Automating Victory:

Beating browser games with accessible Python

Jon Gaul: An abridged history

Georgia Tech grad

USA → Japan

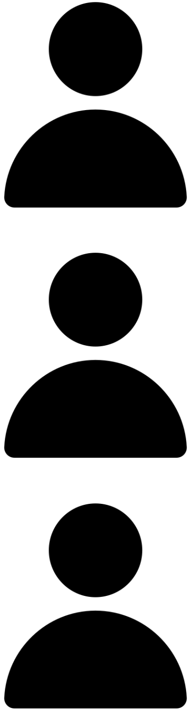
Software Engineer at HENNGE

- Bouldering
- Wrote a kid's book
- Love video essays

HENNGE

New Hampshire
Vermont
Massachusetts
Maine
tana
North Dak
Minnesota
Wisconsin
South Dakota
Michigan
New York
yoming
Rhode Island
Connecticut
Nebraska
Iowa
Pennsylvania
New Jersey
Ohio
Delaware
Colorado
Illinois
Indiana
Maryland
Kansas
Missouri
Virginia
District of Columbia
Kentucky
North Carolina
West Virginia
Oklahoma
Arkansas
Tennessee
South Carolina
U.S. Virgin Islands
ew Mexico
Puerto Rico
Alabama
Georgia
Mississippi
Texas
Guam
Louisiana
Northern Mariana Islands
Florida
American Samoa
waii

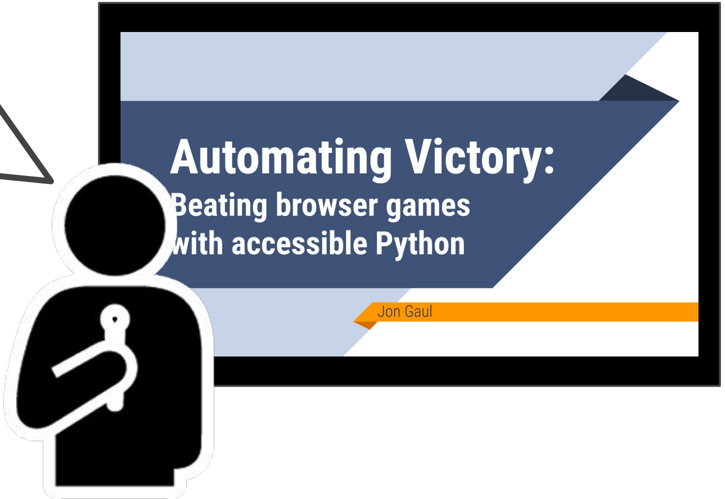
What’s on the schedule



Project management tips

Useful I/O libraries

(Here’s a cool video game!)



Automating Victory:
Beating browser games
with accessible Python

Jon Gaul

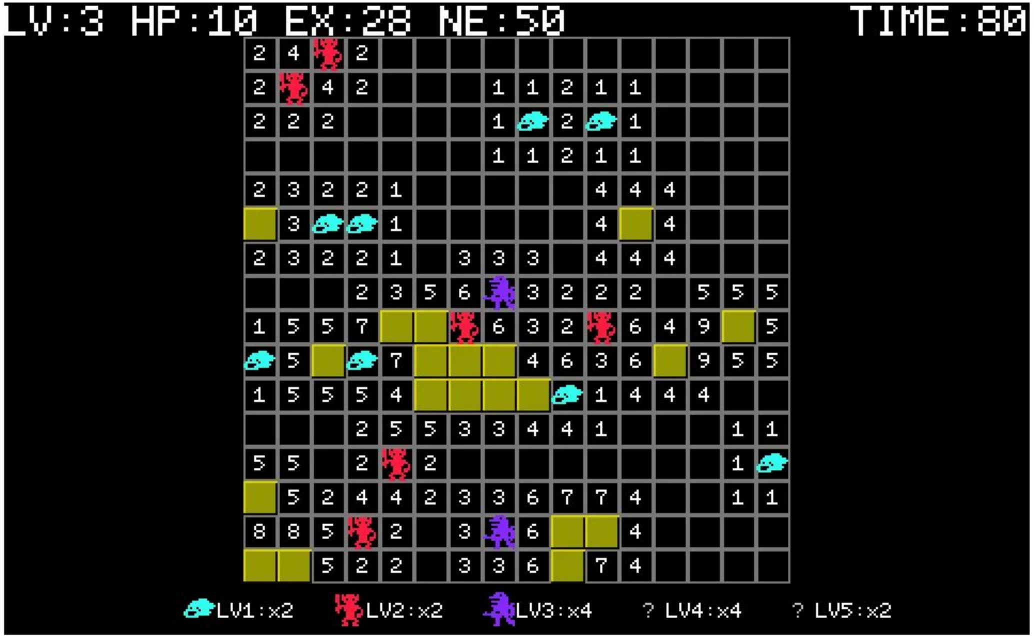
マモノスイーパー / Mamono Sweeper

Dev: **Hojamaka Games**

https://hojamaka.com/

- Flash game (2010)
- Android game (2015)
- HTML5 game (2016)

Minesweeper x RPG

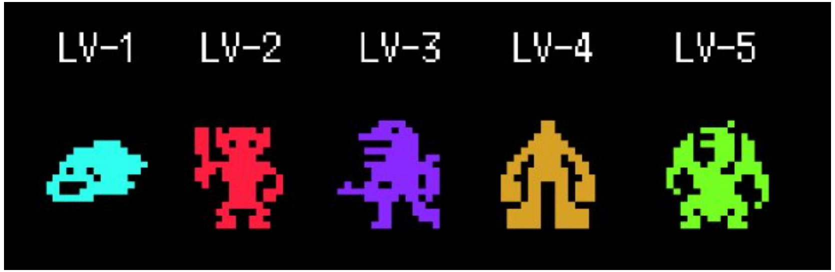


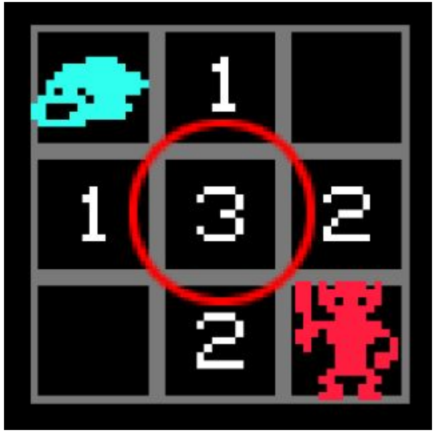
LV:3 HP:10 EX:28 NE:50 TIME:80

2	4	🔴	2												
2	🔴	4	2				1	1	2	1	1				
2	2	2					1	🔵	2	🔵	1				
							1	1	2	1	1				
2	3	2	2	1						4	4	4			
🟨	3	🔵	🔵	1						4	🟨	4			
2	3	2	2	1		3	3	3		4	4	4			
			2	3	5	6	🟣	3	2	2	2		5	5	5
1	5	5	7	🟨	🟨	🔴	6	3	2	🔴	6	4	9	🟨	5
🔵	5	🟨	🔵	7	🟨	🟨	🟨	4	6	3	6	🟨	9	5	5
1	5	5	5	4	🟨	🟨	🟨	🟨	🔵	1	4	4	4		
			2	5	5	3	3	4	4	1				1	1
5	5		2	🔴	2									1	🔵
🟨	5	2	4	4	2	3	3	6	7	7	4			1	1
8	8	5	🔴	2		3	🟣	6	🟨	🟨	4				
🟨	🟨	5	2	2		3	3	6	🟨	7	4				

🔵LV1:x2 🔴LV2:x2 🟣LV3:x4 ? LV4:x4 ? LV5:x2

マモノスイーパー / Mamono Sweeper





Click any monster at or below your level

Mistakes lose HP (and then the game!)

