.07
3.13.2	0.86	1.60	1.85	0.93	0.99	1.07

Integers	50 000		
Version	**Power Up**	**Speed Up**	**Green Up**
2.7.18	1	1	1
Nuitka	1.02	1.26	1.23
Codon	0.99	22.74	23.07

Shell Sort

Shell Sort

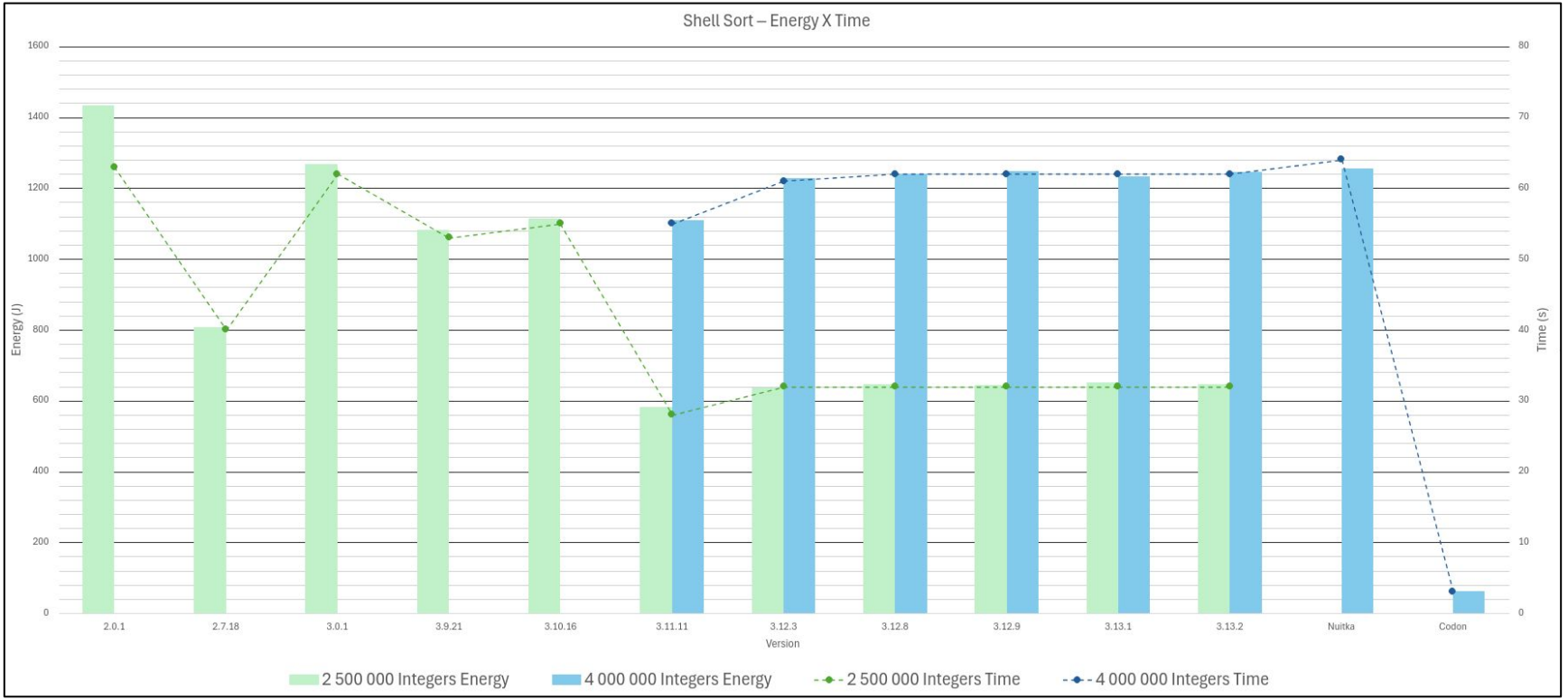
- Best case
  - O(N * $\log^2(N)$)
- Worst case
  - O(2^N)
- Average case
  - O(N * $\log^2(N)$)

Profile

- *Object of Research*
  - Memory management

Shell Sort

Energy X Time



Shell Sort

Integers	2 500 000		4 000 000	
Version	**Energy (J)**	**Time (s)**	**Energy (J)**	**Time (s)**
2.0.1	1 434	63	–	–
2.7.18	806	40	–	–
3.0.1	1 268	62	–	–
3.9.21	1 082	53	–	–
3.10.16	1 113	55	–	–
3.11.11	582	28	1 110	55
3.12.3	636	32	1 228	61
3.12.8	646	32	1 240	62
3.12.9	645	32	1 248	62
3.13.1	652	32	1 233	62
3.13.2	646	32	1 245	62

Energy X Time

Integers	4 000 000	
Version	**Energy (J)**	**Time (s)**
Nuitka	1 255	64
Codon	62	3

