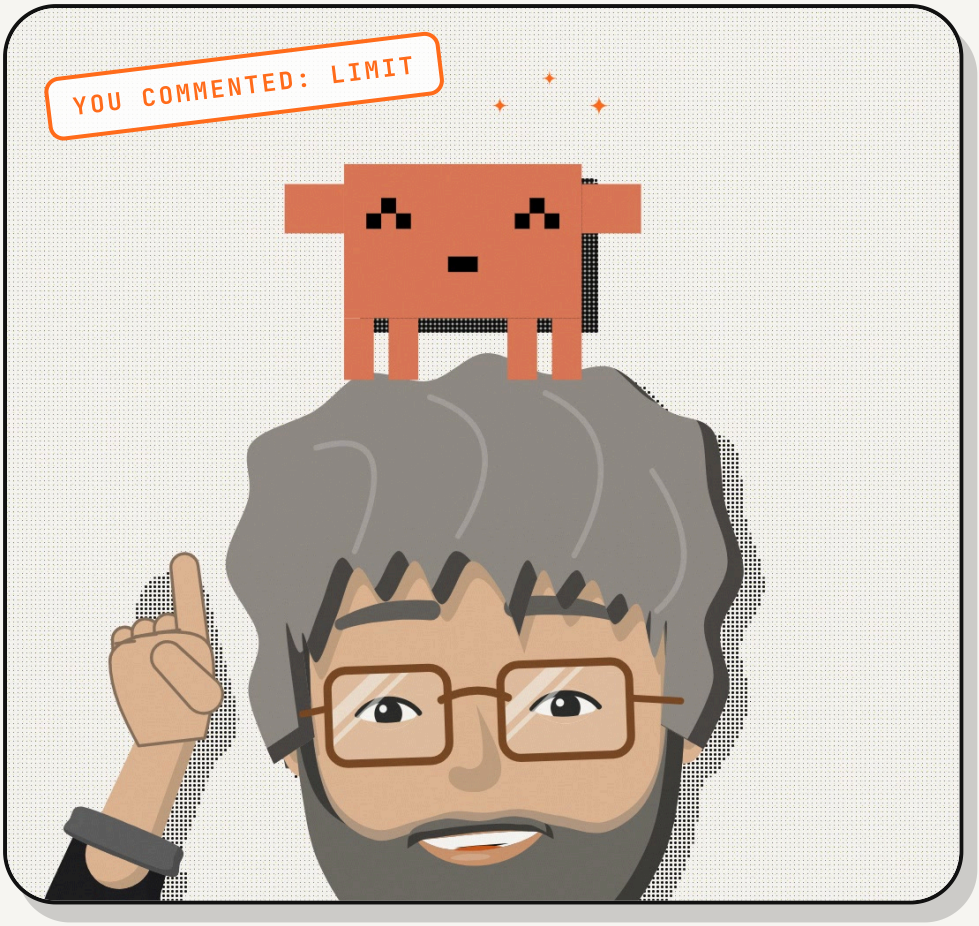
LIMIT

FREE GUIDE



What Arena really costs *your limit*

The full math behind the video: what the skill really does, what one tester measured, where the 39 million tokens come from, why your Pro plan can't carry it, and how to check any hype skill before it eats your limit. With all the sources.