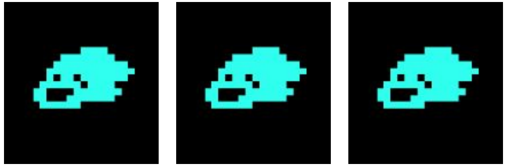
Numbers are the sum of neighboring monsters’ levels

マモノスイーパー / Mamono Sweeper

For 3 with nothing revealed:

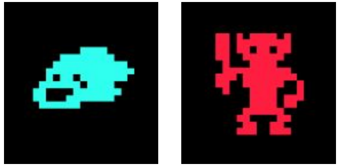
Minesweeper: 56 possibilities

Mamono Sweeper: **120 possibilities**



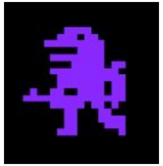
56 arrangements

+



56 arrangements

+



8 arrangements

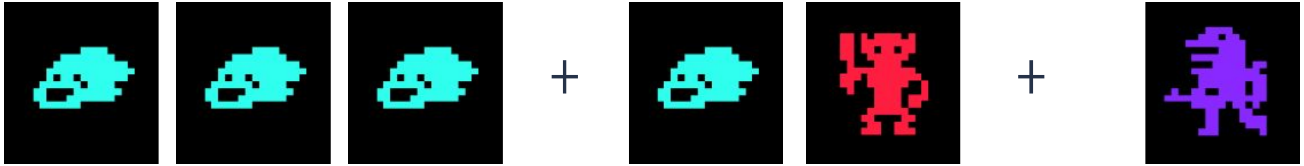
マモノスイーパー / Mamono Sweeper

For 3 with nothing revealed:

Minesweeper: 56 possibilities

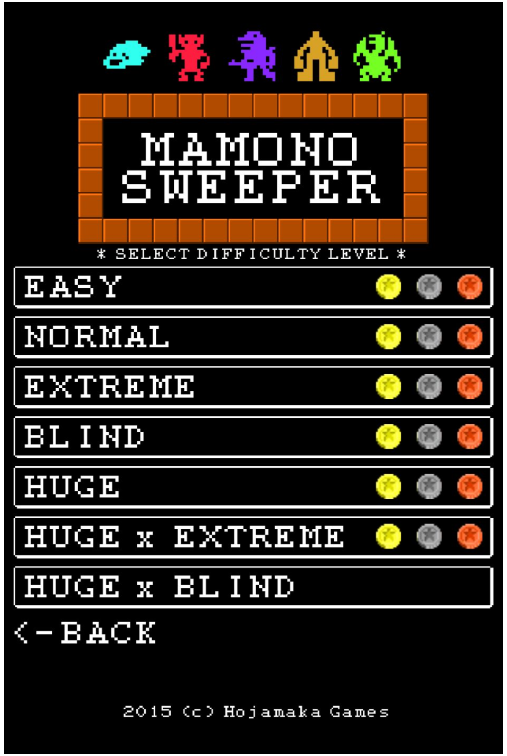
Mamono Sweeper: **120 possibilities**

Often doesn’t matter



56 arrangements 56 arrangements 8 arrangements

マモノスイーパー / Mamono Sweeper



MAMONO SWEEPER

* SELECT DIFFICULTY LEVEL *

EASY

NORMAL

EXTREME

BLIND

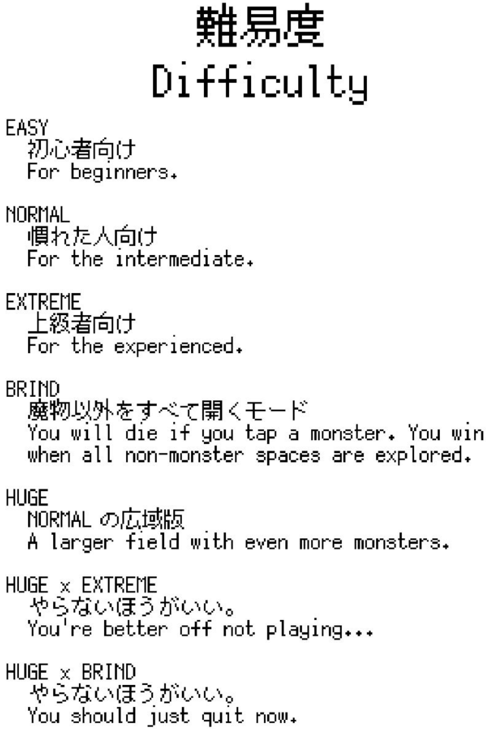
HUGE

HUGE x EXTREME

HUGE x BLIND

<-BACK

2015 (c) Hojamaka Games



難易度
Difficulty

EASY
初心者向け
For beginners.

NORMAL
慣れた人向け
For the intermediate.

EXTREME
上級者向け
For the experienced.

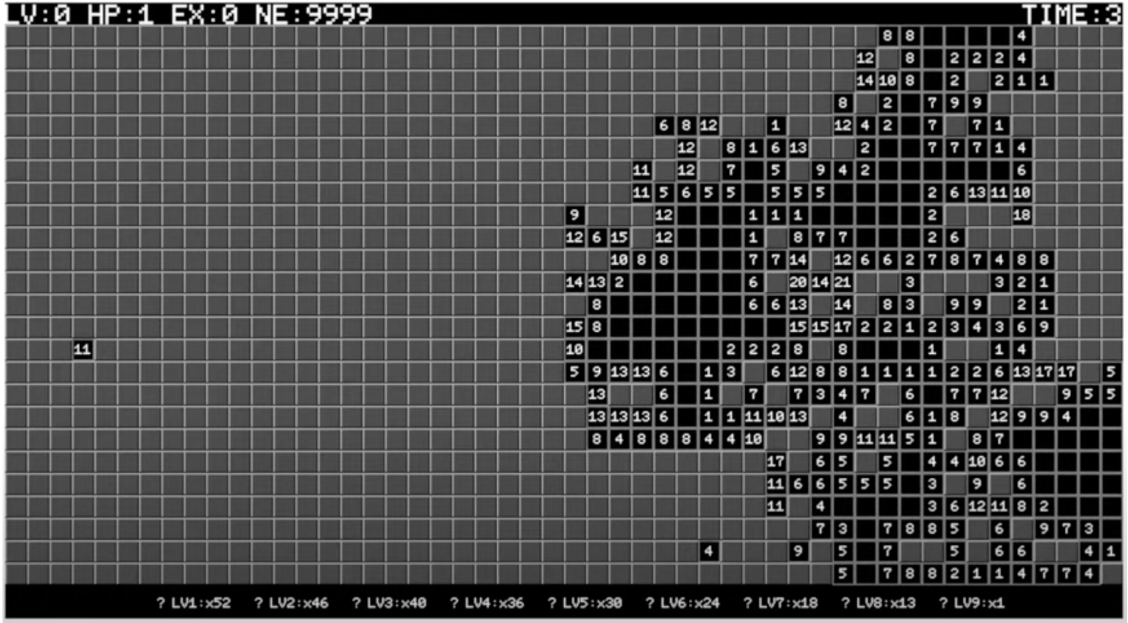
BRIND
魔物以外をすべて開くモード
You will die if you tap a monster. You win when all non-monster spaces are explored.

HUGE
NORMAL の広域版
A larger field with even more monsters.

HUGE x EXTREME
やらないほうがいい。
You're better off not playing...

HUGE x BRIND
やらないほうがいい。
You should just quit now.

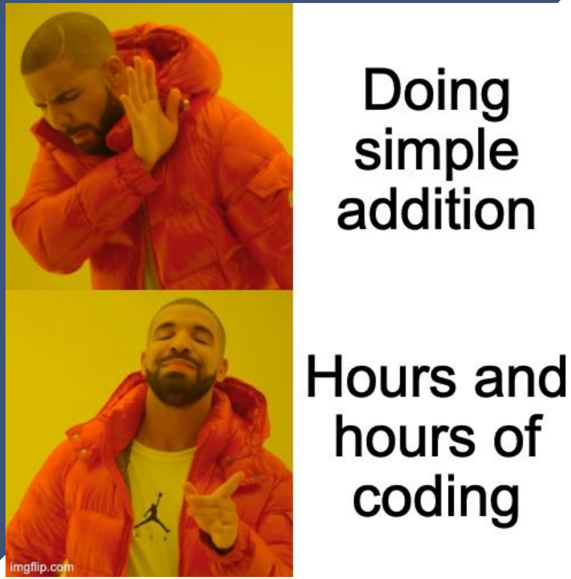
マモノスイーパー / Mamono Sweeper



```
LV:0 HP:1 EX:0 NE:9999                                    TIME:3

? LV1:x52   ? LV2:x46   ? LV3:x40   ? LV4:x36   ? LV5:x30   ? LV6:x24   ? LV7:x18   ? LV8:x13   ? LV9:x1
```