Shell Sort

Power Up X Speed Up X Green Up

Integers	2 500 000			4 000 000		
Version	Power Up	Speed Up	Green Up	Power Up	Speed Up	Green Up
2.0.1	1	1	1	–	–	–
2.7.18	0.90	1.60	1.78	–	–	–
3.0.1	0.90	1.02	1.13	–	–	–
3.9.21	0.91	1.20	1.33	–	–	–
3.10.16	0.90	1.15	1.29	–	–	–
3.11.11	0.92	2.26	2.46	1	1	1
3.12.3	0.89	2.01	2.25	0.98	0.89	0.90
3.12.8	0.89	1.97	2.22	1.00	0.89	0.89
3.12.9	0.89	1.97	2.22	0.99	0.88	0.89
3.13.1	0.89	1.95	2.20	0.98	0.88	0.90
3.13.2	0.88	1.96	2.22	0.98	0.88	0.89

Integers	4 000 000		
Version	Power Up	Speed Up	Green Up
3.11.11	1	1	1
Nuitka	0.97	0.86	0.88
Codon	0.99	17.86	18.03

Merge Sort

Merge Sort

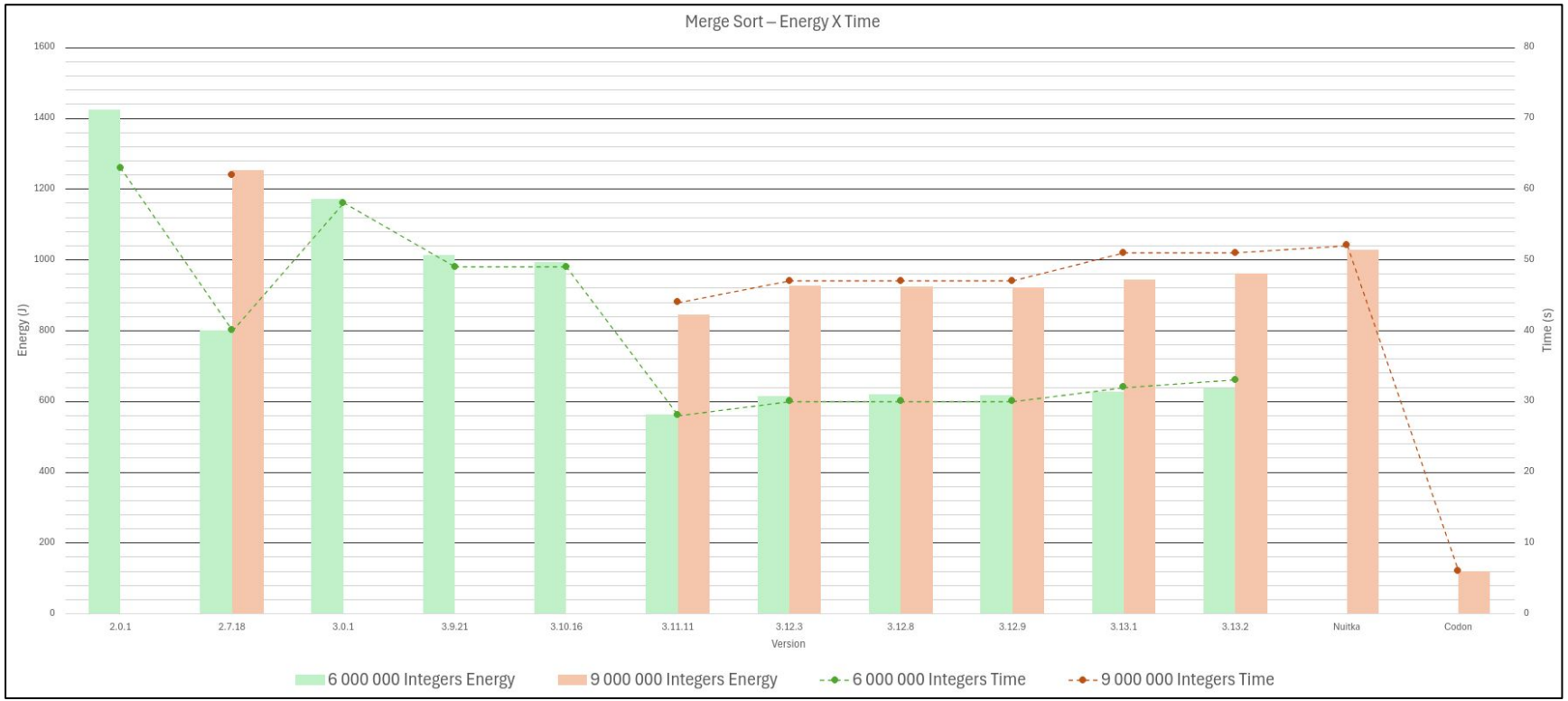
- Best case
  - O(N * log²(N))
- Worst case
  - O(N * log²(N))
- Average case
  - O(N * log²(N))