// 01

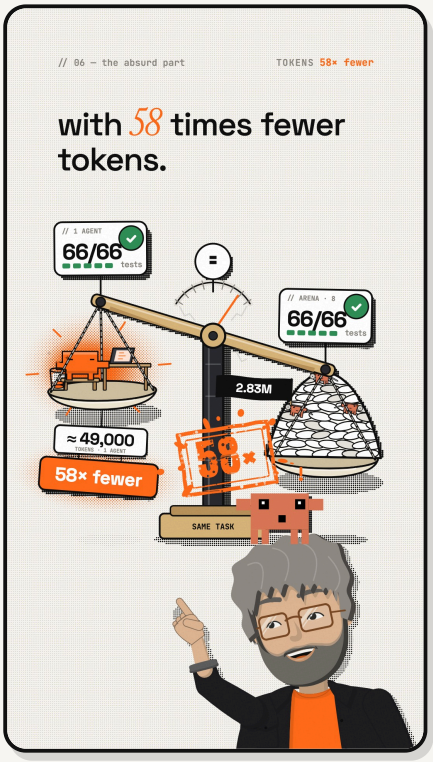
At a glance

QUESTION	SHORT ANSWER
What is Arena?	A community skill on GitHub (Jakeschincariol/arena-skill), not an Anthropic feature
What does it do?	Starts 100 sub-agents by default that attack each other until one answer is left
How many calls?	595 AI calls in 70 waves of 10, for a single task (per the README)
Measured	8 agents = 43 calls ≈ 2.83M tokens (one tester, one test)
Scaled up	100 agents ≈ 39M tokens (estimated, linear extrapolation)
The result	A single agent passed the same 66 tests with about 58× fewer tokens
Your plan	Pro and Max have a five-hour limit and a weekly limit, shared with the Claude chat

// FROM THE VIDEO







// 02

What the hype posts don’t tell you

Arena is real, and the idea sounds great: 100 versions of Claude get the same task, each with a different strategy card, they attack each other in pairs, a judge scores, the loser is out. 100 become 50, then 25, 13, 7, 4, 2, 1. What the posts leave out:

- **It's not an Anthropic feature.** The skill comes from a single developer, is MIT licensed on GitHub and is not part of Anthropic's official skills. Nobody at Anthropic has reviewed it.
- **The default is the maximum.** Without a flag, `/arena` starts with 100 agents. According to the skill's own instructions, Claude states the size in one line ("100 agents, 7 rounds, 595 sub-agent calls") and starts, without asking.
- **It can suggest itself.** The skill is described so that Claude can reach for it without a command, for example when you write "try again" or "bad answer". In that case it is supposed to ask first. Read that question carefully.
- **Hundreds of approvals.** Every sub-agent writes a file into the `.arena/` folder. In the default permission mode that means one approval per file, hundreds on a big run.
- **It takes a while.** Claude Code runs at most 10 tool calls at once by default. That's why the arena runs in waves, 70 of them on a full run.



THE MENTOR SAYS

A skill with 1,000 stars is popular. That says nothing about what it costs you per prompt.

// 03

The full math

The developer publishes the number of calls himself, in the README under "What it costs". The tokens per call aren't listed there, because they depend on your task: "Every call reads the task and one or two solutions, so the bill grows with the size of the task."

AGENTS	ROUNDS	CALLS	WAVES OF 10
100 (default)	7	595	70
64	6	379	49
32	5	187	28
16 (`--quick`)	4	91	16
8	3	43	10
4	–	19	–

Source: the repo's README; the 4-agent row comes from the measurement in issue #1.

What one tester measured

In issue #1 a user (mersad31) measured the cost before using the skill for real work. The task: a Python calculator with clear safety rules, plus 66 hidden tests and a blind judge.

RUN	CALLS	TOKENS
A single agent	1	≈ 49,000
Arena with 8 agents	43	≈ 2.83M

That's about 66,000 tokens per call. The most expensive phases were attack and defend, together about 65% of the bill.

Where the 39 million come from

The math is simple and explicitly an estimate: 2.83M tokens for 43 calls, scaled up to 595 calls, gives about 39M tokens. For `--quick` (91 calls) the tester gets about 6M. That holds for exactly this task. A longer task costs more, a very short one less.

For comparison: the AI council from my next video (roast-room) states 21 sub-agent runs and "about 1M tokens" for a full council in its own instructions.

What came out of it

The single agent and the arena winner both passed all 66 of 66 tests. For tasks a single agent already solves well, the arena cost about 58 times as much for the same functional result, according to the tester. He found one advantage with a hostile edge case that the arena winner caught cleanly. On top of that: 6 of 8 solutions ended up with the same approach, and you only get one answer back at the end.

// 04

Why your Pro plan can’t carry it

According to Anthropic, Pro and Max have two limits: one per five hours and one per week. Both apply together to Claude in the browser, in the app and in Claude Code. Everything you do there draws from the same pool, every sub-agent included: "The subagent's own requests still draw on your usage", says the Claude Code documentation.

Anthropic doesn't publish how many tokens fit into a Pro plan. So there is no exact "Pro = X tokens" calculation, and anyone who shows you one is estimating. What there is, are two clear signs:

1. **The user report in issue #1:** one prompt, 16 minutes of runtime, then all tokens were gone, and an answer never arrived.
2. **The order of magnitude:** a single agent needed about 49,000 tokens in the test. The full arena is estimated at roughly 800 times that.

