My name is Cale. I am an AI Productivity Researcher who just recovered from 5 years of debilitating Post-traumatic Stress Disorder using a technique I developed called the A\* (A-Star) Startup Method, or A-Startup Method for short, and I am You Language (IMUL) to maximize productivity and learning rates with the power of the pen, positive psychology, neuroscience, and AI. I am seeking college students and startup founders who would be willing to participate in a study on changes in flow-state time, Issue and Task completion rates, weight, mood, and GPA. You can think of your college career as a startup because the timeline and workloads are very similar. By participating in this study, you get free startup consulting and Git training and possible side-gigs.

The A-Startup Method is based on the physics principle that resistance over time equals work, that work is a cost to accomplish goals, and that in order to maximize productivity and happiness you must minimize cost through minimizing resistance and maximize reward through maximizing health, that the default assignment of weights for cost and reward are one unit, or in other words [A\* AI](https://www.wikiwand.com/en/A*_search_algorithm).

All you need to get started are 3 notebooks (preferably the fine-grid graph composition that are $0.79 each at Walmart) and clipboards. The logs are used to compartmentalize your A-Startup Operating System into three threads, a Primary Thread, a Secondary Thread, and a Self-care Thread. The goal of the Self-care thread is to prioritize the optimization of your mental and physical health first in order to optimize the Secondary and Primary threads. The goal of the secondary thread is to optimize the primary thread by eliminating resistance in the system during the Hour of Seconds (HoSe), and the goal of the primary thread is to maximize flow state time. The HoSe is an hour dedicated to your Secondary Thread two times per day. The HoSe is split into one Day-time session and one Night-time session, a flow-state optimization technique called the Night and day productivity pattern. Resistance is defined as Issues, Tasks, and Mental obstacles that reduce the amount of time you spend in the flow state. Resistance over time is defined as work, and work is defined as a cost to accomplish goals and cost to your health.

The HoSe begins each early every weekday morning (or night-owl equivalent) without snoozing. You spend 20 minutes wake, taking a shower (preferably ending in cold water), getting dressed, and getting ONE cup of coffee or tea ready. You then spend 20 minutes doing a Self-care Journal where you also document Positive Psychological Rewards for complying with the goals you set and Positive Social Norms you define as sins to violate or that make others not look up to you. These are contracts that include I wills and I will nots (@see example below).

After the journal you do a 20-minute walk or cardio workout or eliminate resistance from unclean workspaces if it's raining, followed by 20 minutes of sit-ups/push-ups/weights/etc, to which I have an IMUL Expression (i.e. an optimized A-Startup stack machine operation that pops everything off the stack without a stack overflow) for this. After the warmup you then spend EXACTLY one-hour cranking on your Secondary Thread in order to maximize progress on the most challenging issues in the flow state. This is called the Day-time session. You then eat breakfast, meditate, play music, or read until you go to work (which is your Primary thread if your work full-time). By eating breakfast after the HoSe it has the effect of classically conditioning the HoSer to HoSe when they get hungry and to further reinforce the habit through through time stamping the Positive Psychological Rewards for achieving your goals. You can also use the Self-care Log to document your weight and what calories you eat.

// Here is an example of I am You Language (IMUL), which is duck-typed and

// based on Doxygen, Markdown, and Script2, which uses C++-style

// comments.

>SelfCare @6:05PM

$Shower @18

$Coffee @19

>@20 Good morning! Here is my morning log! In IMUL '>' pushes items onto the stack and '<' pops them off. `@9` means 9 minutes after that last recorded hour. ASCII Timestamps cascade through objects but can be out of order. The way IMUL works to enhance productivity is by providing an out of order task check off method using the '@' timestamps. To schedule an event use the '$' symbol. Using '$' allows you to use a one-line rather than pushing another tree node on the stack.

>PostivivePsychologicalRewards

$Reward: play music.

$Reward: play Pokemon with my son.