Profile

- *Object of Research*
  - Memory management
  - Context switching
  - *Recursion*

Merge Sort

Energy X Time



Merge Sort

Integers	6 000 000		9 000 000	
Version	**Energy (J)**	**Time (s)**	**Energy (J)**	**Time (s)**
2.0.1	1 423	63	–	–
2.7.18	801	40	1 252	62
3.0.1	1 171	58	–	–
3.9.21	1 012	49	–	–
3.10.16	994	49	–	–
3.11.11	563	28	845	44
3.12.3	615	30	926	47
3.12.8	618	30	923	47
3.12.9	616	30	921	47
3.13.1	627	32	944	51
3.13.2	636	33	960	51

Energy X Time

Integers	9 000 000	
Version	**Energy (J)**	**Time (s)**
Nuitka	1 028	52
Codon	118	6

Merge Sort

Power Up X Speed Up X Green Up

Integers	6 000 000			9 000 000		
Version	**Power Up**	**Speed Up**	**Green Up**	**Power Up**	**Speed Up**	**Green Up**
2.0.1	1	1	1	–	–	–
2.7.18	0.90	1.60	1.78	1	1	1
3.0.1	0.90	1.09	1.22	–	–	–
3.9.21	0.91	1.29	1.41	–	–	–
3.10.16	0.91	1.30	1.43	–	–	–
3.11.11	0.89	2.26	2.53	0.95	1.40	1.48
3.12.3	0.90	2.09	2.31	0.96	1.30	1.35
3.12.8	0.91	2.09	2.30	0.96	1.30	1.36
3.12.9	0.92	2.12	2.31	0.97	1.32	1.36
3.13.1	0.87	1.97	2.27	0.92	1.22	1.33
3.13.2	0.87	1.94	2.24	0.92	1.20	1.30

Integers	9 000 000		
Version	**Power Up**	**Speed Up**	**Green Up**
2.7.18	1	1	1
Nuitka	0.97	1.18	1.22
Codon	1.06	11.20	10.60

Quick Sort

Quick Sort

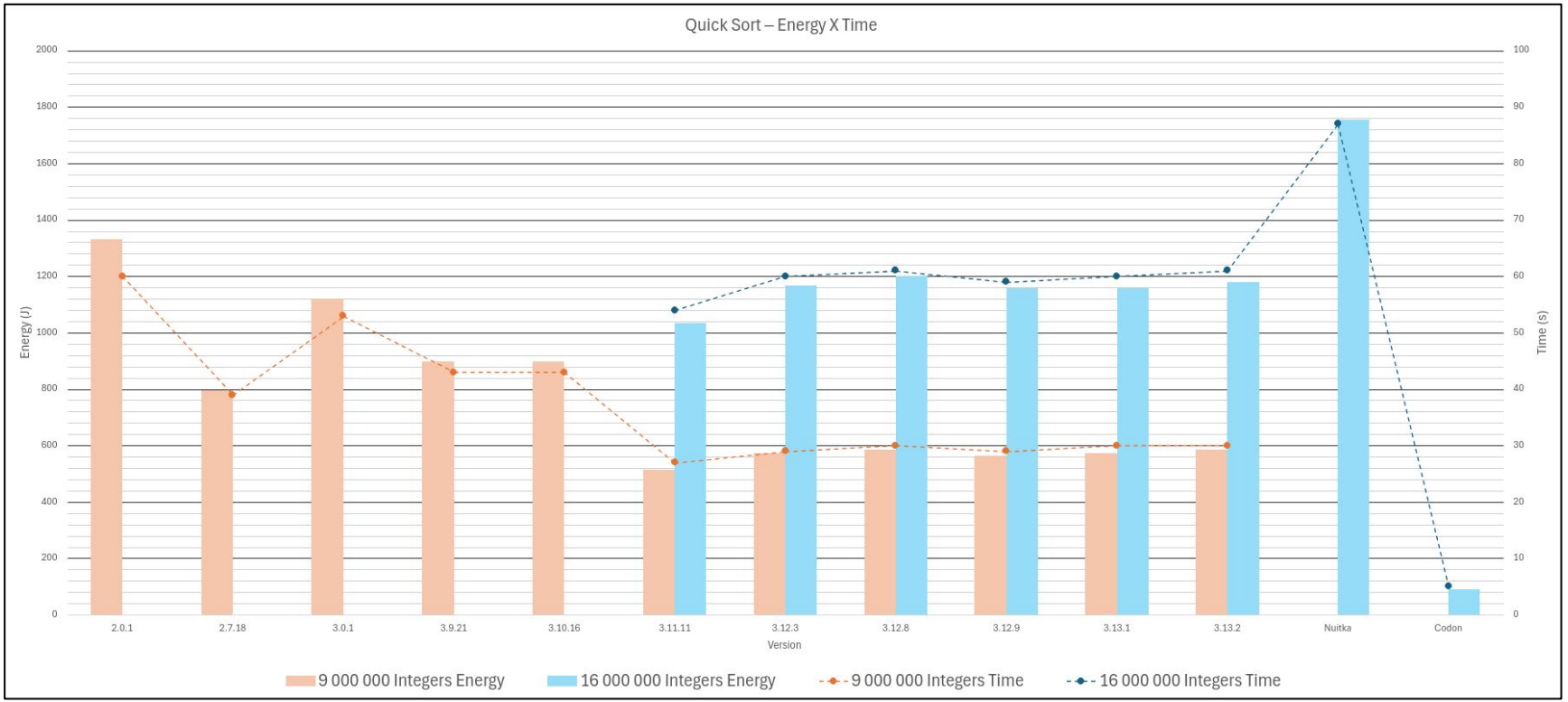
- Best case
  - O(N * $\log^2(N)$)
- Worst case
  - O(2^N)
- Average case
  - O(N * $\log^2(N)$)