How to eliminate careless math errors: don’t do math



Do what I do but better

Look at the board

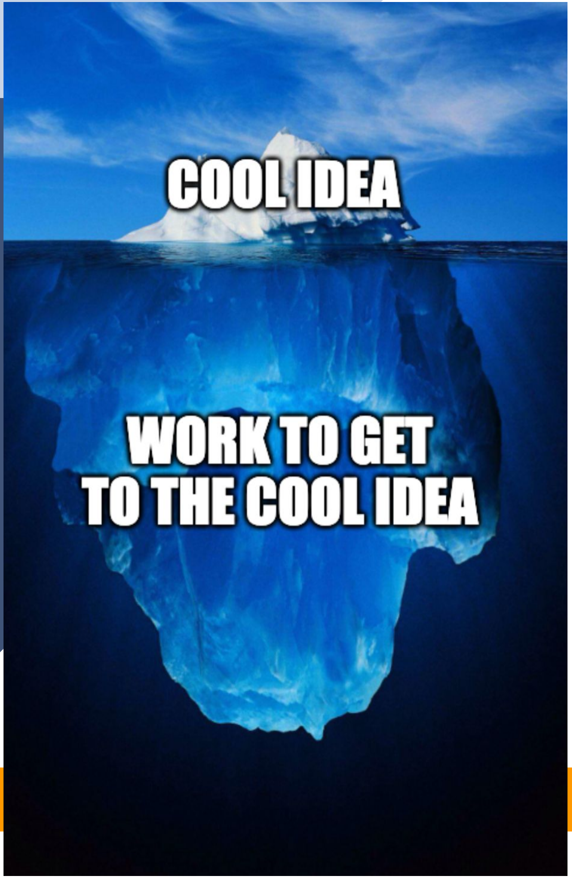
Parse into a numerical model

Identify safe moves

Make those moves

Repeat until win or loss

Then I ignored the project for 3 years



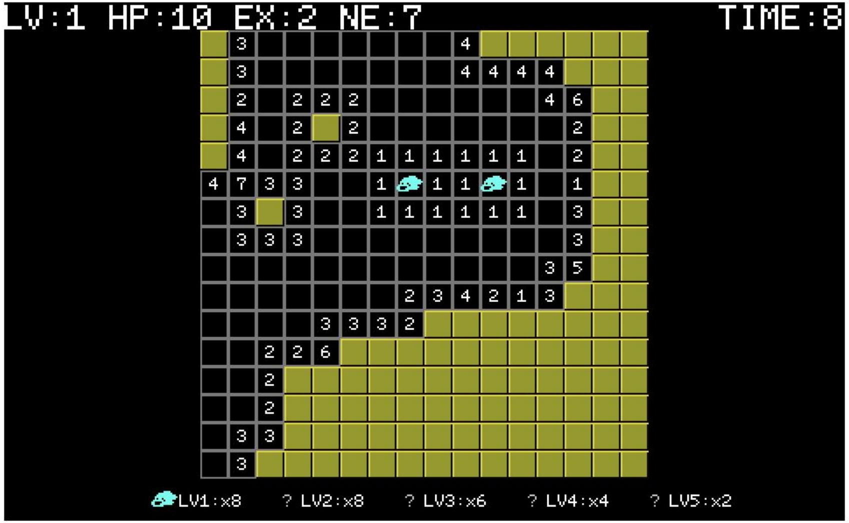
1: *Really* minimize your Minimum Viable Product (MVP)

Look at the board

MSS + **OpenCV** to the rescue

- `mss().grab(...)`
- `cv2.matchTemplate(...)`
- Where is LV:1?

OpenCV



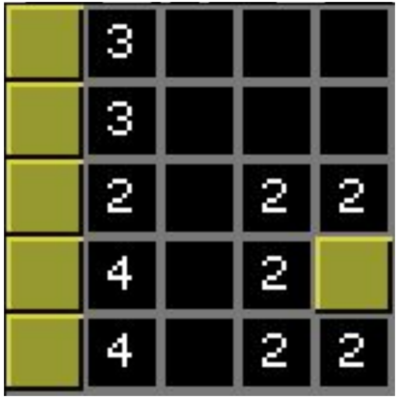
LV:1 HP:10 EX:2 NE:7 TIME:8

▨	3								4	▨	▨	▨	▨	▨	▨
▨	3								4	4	4	4	▨	▨	▨
▨	2		2	2	2							4	6	▨	▨
▨	4		2	▨	2								2	▨	▨
▨	4		2	2	2	1	1	1	1	1	1		2	▨	▨
4	7	3	3			1		1	1		1		1	▨	▨
	3	▨	3			1	1	1	1	1	1		3	▨	▨
	3	3	3										3	▨	▨
												3	5	▨	▨
							2	3	4	2	1	3	▨	▨	▨
				3	3	3	2	▨	▨	▨	▨	▨	▨	▨	▨
		2	2	6	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨
		2	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨
		2	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨
	3	3	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨
	3	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨

LV1:x8 ? LV2:x8 ? LV3:x6 ? LV4:x4 ? LV5:x2

Parse into a model: Can I get your number?

How to get the values in each spot



	3			
	3			
	2		2	2
	4		2	
	4		2	2

`value`

0	3	0	0	0
0	3	0	0	0
0	2	0	2	2
0	4	0	2	0
0	4	0	2	2

`is_revealed`

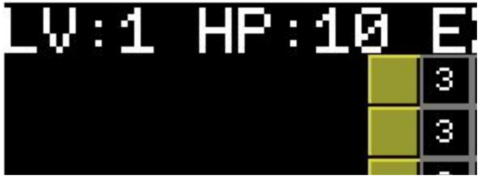
F	T	T	T	T
F	T	T	T	T
F	T	T	T	T
F	T	T	T	F
F	T	T	T	T

Parse into a model: Can I get your number?

How to get the values in ~~each spot?~~

→ Just get current level!

Optical Character Recognition?



LV:1 HP:10 E

3

3

Parse into a model: Can I get your number?

How to get the values in ~~each spot?~~

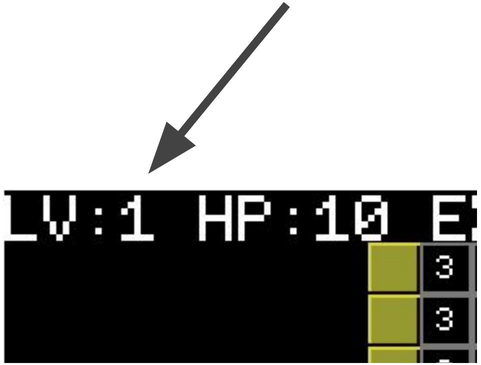
→ Just get current level!

~~Optical Character Recognition?~~

NO! Too much work!

Compare with references

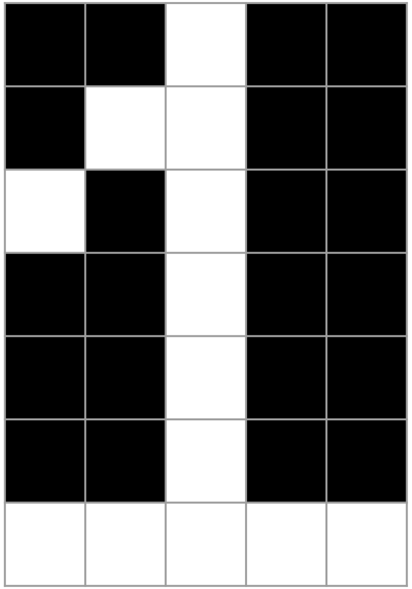
→ `int_from_pixels()`



LV:1 HP:10 E

3

3

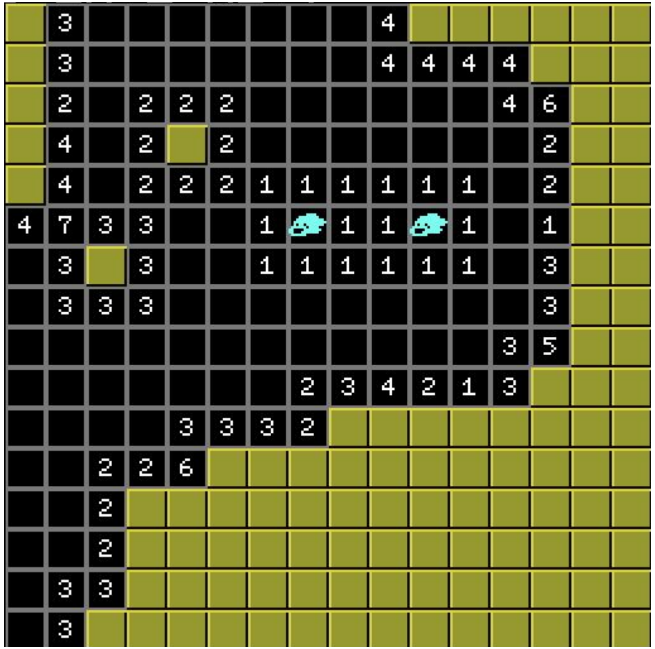


Parse into a model: Can I get your number?