<@30

>PostivieSocialNorms

$ I will not be rude to people on the internet.

$I will not openly insult religion or those engaging in identity politics.

$I will give the appearance that I am humble.

$I will not tarnish my public image.

$I will be patient with people.

$I will not look down upon those with less experience than me.

<@35

>Workout @38

$ShortHillClimb @40

$Strech @7:06AM

$LegAndSitups @8

$HandleBarPushUps @10

$LegAndSitups @22

$HandleBarPushUps @16

$DumbellCurlsAndPress @14

$BentOverRows @18

$DumbellCurlsAndPress @20

$BentOverRows @12

<@25

$Breakfast: Beagle with cream cheese. @30

$Lunch Subway sub of the day. @12:30PM

>@3:30PM

$Schedule doctors appoint for my son.

$Schedule dentist appoint for my son.

<

$Dinner Pizza Slice x3 @5:30PM

><@8PM

Essentially this is a graph-based to-do list that connects to the Internet-of-Things and you can run Python or Julia in Markdown blocks or control Jupyter notebooks. We then use the ASCII Timestamps to document when you started and finished tasks; which is the lifeblood of the A-Startup Method. We then use this information with some AI to maximize your productivity and learning rate.

In order to maximize Issue-completion rate you need the Night-time session, typically 8:30PM-9:30PM. This is the time where you take care of random technical glitches and learn; reading is highly encouraged at this time. After the night-time session you then spend a half hour cleaning your workspace and house, you leave your Secondary log on your desktop open ready to go to town on with a to-do list of issues, before going to sleep and getting good rest. This will maximize your learning rate by making non-confused neurons that fire together wire together, allowing your brain to get some time off the screen before sleep, minimize resistance in the secondary thread, and it will allow you to learn and work in your sleep; which is the best learning you can do.

You'll see right away that if you don't prepare for the HoSe the night before you can't get very far because you get caught up doing technical tasks that shut the show down left and right. If you do not do the Night HoSe then you don't have to feel bad, you can still do the setup hour before you work out. Just do your journal in the morning, power on your workstation, spend an hour getting a to-do list of tasks, do your workout, and then do the Day HoSe. This however will reduce the amount of time spent in the flow state on the Primary thread. If you miss the HoSe, just proceed without guilt and clock off at 5:00PM and start again at 8:30PM. The key is to forget, prepare, and each time get a little bit better at hitting the 4x20min in the morning before the HoSe.

The reason for the limit of 1 hour is that it's the optimal time for your brain to learn without over-learning in a confusion pattern (i.e. over-fitting). It also allows you to think about the secondary thread while working on the primary thread, and the use of 3 logs achieves the goal of rapid documentation of ideas that assist in maximizing flow-state time in the primary and secondary threads when used with IMUL to convert your ideas to Markdown.

The system is based on what is called Issue-drive Development, so research GitHub Issue Tracking System (ITS) if you're not familiar. If you could imagine a stack machine with to quick-to-write minute timer system along with use of Git branches in format IssueNumber (Example: Issue1, Issue2, ...), TaskNumber (Example: Task1, Task2,...), and Seam Trees (advanced topic) to create a basic compiler to prevent stack overflows and ensure that tasks pushed on the stack get popped off and to facilitate context-specific activities.

All I need you to do is set your projects up using Issue-tickets generated either on paper on through the ITS and make your commit messages include the #IssueTicketNumber. We then want to use the comments in the Issue Ticket to documents Tasks using #A format. For example, an Issue Ticket could read like:

For example:

>SecondaryThread @1-9-2019

$coffee @8:20

>script2 @24PM

>Issue52

// Note: We typed Issue52 into GitHub and made a branch in our repo.

>@25 We just pushed the script2 repo on the stack followed by Issue52, which maps to #52 in the GitHub Issue Tracking System. This is not a Task so this block will not appear on the issue ticket chat log.