Profile

- *Object of Research*
  - Memory management

Quick Sort

Energy X Time



Quick Sort

Integers	9 000 000		17 000 000	
Version	**Energy (J)**	**Time (s)**	**Energy (J)**	**Time (s)**
2.0.1	1 330	60	–	–
2.7.18	797	39	–	–
3.0.1	1 120	53	–	–
3.9.21	899	43	–	–
3.10.16	897	43	–	–
3.11.11	513	27	1 034	54
3.12.3	572	29	1 166	60
3.12.8	586	30	1 200	61
3.12.9	565	29	1 157	59
3.13.1	572	30	1 156	60
3.13.2	584	30	1 178	61

Energy X Time

Integers	17 000 000	
Version	**Energy (J)**	**Time (s)**
Nuitka	1 755	87
Codon	91	5

Quick Sort

Power Up X Speed Up X Green Up

Integers	9 000 000			17 000 000		
Version	**Power Up**	**Speed Up**	**Green Up**	**Power Up**	**Speed Up**	**Green Up**
2.0.1	1	1	1	–	–	–
2.7.18	0.92	1.54	1.67	–	–	–
3.0.1	0.95	1.12	1.19	–	–	–
3.9.21	0.93	1.38	1.48	–	–	–
3.10.16	0.93	1.38	1.48	–	–	–
3.11.11	0.85	2.21	2.59	1	1	1
3.12.3	0.87	2.02	2.32	1.03	0.91	0.89
3.12.8	0.88	2.00	2.27	1.04	0.90	0.86
3.12.9	0.86	2.03	2.35	1.02	0.92	0.89
3.13.1	0.86	2.00	2.32	1.01	0.91	0.89
3.13.2	0.87	1.99	2.28	1.02	0.90	0.88

Integers	17 000 000		
Version	**Power Up**	**Speed Up**	**Green Up**
3.11.11	1	1	1
Nuitka	1.06	0.62	0.59
Codon	1.01	11.45	11.33