How to get the numbers in each square?

Modify `int_from_pixels()`

Always easier to modify existing code!

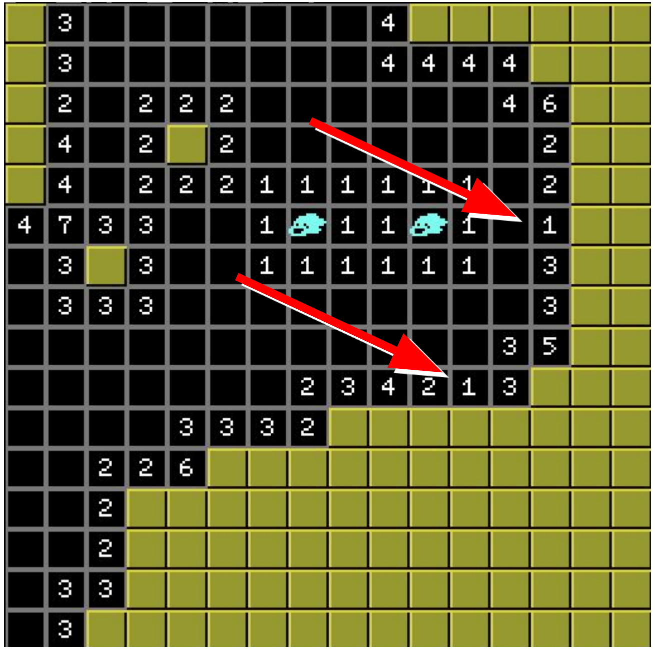


🟨	3								4	🟨	🟨	🟨	🟨	🟨	🟨
🟨	3								4	4	4	4	🟨	🟨	🟨
🟨	2		2	2	2							4	6	🟨	🟨
🟨	4		2	🟨	2								2	🟨	🟨
🟨	4		2	2	2	1	1	1	1	1	1		2	🟨	🟨
4	7	3	3			1		1	1		1		1	🟨	🟨
	3	🟨	3			1	1	1	1	1	1		3	🟨	🟨
	3	3	3										3	🟨	🟨
												3	5	🟨	🟨
							2	3	4	2	1	3	🟨	🟨	🟨
				3	3	3	2	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
		2	2	6	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
		2	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
		2	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
	3	3	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
	3	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨

Identify the Minimum Viable Product (MVP)

What is the simplest strategy?

If a square's number is equal or lower than the current level, click its neighbors.

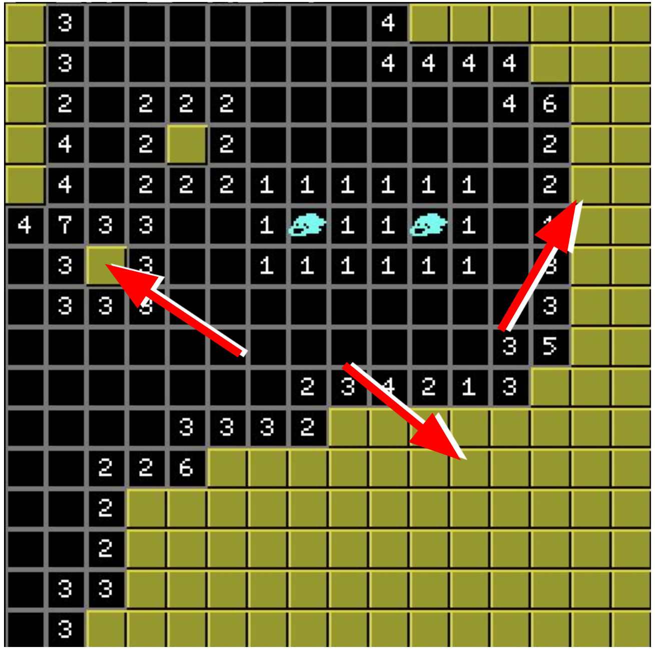


Identify the Minimum Viable Product (MVP)

What is the simplest strategy?

~~If a square's number is equal or lower than the current level, click its neighbors.~~

Click randomly



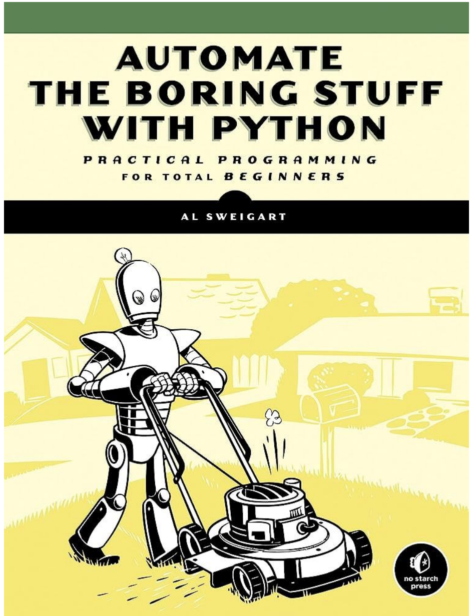
🟨	3								4	🟨	🟨	🟨	🟨	🟨	🟨
🟨	3								4	4	4	4	🟨	🟨	🟨
🟨	2		2	2	2							4	6	🟨	🟨
🟨	4		2	🟨	2								2	🟨	🟨
🟨	4		2	2	2	1	1	1	1	1	1		2	🟨	🟨
4	7	3	3			1	●	1	1	●	1		1	🟨	🟨
	3	🟨	3			1	1	1	1	1	1		3	🟨	🟨
	3	3	3										3	🟨	🟨
												3	5	🟨	🟨
							2	3	4	2	1	3	🟨	🟨	🟨
				3	3	3	2	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
		2	2	6	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
		2	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
		2	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
	3	3	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
	3	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨

Clicking?

PyAutoGUI - give your scripts a mouse

Easy to customize

Automate the Boring Stuff with Python


AUTOMATE THE BORING STUFF WITH PYTHON
PRACTICAL PROGRAMMING FOR TOTAL BEGINNERS
AL SWEIGART
no starch press

Do what I do but better

- ✅ Look at the board
- ✅ Parse into a model
- ✴️ Identify safe moves
- →
- ✅ Make those moves
- ✅ Repeat until win or loss

Get an MVP that looks like this…

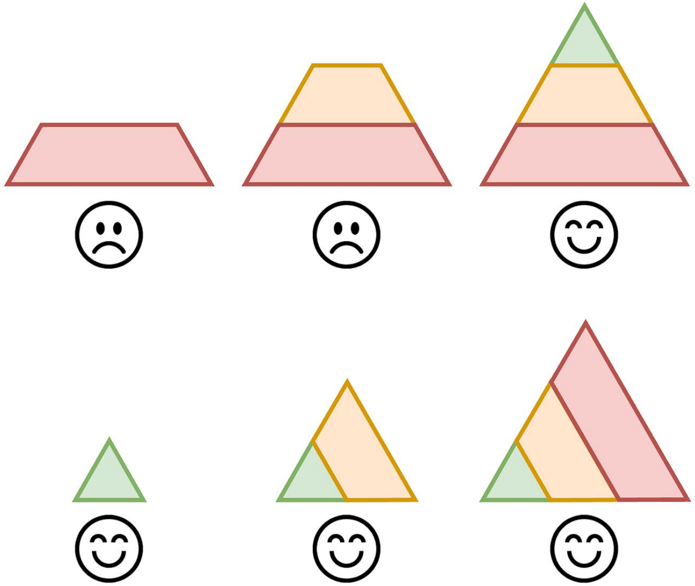
[flashing lights warning]

