Insights from *Algorithms to Live By* by Brian Christian and Tom Griffiths

"In this book, we explore the idea of human algorithm design—searching for better solutions to the challenges people encounter every day. Applying the lens of computer science to everyday life has consequences at many scales. Most immediately, it offers us practical, concrete suggestions for how to solve specific problems." - Brian Christian and Tom Griffiths

Here are three proven algorithms that efficiently solve the following three problems:

1. Among a group of qualified candidates, who should I hire?
2. Once I hire someone, how should I onboard them?
3. After a trial period, should I continue to invest in the new hire or try someone new?

The Selecting Algorithm

Imagine you have a group of 20 qualified candidates for a position. You don't want to be rash and take the first person you interview, but you also don't want to spend hours interviewing all 20. Luckily, there's a third option: a mathematically validated algorithm called the 37% rule. The 37% rule states that you should evaluate 37% of your options with no intention of selecting any of them (i.e., reject the first 37% unconditionally). After evaluating the first 37%, pick the next option you encounter that is better than the options you've sampled.

When choosing between 20 qualified candidates, interview the first seven candidates (37% of 20 is 7.4) with no intention of hiring those seven. Continue interviewing the remaining 13 candidates and stop the instant you find someone better than any of the seven you initially interviewed.

If you believe there is at least a 50% chance the candidates you initially passed over will accept your offer (they didn't move on while you were sampling and are still interested in the position), then the 37% rule turns into the 61% rule - hold off hiring until you interview at least 12 candidates and if you don't find a better candidate in the remaining eight, go back and pick the best person you interviewed amongst all 20.

Both methods give you the chance to determine your preference during a sampling period and provide a good benchmark for which to base your final decision on. The 37% rule and 61% rule are not guaranteed to get you the result you want, but they do give you the highest probability of attaining a favorable outcome.

The Onboarding Algorithm

When a new computer joins a Transmission Control Protocol (TCP) network, it receives a single packet of information from a computer in the network and responds with an 'acknowledge' packet. With each acknowledgement, the sender doubles the number of packets. However, the moment the receiving computer gets bogged down with information and fails to send an acknowledge packet to the sender, an algorithm called Additive Increase Multiplicative Decrease (AIMD), takes over. The AIMD algorithm cuts the number of packets being sent by half, which frees up space on the network and allows the receiving computer to catch up. Once the receiving computer starts acknowledging information again, the sender can add one additional packet to each new message. When another failure occurs, the AIMD algorithm halves the message again. The authors say **"Essentially, AIMD takes the form of someone saying, 'A little more, a little more, a little more, whoa, too much, cut way back, okay a little more, a little more…'"**

You can use a similar algorithm to rapidly onboard a new hire. When you hire a freelancer, give them a tiny task to complete. After they deliver a satisfactory result, double their workload and keep doubling it until you get back work you are not satisfied with. At that point, immediately cut the workload by half. With this exponential increase in work, you can quickly find the freelancer's capacity. Now, only increase their workload by one unit at a time and give them a chance to build back up slowly. When your new hire eventually reaches the level they failed at before, they'll have far more confidence and will be more likely to surpass that level.

The Switching Algorithm

In the 1970s, a young mathematician named John Gittins was tasked with optimizing Unilever's pharmaceutical investments. Gittins had to determine if Unilever should continue investing in a proven drug or pivot to a new drug on trial. Based on several iterations and months of calculations, Gittins came up with a table of values known as the 'Gittins Index.' The index was used to help Unilever determine if they should pivot to a new drug believed to be 90% as effective and profitable as a current drug by evaluating the number of successes and failures of the two drugs (number of times the drugs were or were not successful at treating a disease) and going with the drug which had a higher corresponding Gittins Index.

You don't need to consult the Gittins Index table to determine if you should switch to a new hire or a new project, you only need to remember this key finding: an untested option with zero successes and failures has a Gittins index of 0.7. Meaning, if a project or person is not meeting your expectations at least 70% of the time, you should consider switching to a new project or person with similar potential.

"The unknown has a chance of being better, even if we actually expect it to be no different, or if it's just as likely to be worse. The untested rookie is worth more (early in the season, anyway) than the veteran of seemingly equal ability, precisely because we know less about him. Exploration in itself has value, since trying new things increases our chances of finding the best." - Brian Christian and Tom Griffiths