Tim Sort

Tim Sort

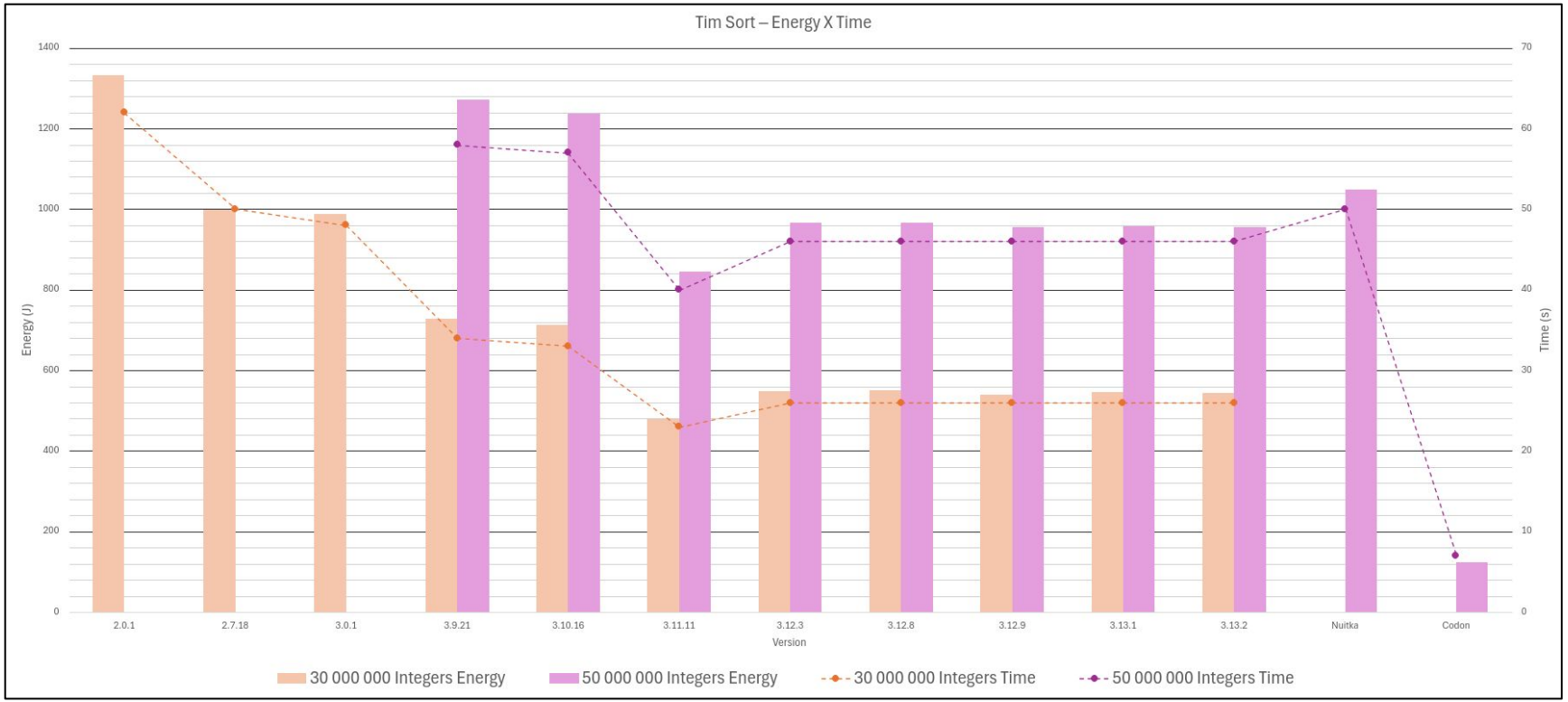
- Best case
  - O(N)
- Worst case
  - O(N * $\log^2(N)$)
- Average case
  - O(N * $\log^2(N)$)

Profile

- *Object of Research*
  - Memory management
  - Context Switching
  - Recursion

Tim Sort

Energy X Time



Tim Sort

Integers	30 000 000		50 000 000	
Version	**Energy (J)**	**Time (s)**	**Energy (J)**	**Time (s)**
2.0.1	1 333	62	–	–
2.7.18	999	50	–	–
3.0.1	987	48	–	–
3.9.21	727	34	1 271	58
3.10.16	712	33	1 238	57
3.11.11	478	23	844	40
3.12.3	549	26	966	46
3.12.8	550	26	967	46
3.12.9	540	26	955	46
3.13.1	547	26	957	46
3.13.2	544	26	955	46

Energy X Time

Integers	50 000 000	
Version	**Energy (J)**	**Time (s)**
Nuitka	1 049	50
Codon	124	7

Tim Sort

Power Up X Speed Up X Green Up

Integers	30 000 000			50 000 000		
Version	Power Up	Speed Up	Green Up	Power Up	Speed Up	Green Up
2.0.1	1	1	1	–	–	–
2.7.18	0.92	1.23	1.33	–	–	–
3.0.1	0.95	1.28	1.35	–	–	–
3.9.21	1.01	1.84	1.83	1	1	1
3.10.16	1.00	1.87	1.87	0.99	1.02	1.03
3.11.11	0.97	2.69	2.79	0.96	1.44	1.51
3.12.3	0.97	2.35	2.43	0.96	1.26	1.32
3.12.8	0.97	2.35	2.42	0.96	1.27	1.31
3.12.9	0.97	2.39	2.47	0.96	1.28	1.33
3.13.1	0.97	2.36	2.44	0.95	1.27	1.33
3.13.2	0.96	2.36	2.45	0.95	1.27	1.33

Integers	50 000 000		
Version	Power Up	Speed Up	Green Up
3.9.21	1	1	1
Nuitka	0.97	1.18	1.21
Codon	0.84	8.63	10.25

Querying

Querying

Dataset

USER		
Field	**Type**	**Constraint**
username	VARCHAR(75)	PK
name	VARCHAR(75)	–
gender	VARCHAR(1)	ENUM
birth_date	DATE	–
account_creation	DATE	–
pay_method	VARCHAR(16)	ENUM
account_status	VARCHAR(8)	ENUM