Minimize your MVP

This fails to execute a real strategy

vs

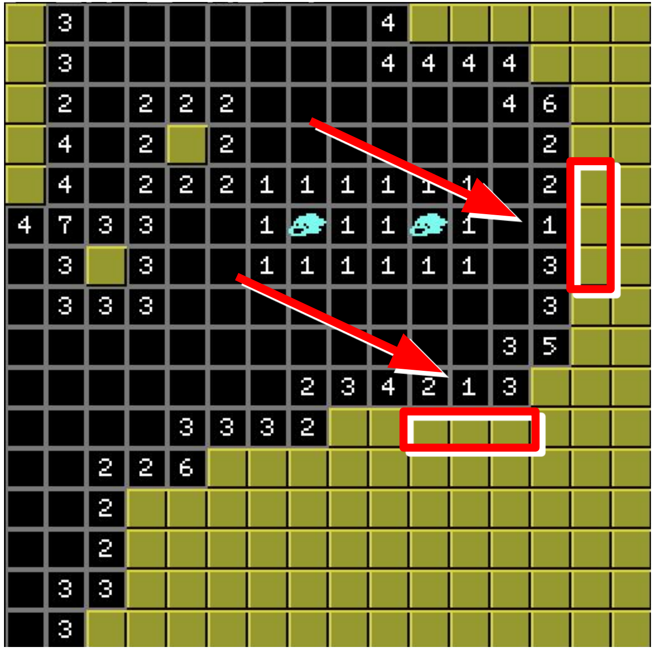
This succeeds at playing the game



Revisiting strategies

If a square's number is equal or lower than the current level, click its **neighbors**.

Red boxed squares *cannot* have a monster above level 1



Revisiting strategies

Pseudocode:

```
for row:
    for column:
        if grid[row, column] ≤ level:
            for neighbor:
                safe_to_click.add(neighbor)
```

Revisiting strategies

Pseudocode:

```
for row:
   for column:
         if grid[row, column] ≤ level:
            for neighbor:
                  safe_to_click.add(neighbor)
```

It’s functional, but…

- Look at the board
- Parse into a model
- **Identify safe moves** ← **My original question**
- Make those moves
- Repeat until win or loss

2: You can’t waste time if you’re learning

NumPy: The MVP’s MVP

NumPy: Accessible and powerful

Fast array/matrix manipulation

“how can I solve x” vs. “study this whole library”

Revisiting strategies

Can this be rewritten?

```
for row:
    for column:
        if grid[row, column] ≤ level:
            for neighbor:
                safe_to_click.add(neighbor)
```

Revisiting strategies

Can this be rewritten? Yes!

```
low_neighbors = grid <= level
for x, y in np.argwhere(low_neighbors):
    for neighbor:
        safe_to_click.add(neighbor)
```

Revisiting strategies

Can this be rewritten? Yes!

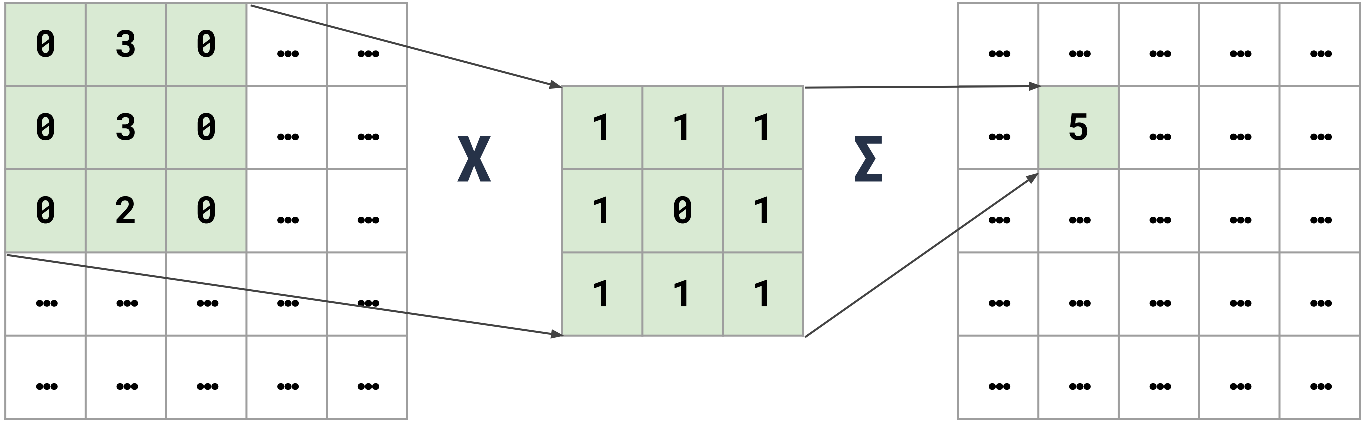
```
low_neighbors = grid <= level
for x, y in np.argwhere(low_neighbors):
    for neighbor:
        safe_to_click.add(neighbor)
```

“Convoluted?” It’s actually quite simple

Do something based on neighbors: **convolution**

Image processing / Machine learning

scikit-learn works (and uses NumPy!)



0	3	0	…	…
0	3	0	…	…
0	2	0	…	…
…	…	…	…	…
…	…	…	…	…

X

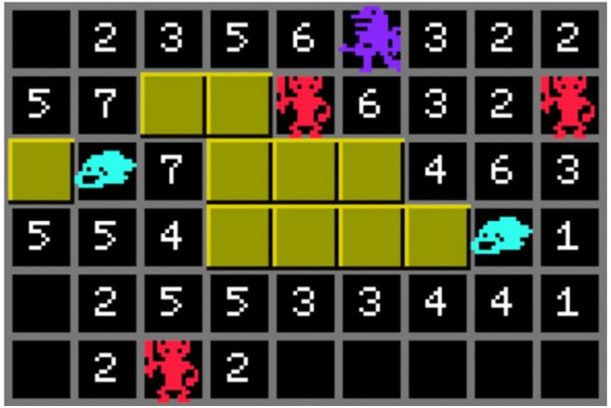
1	1	1
1	0	1
1	1	1

Σ

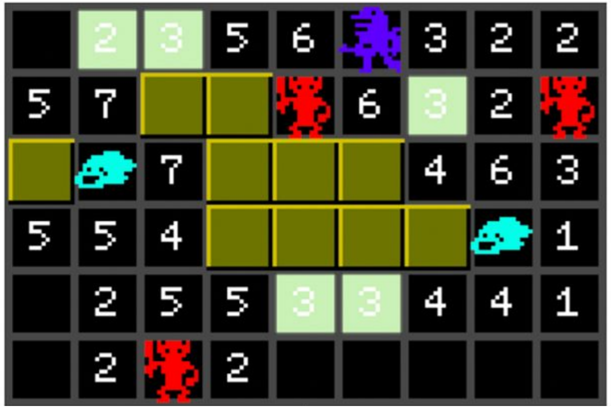
…	…	…	…	…
…	5	…	…	…
…	…	…	…	…
…	…	…	…	…
…	…	…	…	…

Putting it all together

```
low_neighbors = known_neighbors <= game_level
```



	2	3	5	6		3	2	2
5	7	🟨	🟨		6	3	2	
🟨		7	🟨	🟨	🟨	4	6	3
5	5	4	🟨	🟨	🟨	🟨		1
	2	5	5	3	3	4	4	1
	2		2					