<@26

>#A @27 This is a Task so it will show up in the GitHub Issue Ticket chat log. The unit tests are still broken from the botched Unicode update. This is embarrassing. Please pull the repo because the compile errors are not fixed, set the SEAM to \_0\_0\_0\_\_03 and work on the seam test.

<

>#B @30 Accomplish ABC goal.

@cost := 2

@reward := 3

@pre Fix compiler errors in\_0\_0\_0\_\_02.

Here is the body of the comment. Each Task gets one comment. You may use Markdown in these blocks.

## TaskB Sub-goal

Here is an example of how you would run code in other languages using markdown. All you have to do is make a single-line comment with the keyword Do with no spaces like `#Do` in Python or Julia.

```Julia

#Do

println (‘Hello world!’)

```

<@35

>Issue32 @40

>Task1 @44 Accomplish XYZ goal.

><@4:00PM

It then looks like this in GitHub:

Issue52 Title: Fix bug in the XYZ system. #1

Issue52 Body: Foo are you.

Comment1

```

>#A @2019-1-8@8:27AM

```

\*Cost:\* \*\*1\*\* \*Reward:\* \*\*1\*\*

This is a Task so it will show up in the GitHub Issue Ticket chat log. The unit tests are still broken from the botched Unicode update. This is embarrassing. Please pull the repo because the compile errors are not fixed, set the SEAM to \_0\_0\_0\_\_03 and work on the seam test.

EndComment1

Comment2

```Script2

> #A @2019-1-8@8:30AM

```

\*Cost:\* \*\*1\*\* \*Reward:\* \*\*1\*\*

Accomplish ABC goal.

Here is the body of the comment. Each Task gets one comment. You may use Markdown in these blocks...

```

>#a @2019-1-8@8:35AM

```

\*Cost:\* \*\*1\*\* \*Reward:\* \*\*1\*\*

Here is a sub-task 52.A.a that runs some Julia code using markdown. All you have to do is make a single-line comment with the keyword Do with no spaces like `#Do` in Python or Julia or `//Do` in C++.

```Script2

>>> Hello world!

```

```Script2

<#a @2019-1-8@9AM

<#A @2019-1-8@9AM

```

EndComment2

The pen-and-paper stack machines when combined with IssueTicket tags then optimize your development time help you keep focused on a single issue in order to minimize Ticket Completion times. The task system seamlessly integrates into your GitHub flow and directly translates your paper log into the Issue Tracking System, including drawings and code snippets. It's a night and day difference when you write down what stacked state you’re in, and to isolate larger issues into smaller tasks to help keep the ball moving.

The final element is the daily, weekly, and monthly log review and daily shutdown procedure. Each day you must review the previous day's log, and each week, you must review the logs from the week before, and review the entire month's logs on the first of each month. This allows neurons that fire together to wire together. At the end of each day you must shutdown, clean up, and get ready for the next day's primary thread. The time you need for this depends on your job, but typically it's less than one hour. This again uses the night-and-day principle where you focus on random technical tasks that create resistance in the system. It is more important that you shut down at 5PM (or night-shift equivalent) and take a break from 5:00PM-8:30PM and take the weekends off because this allows your brain to shut down for a few hours and return to normal life so that you can relax and take care of eliminating resistance in your home life.

The schedule is slightly different on Friday. On Friday you must do your night HoSe from 4PM-5PM and have your to-do list ready and prepped with all your homework done ahead of time like an A grad student, ready to HoSe on Monday with your grove on like a HoSer. And if you get all your work done, you get to go out dancing, spend time with your partner(s), and MAXIMIZE the time with your friends and family.

If you would like to participate in the A-Startup Productivity Study, the world could really use your help creating open-source IMUL Expressions to maximize your startup's chance of success, GPA, and health with the power of the A\*. Please email me at cale.mccollough@gmail.com so we can get a meetup going on. Thanks.