RIDE		
Field	**Type**	**Constraint**
id	VARCHAR(75)	PK
date	DATE	–
driver	INTEGER	FK
user	VARCHAR(75)	FK
city	VARCHAR(64)	–
distance	INTEGER	–
score_user	FLOAT	–
score_driver	FLOAT	–
tip	FLOAT	–
comment	TEXT	–

DRIVER		
Field	**Type**	**Constraint**
id	INTEGER	PK
name	VARCHAR(75)	–
gender	VARCHAR(1)	ENUM
birth_date	DATE	–
account_creation	DATE	–
account_status	VARCHAR(8)	ENUM
car_class	VARCHAR(8)	ENUM
license_plate	VARCHAR(8)	–
city	VARCHAR(64)	–

Querying

User Total Spent

Querying

- Load
  - 1.38GB
- Validate + Structure
  - 107 900 026 words
    - avg. 12 length
  - Index Users by *username*
  - Index Drivers by *driver_id*
  - Group Rides by *username*

User Total Spent

- Query
  - Access Rides by *username*
    - Access Drivers by *driver_id*
      - Get *car_class*
        - Calculate **fee**
    - Access Users by *username*
      - Get info
      - Get *birthday_date*
        - Calculate **age**

Querying

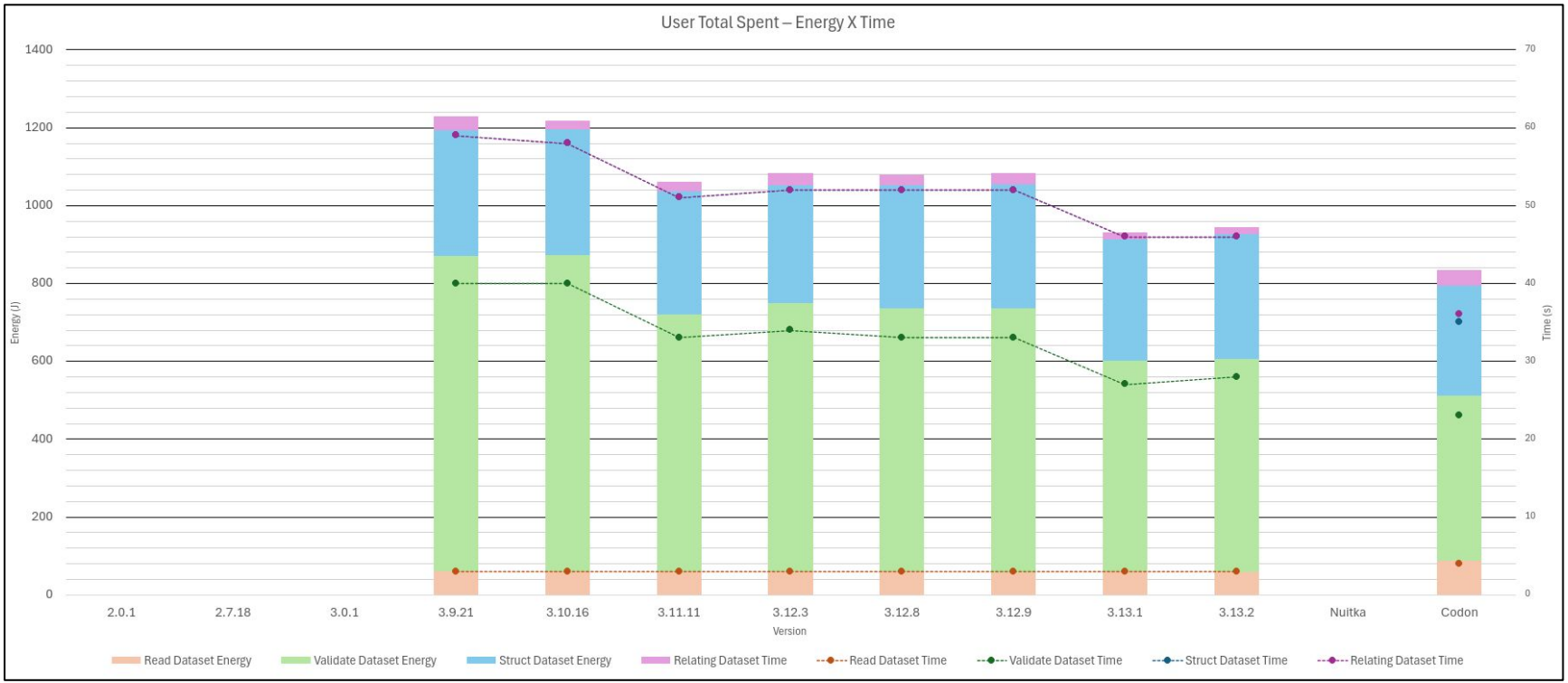
- Best case
  - Θ(1)
- Worst case
  - Θ(N)
- Average case
  - Θ(N)