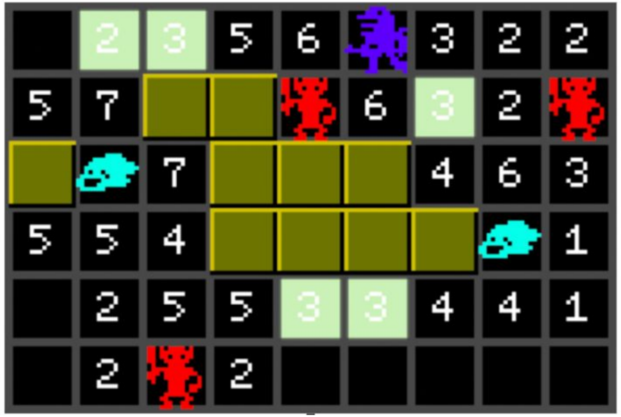


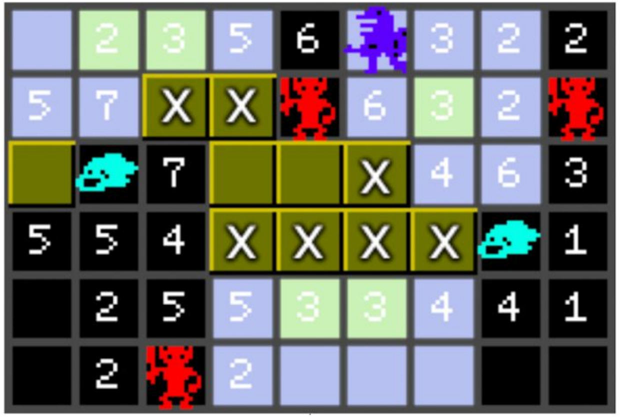
	2	3	5	6		3	2	2
5	7	🟨	🟨		6	3	2	
🟨		7	🟨	🟨	🟨	4	6	3
5	5	4	🟨	🟨	🟨	🟨		1
	2	5	5	3	3	4	4	1
	2		2					

Putting it all together

```
kernel = [[1, 1, 1], [1, 0, 1], [1, 1, 1]]

has_low_neighbor = convolve2d(low_neighbors, kernel) > 0
```





Putting it all together

```
low_neighbors = np.array(known_neighbors) <= game_level

kernel = [[1, 1, 1], [1, 0, 1], [1, 1, 1]]

has_low_neighbor = convolve2d(low_neighbors, kernel) > 0

safe_to_click = np.all(has_low_neighbor, game_unrevealed)
```

No loops → 10x speedup!

[also flashing lights again]

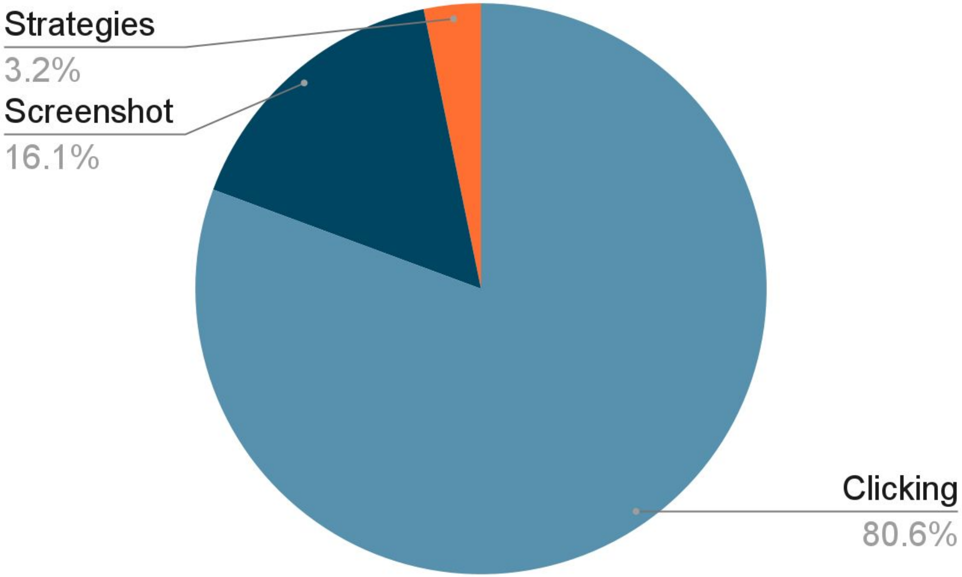
I/O is blocking

…doesn’t change times

Eliminate thinking time

→ 3% overall speedup

Was this worthwhile?



Just in time learning

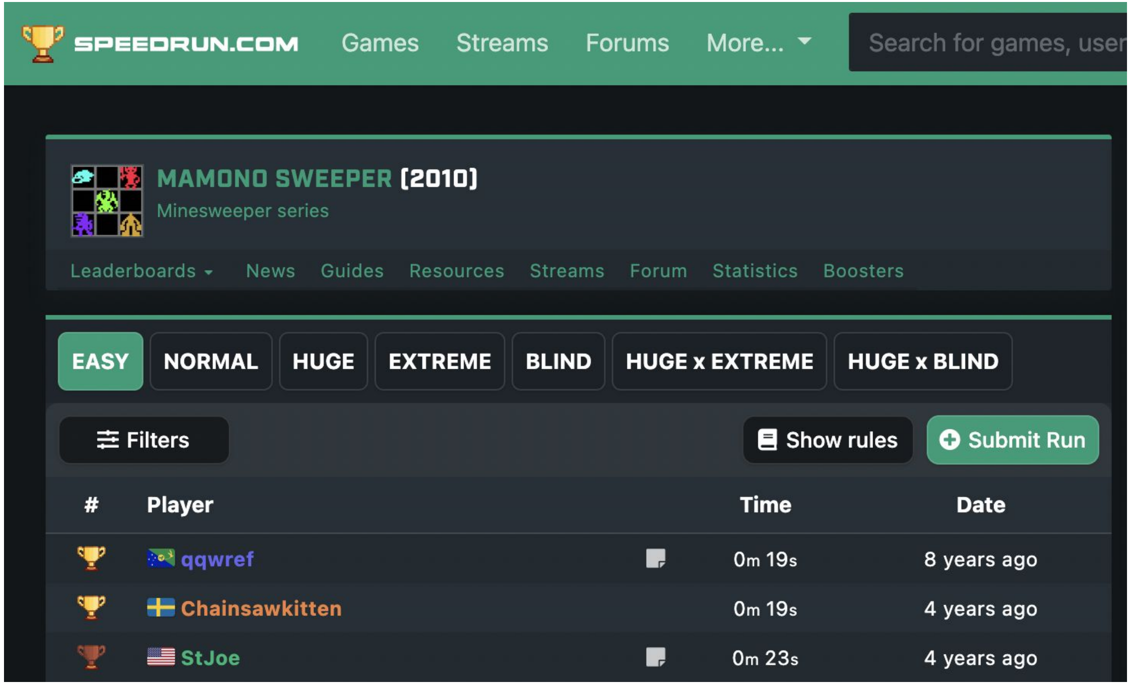
Of course it was!!

1: NumPy and scikit-learn get even more important

2: This is a great way to learn

3: Have a goal in mind

Takeaway 3: Have a goal in mind



MAMONO SWEEPER [2010]

Minesweeper series

#	Player	Time	Date
🏆	qqwref	0m 19s	8 years ago
🏆	Chainsawkitten	0m 19s	4 years ago
🏆	StJoe	0m 23s	4 years ago

Takeaway 3: Have a goal in mind

Difficulty:	Human record (MM:SS):	Computer record (MM:SS):
Easy	0:19	
Normal	1:44	
Huge	6:23	
Extreme	3:43	
Blind	2:06	
Huge x Extreme	12:54	
Huge x Blind	9:59	

*last flashing lights warning

Takeaway 3: Have a goal in mind

Difficulty:	Human record (MM:SS):	Computer record (MM:SS):
Easy	0:19	0:01
Normal	1:44	0:05
Huge	6:23	0:14
Extreme	3:43	0:08
Blind	2:06	0:04
Huge x Extreme	12:54	0:21
Huge x Blind	9:59	0:13

What’s next?

Smarter guessing

- ~1/20 successes for Blind, ~1/100 for Huge x Blind

Two strategies:

- Neural networks or statistics

Guessing Games

Neural Network:

+ Captures intuition
+ Easy to generate data

- May not pay off for effort?

