How to Teach Engineering

Sep 15, 2024 • Christopher Kalitin

I've started university. I have a lot of ideas on university that'll probably evolve over my time here, might as well write them down in the early weeks to be able to look back on this in the coming

years.

The Problem

I have an introduction to engineering class. I'm in a group of 5 and our purpose is to design a cardboard chair. We brainstorm cardboard chair ideas. We take quizzes on how to brainstorm. Then take more quizzes on how to pick which idea to take. Then finally after 4 weeks maybe we get to actually pick one of our own ideas. But it's very important that before that we play with paper for two hours and carefully observe how paper bends when we play with it. This is a requirement for playing with cardboard later in the class.

Who the actual fuck teaches engineering like this?

Ah yes make me wake up at 6 am every morning to take your quiz on how to play with cardboard. You haven't even given us cardboard yet, just paper. We have to work our way up to cardboard? really?

Make us write 3000 words over a month before you let us play with cardboard. I'm going to give my 2 year old cardboard out of spite. Watch him build a better chair at 3 than your months of lead up to giving 18 year olds cardboard.

The Real Process

I've had a tiny bit of experience with engineering projects.

This experience - and importantly comparing it to how my peers think about engineering - has illustrated to me exactly how useless this APSC 100 class is.

Engineering processes are emergent. You can't teach programming by focusing on syntax for two months before actually writing your own programs. Put ideas before implementation skills.

My intro to C programming class is only slightly better than intro to engineering. At least we build. It may be building a single function to output the value of x * y, but it's something. Afterwards you can show your classmates your 2 line solution versus their 10 lines and brag about how you're a better programmer. No one in the class has any idea what projects they want to make, but we at least get to build.

Out of this process of building, you start to understand why you actually do need to understand syntax and proper programming conventions. Then, you'll gladly read the textbook and learn the proper methods - you see the exact manifestations and efficiency gains by doing things the proper way.

Instead of this exciting process of exploration and figuring things out yourself - literally the process of science - you are forced to write 1000 words and with your group discuss the stakeholders for your cardboard chair.

The process of brainstorming -> evaluating -> building (and iterating back to any given step) is very useful. Let us figure that out ourselves.

The ideal of an engineer is someone who can turn dirt into value. How will you ever be able to do anything new if you're told everything that you must do by an authority?

A Beacon of Beauty in a Sea of Depressed Teenagers

I applied to the UBC Solar design team. Did my interview where I asked questions for 2 hours and answer questions for one hour. I was accepted.

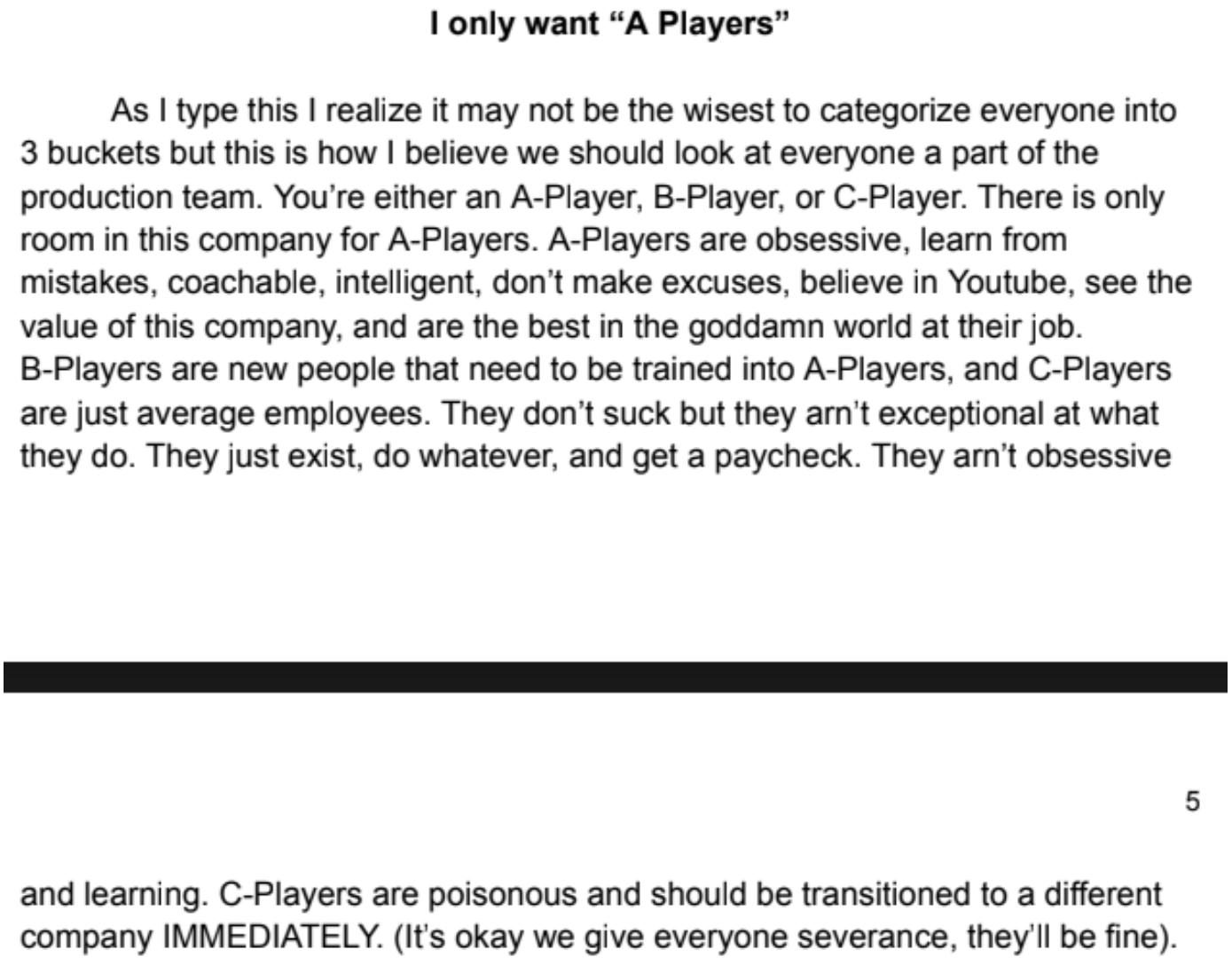
By asking endless technical questions to the team I explored the space of ideas and drew my own insights. This process was the most fun I've had in months. Talking to techincal people and asking endless questions is the most fun and useful things in the world. I didn't fundamentally understand how engines or fuel cells worked before this week. By asking 2 hours of technical questions, I finally grokked it.