User Total Spent

- *Object of Research*
  - Memory management
  - Context Switching
  - Parsing
  - Iteration

Querying – User Total Spent

Energy X Time



Querying – User Total Spent

Table accesses	1		10		33	
Version	Energy (J)	Time (s)	Energy (J)	Time (s)	Energy (J)	Time(s)
2.0.1	–	–	–	–	–	–
2.7.18	–	–	–	–	–	–
3.0.1	–	–	–	–	–	–
3.9.21	1 229	59	1 196	60	1 224	59
3.10.16	1 219	59	1 203	59	1 225	59
3.11.11	1 061	52	1 041	52	1 066	52
3.12.3	1 084	53	1 063	53	1 084	53
3.12.8	1 080	52	1 050	53	1 081	53
3.12.9	1 083	53	1 061	53	1 086	53
3.13.1	931	46	918	46	932	46
3.13.2	945	46	926	46	942	46

Energy X Time

Table accesses	1		10		33	
Version	Energy (J)	Time (s)	Energy (J)	Time (s)	Energy (J)	Time(s)
Nuitka	–	–	–	–	–	–
Codon	833	37	833	37	835	37

Querying – User Total Spent

Accesses	33		
Version	**Power Up**	**Speed Up**	**Green Up**
2.0.1	–	–	–
2.7.18	–	–	–
3.0.1	–	–	–
3.9.21	1	1	1
3.10.16	1.01	1.01	1.00
3.11.11	1.00	1.15	1.15
3.12.3	1.00	1.13	1.13
3.12.8	1.00	1.13	1.13
3.12.9	1.00	1.13	1.13
3.13.1	0.98	1.29	1.31
3.13.2	0.99	1.28	1.30

Power Up X Speed Up X Green Up

Accesses	33		
Version	**Power Up**	**Speed Up**	**Green Up**
3.9.21	1	1	1
Nuitka	–	–	–
Codon	1.11	1.62	1.47

Querying

Drivers Top Score

Querying

- Load
  - 1.38GB
- Validate
  - 107 900 026 words
    - avg. 12 length

Drivers Top Score

- Structure
  - Index Drivers by *driver_id*
  - Group Rides by *driver_id*
- Query
  - Access Rides by *driver_id*
    - Group AVG(*score*) by *driver_id*
  - Sort DESC
  - Slice

Querying

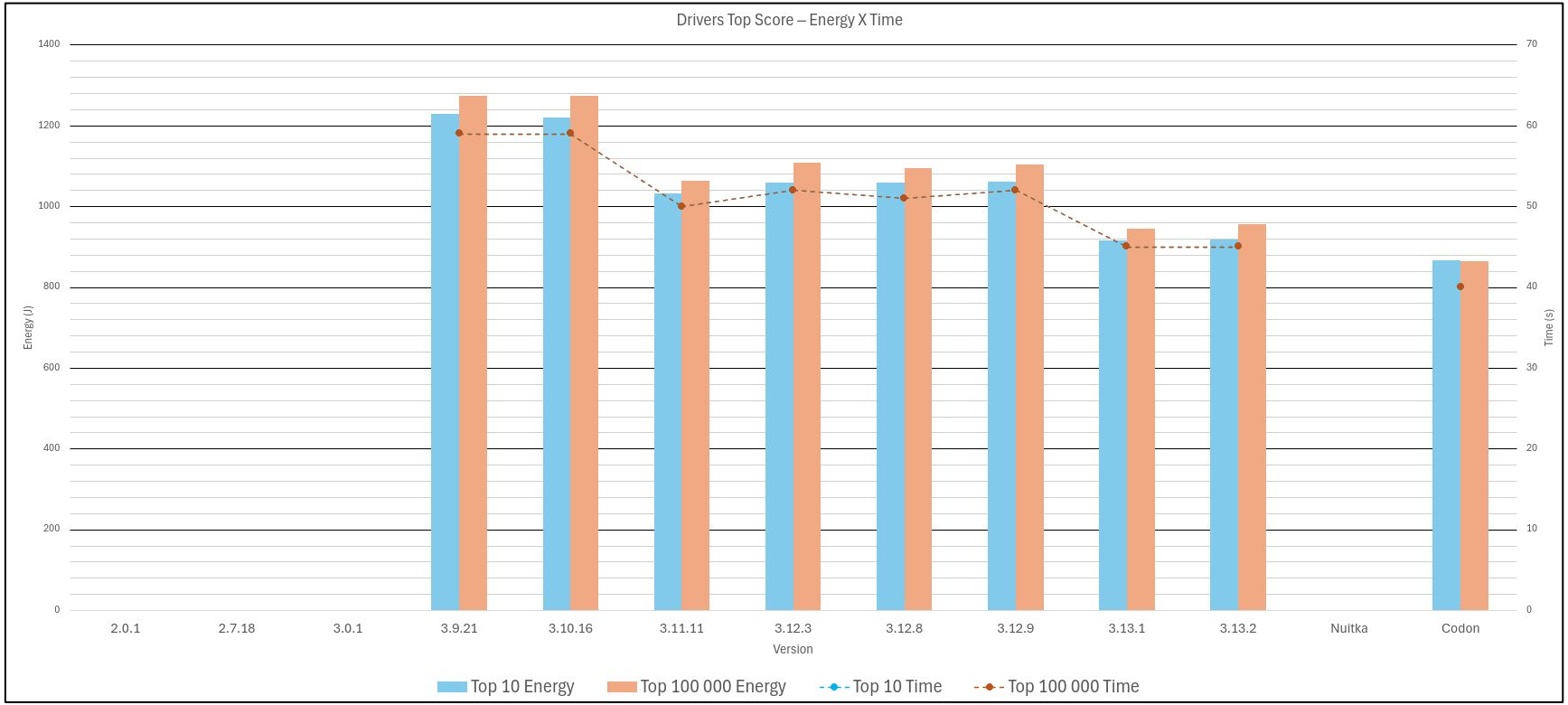
- Best case
  - Θ(N)
- Worst case
  - Θ(N)
- Average case
  - Θ(N)