When the limit hits in the middle of a run, the tournament stops. You still paid for the waves before. That's why I said in the video: on a Pro plan, your limit is empty before the arena is done.



THE MENTOR SAYS

The dangerous part of this skill isn't the idea. It's the default of 100 and that it starts without asking.

// 05

Keep an eye on your limit

WHERE	WHAT YOU SEE
`/usage` in Claude Code	bars for your plan limit, plus a breakdown of how much went to skills, sub-agents, plugins and MCP servers (`d` = 24 hours, `w` = 7 days)
`/status` in Claude Code	your remaining allocation, per Anthropic's help center
claude.ai → Settings → Usage	your limit in the browser, plus usage credits and a monthly spend limit
Warnings	Claude tells you when you're getting close to a limit

Claude Code · Manage costs

the official docs on `/usage`, sub-agents and token usage

code.claude.com ↗

Claude Code with Pro or Max

five-hour and weekly limit, what happens at the limit

support.claude.com ↗

My advice: if you turned on usage credits, which let you keep working past your plan limit, set a monthly spend limit in claude.ai under Settings → Usage before you try any multi-agent skill. Without a limit, `/usage` shows "Unlimited" there.

// 06

How to check any skill before you install it

You don't need to code for this. Five minutes are enough:

1. **Who's behind it?** An official vendor or a single person? Is the skill from Anthropic or only in a private repo?
2. **Is there a cost section?** Good skills tell you how many calls they make. Arena even does. Read it.
3. **How many sub-agents does it start?** Search the `SKILL.md` for "Agent", "sub-agent", "parallel" or "wave". Every sub-agent draws from your limit.
4. **What's the default?** Does it start at the maximum or at a small size? Does it ask before it starts?
5. **When does it start itself?** The description at the top of the `SKILL.md` decides which sentences make Claude suggest it on its own.
6. **What do the issues say?** Click "Issues" in the repo. Complaints about cost, endless loops or empty limits show up there first.
7. **Test small, then measure.** First a tiny task, then open `/usage` and check what percentage it cost.

Let Claude check the skill without installing it

```
Don't install anything. Read this GitHub repo: [repo link]. Open the README and
SKILL.md and tell me: 1. How many sub-agents or model calls does the skill start per
run by default? 2. Which options make it smaller? 3. Does it ask before it starts, or
does it start right away? 4. Which sentences make it start on its own? 5. What do the
open issues say about cost or limits? Answer with a short table and a clear
recommendation for a Pro plan.
```

```
I'm on a Pro plan. Before I use the skill [name]: roughly estimate how much of my
five-hour limit one run with the default settings could cost, and tell me honestly if
you can't know. Then suggest the smallest setting I can test it with.
```

// 07

If you still want to test Arena

Then never with the default. Show the size first, it writes nothing and costs nothing:

```
python3 skills/arena/bracket.py plan --agents 4
```

The command shows rounds, calls and waves for the number you pick. With `/arena --agents 4` it's 19 calls according to the measurement, with `/arena --quick` (16 agents) 91. Keep the task short and clear, because every call reads it again. Afterwards open `/usage` and check what the run cost.

SOURCE	WHAT'S IN IT
arena-skill	README with the cost table and instructions
Issue #1	measurement (2.83M tokens, 66/66 tests) and the user report
PR #2	proposal to show a token estimate and ask first (not merged as of 27 Oct 2026)

// 08

Checklist before the next hype skill

- [ ] Is the skill official or from a single person?
- [ ] Read the cost section in the README
- [ ] Found the default number of sub-agents
- [ ] Checked whether it asks before it starts
- [ ] Read the open issues about cost and limits
- [ ] Set a spend limit for usage credits
- [ ] Ran a tiny task first, then checked `/usage`
- [ ] Didn't mistake stars and views for quality

// 09

My tip

More agents don't mean more truth, mostly more usage. Before you install a skill that promises "100 Claudes", ask: would one well-briefed Claude manage this too? Usually yes. My next video shows an AI council with around 20 calls instead of 595.

All numbers, links and commands checked against the original sources (README, SKILL.md, issue #1, PR #2, Claude Code docs, Anthropic help center) on 27 Oct 2026. Estimates are marked.



THE MENTOR SAYS

Follow me, and warnings like this reach you before the next hype eats your limit.

— @reposprites

Want to make animated videos like this too? Visit my website and get the reel engine.

reposprites.com ↗