All kids are born explorers, it takes years - sometimes decades - for schools to beat this out of them. Science is the process of exploring the space of ideas and understanding the universe. This is fundamentally a self driven and explorative process. The way you get great engineers and scientists is by letting them practice this process, not by telling them to follow your arbitrary rules on how to build a chair.

The vast majority of students probably would have trouble if you simply said, "You have one month to build a cardboard chair that fits these criteria. Go do it." Wait where do we start? Brainstorm ideas? But I don't like my teammates ideas, how do we manage that? EXACTLY! How do you manage this? You will have to figure it out yourself and in doing this, you'll learn far more than being told what to do. ALL KIDS ARE BORN EXPLORERS. Let them explore and find out how to climb a tree without breaking bones.

Like I said above, if your process of planning how to build a chair is really the best way to do it, we will all reverse engineer our way into it and figure it out ourselves. Through this process we will understand a tremendous amount about how to execute on engineering projects.

Life is a Skill Issue



I only want "A Players"

As I type this I realize it may not be the wisest to categorize everyone into 3 buckets but this is how I believe we should look at everyone a part of the production team. You're either an A-Player, B-Player, or C-Player. There is only room in this company for A-Players. A-Players are obsessive, learn from mistakes, coachable, intelligent, don't make excuses, believe in Youtube, see the value of this company, and are the best in the goddamn world at their job. B-Players are new people that need to be trained into A-Players, and C-Players are just average employees. They don't suck but they arn't exceptional at what they do. They just exist, do whatever, and get a paycheck. They arn't obsessive

5

and learning. C-Players are poisonous and should be transitioned to a different company IMMEDIATELY. (It's okay we give everyone severance, they'll be fine).

Mr Beast's memo/book to new employees got leaked. Everyone has to read this.

Teaching processes makes C tier humans. Making your own processes breeds A tier engineers. God put me on this planet to build, and you're wasting my time by stopping me from exploring the space of ideas.

Everything in life is a skill issue and all skill issues can be solved by increasing your skills. That's why "Skill Issue" is such a powerful term, you are in complete control of whether your problems remain skill issues.

Some of my group members care more about grades than learning engineering. SKILL ISSUE C TIER BEHAVIOUR. You need to learn engineering yourself. It is fundamentally a self driven process of exploration. School isn't trying to help you, the members of the design teams would love to help you, but you need to ask the right questions!

This has been the first in a series of posts I'll probably do on University. My guess is that the principles behind my words will remain true for eternity. This is a one-shot blog post, I've been

inspired by Mr Beasts constant spelling mistakes, just show your raw thoughts and yap. Eventually, your yapping will be intelligent enough that people want to listen.

Subscribe

CKalitin
x.com/CKalitin

Geohot made a blog too. You should be working on hardware