User Total Spent

- *Object of Research*
  - Memory management
  - Context Switching
  - Parsing
  - Iteration

Querying – Drivers Top Score

Energy X Time



Querying – Drivers Top Score

Top	10		100 000	
Version	**Energy (J)**	**Time (s)**	**Energy (J)**	**Time (s)**
2.0.1	–	–	–	—
2.7.18	–	–	–	–
3.0.1	–	–	–	–
3.9.21	1 228	59	1 273	59
3.10.16	1 219	59	1 273	59
3.11.11	1 031	50	1 064	50
3.12.3	1 058	51	1 107	52
3.12.8	1 059	51	1 095	51
3.12.9	1 061	52	1 103	52
3.13.1	915	45	945	45
3.13.2	918	45	955	45

Energy X Time

Top	10		100 000	
Version	**Energy (J)**	**Time (s)**	**Energy (J)**	**Time (s)**
Nuitka	–	–	–	–
Codon	833	37	833	37

Querying – Drivers Top Score

Top	100 000		
Version	**Power Up**	**Speed Up**	**Green Up**
2.0.1	–	–	–
2.7.18	–	–	–
3.0.1	–	–	–
3.9.21	1	1	1
3.10.16	1.00	1.00	1.00
3.11.11	0.99	1.18	1.20
3.12.3	1.00	1.14	1.15
3.12.8	0.99	1.15	1.16
3.12.9	1.00	1.15	1.15
3.13.1	0.98	1.32	1.35
3.13.2	0.98	1.30	1.33

Power Up X Speed Up X Green Up

Top	100 000		
Version	**Power Up**	**Speed Up**	**Green Up**
3.9.21	1	1	1
Nuitka	–	–	–
Codon	1.02	1.50	1.47

Authors Notes

Threats to Validity

- Representativeness
  - CPU Architecture
  - Use Cases

- Energy consumption as a whole

Future Work

- Benchmark data structures
- Benchmark object oriented programming
- Context switching analysis
- Develop data analysis

Conclusion

- *RQ1:* Which interpreter best handles **recursion**?
  - Python 3.12.8
    - Fibonacci
- *RQ2:* Which interpreter best handles **iteration**?
  - Python 3.12.3
    - Sieve of Eratosthenes
- *RQ3:* Which interpreter handles **memory management** best?
  - Python 3.11.11
    - Sorting algorithms
- *RQ4:* Which interpreter handles **data querying and processing** best?
  - Python 3.13.1
    - Querying
- *RQ5:* What's the most **green compiler**, and to which extent?
  - Codon
    - More on the next slide…

Conclusion

RQ5

Overall			
Version	**Power Up**	**Speed Up**	**Green Up**
Top Tiers	1	1	1
Codon	1.00	13.09	13.27

Recursion			
Version	**Power Up**	**Speed Up**	**Green Up**
3.12.8	1	1	1
Codon	0.95	20.99	22.08

Data Manipulation			
Version	**Power Up**	**Speed Up**	**Green Up**
3.13.1	1	1	1
Codon	1.10	1.20	1.09

Iteration			
Version	**Power Up**	**Speed Up**	**Green Up**
3.12.8	1	1	1
Codon	1.09	34.51	34.14

Memory Management			
Version	**Power Up**	**Speed Up**	**Green Up**
3.11.11	1	1	1
Codon	0.98	12.17	12.39

Experimentação em Engenharia de Software

On the Performance of the Python Language

PG 54232
PG 55972
PG 57539