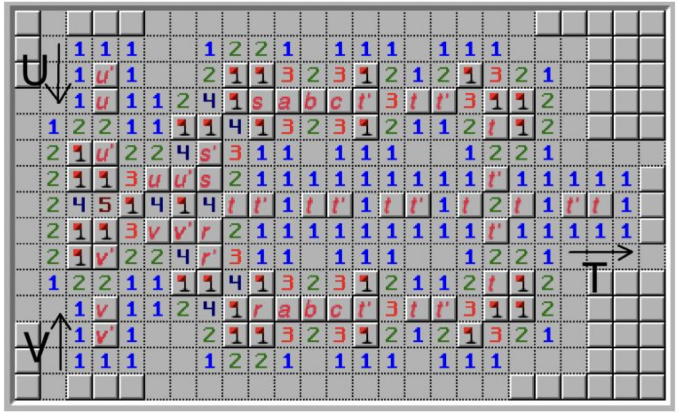
Statistics:

+ Partitioning problem
+ Constraint satisfaction

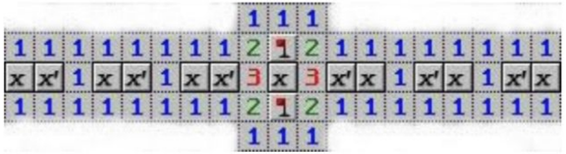
- Hard but guaranteed to work

Not just Hard but Complete

AND Gate



NOT Gate



Finite Minesweeper is NP complete

Infinite Minesweeper is Turing complete

Image from Richard Carini, “Circuits, Minesweeper, and NP Completeness”

Big takeaways

1. Smaller steps are easier to take

2. Learn through optimization

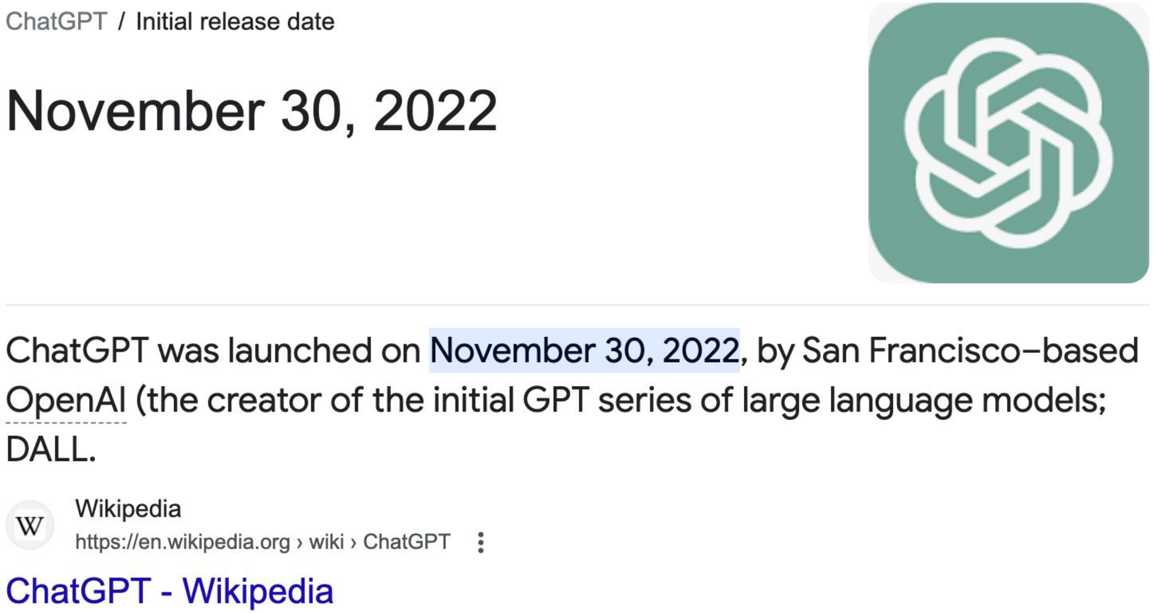
3. Have a goal in mind

MSS screenshots → OpenCV recognition → PyAutoGUI clicking

Why do I feel creeped out?

November 16, 2022: Cute way to automate tricky tasks

Two weeks later…



ChatGPT / Initial release date

November 30, 2022

ChatGPT was launched on November 30, 2022, by San Francisco–based OpenAI (the creator of the initial GPT series of large language models; DALL.

Wikipedia
https://en.wikipedia.org › wiki › ChatGPT

ChatGPT - Wikipedia

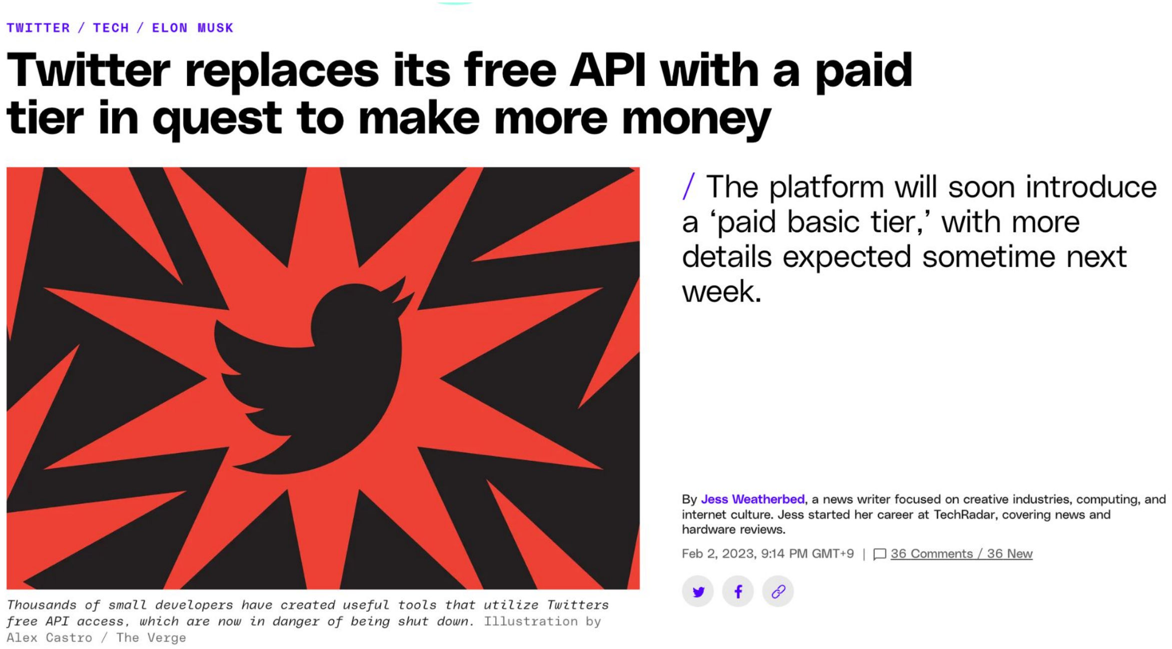
Generalizable automation

Look → Think → Click

Simple to set up, looks like human activity

This has some anti-social potential

“Universal” APIs



TWITTER / TECH / ELON MUSK

Twitter replaces its free API with a paid tier in quest to make more money

/ The platform will soon introduce a ‘paid basic tier,’ with more details expected sometime next week.

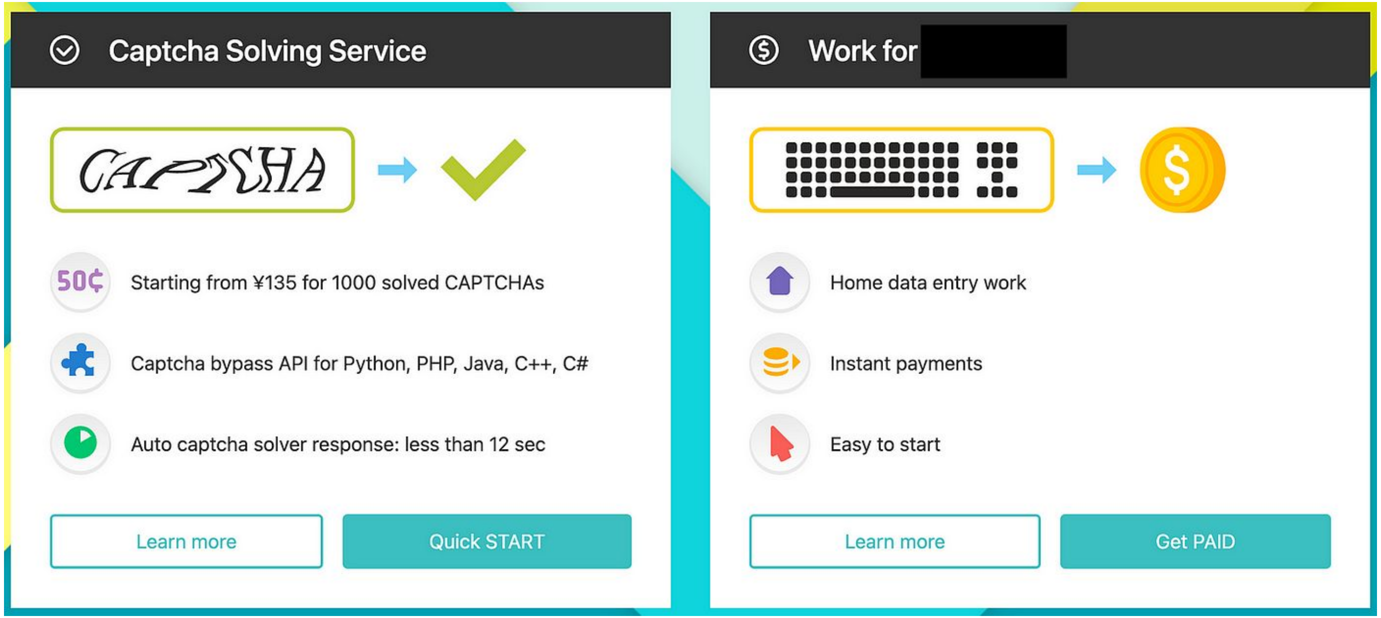
Thousands of small developers have created useful tools that utilize Twitters free API access, which are now in danger of being shut down. Illustration by Alex Castro / The Verge

By Jess Weatherbed, a news writer focused on creative industries, computing, and internet culture. Jess started her career at TechRadar, covering news and hardware reviews.

Feb 2, 2023, 9:14 PM GMT+9 | 36 Comments / 36 New

https://www.theverge.com/2023/2/2/23582615/twitter-removing-free-api-developer-apps-price-announcement

What about captchas?



Captcha Solving Service

Starting from ¥135 for 1000 solved CAPTCHAs

Captcha bypass API for Python, PHP, Java, C++, C#

Auto captcha solver response: less than 12 sec

Work for [redacted]

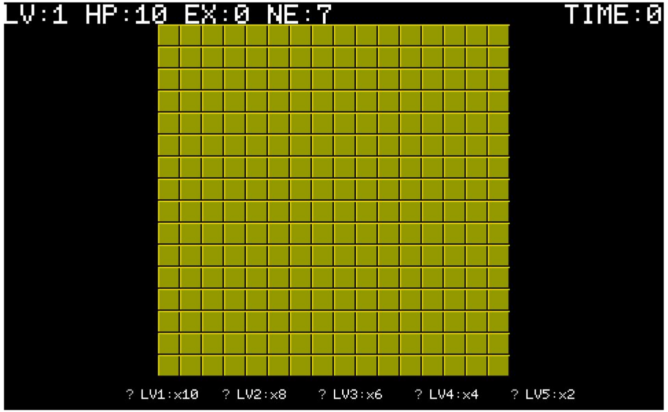
Home data entry work

Instant payments

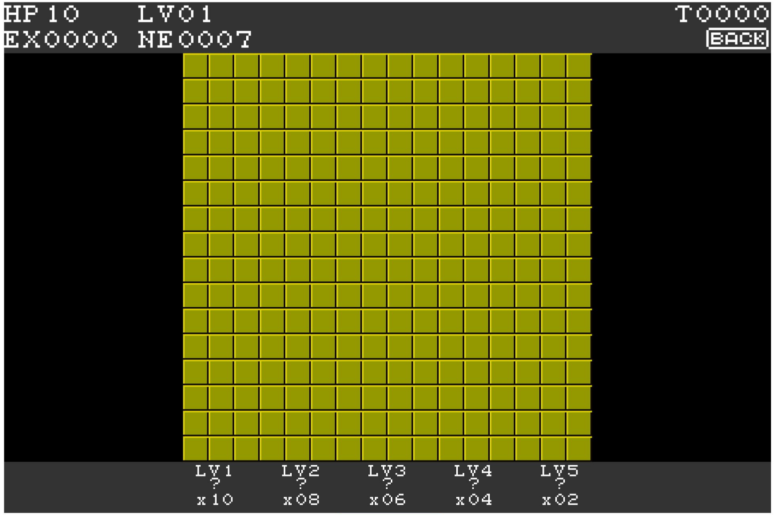
Easy to start

One option: completely overhaul things

Before August 18, 2023



After

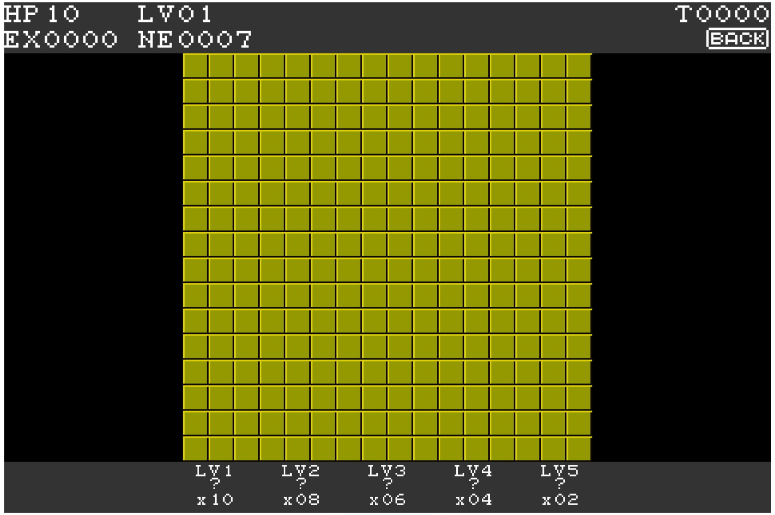


One option: completely overhaul things

- Different landmarks
- New grid widths
- **PyAutoGUI no longer works**

Not a permanent setback

(May be unpopular with users)



What can we do?

As developers:

- Automate what we can
- Sensible human checks
- Limit bad actors

As people:

- Be kind
- Cultivate community
- Make hatred stand out

Thank you!

Any Questions?

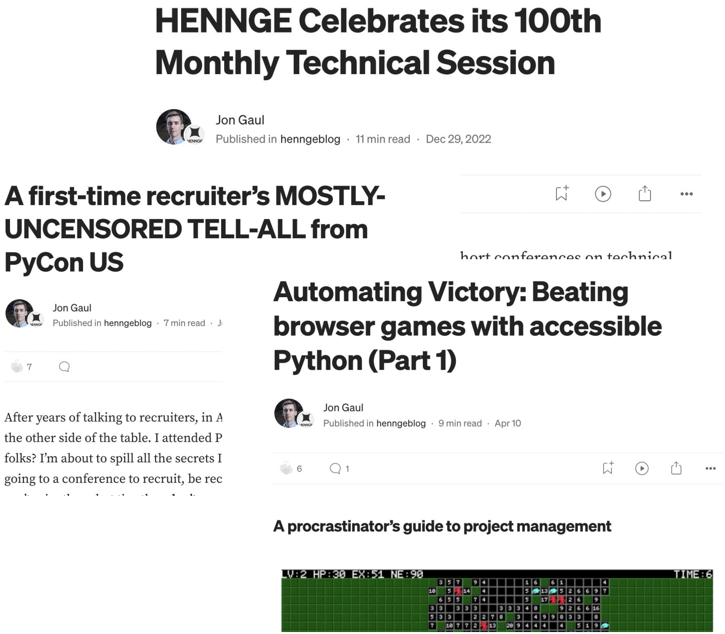
project: https://github.com/jgaul3/TASweeper

email: Jon.Gaul@HENNGE.com

twitter(?): @jon_ghoul

Let’s be coworkers! HENNGE is hiring at https://recruit.hennge.com/en/

Read more of my writing at https://medium.com/henngeblog



HENNGE Celebrates its 100th Monthly Technical Session

Jon Gaul
Published in henngeblog · 11 min read · Dec 29, 2022

hort conferences on technical

A first-time recruiter’s MOSTLY-UNCENSORED TELL-ALL from PyCon US

Jon Gaul
Published in henngeblog · 7 min read · J

7

After years of talking to recruiters, in A
the other side of the table. I attended P
folks? I’m about to spill all the secrets I
going to a conference to recruit, be rec

Automating Victory: Beating browser games with accessible Python (Part 1)

Jon Gaul
Published in henngeblog · 9 min read · Apr 10

6 1

A procrastinator’s guide to project management