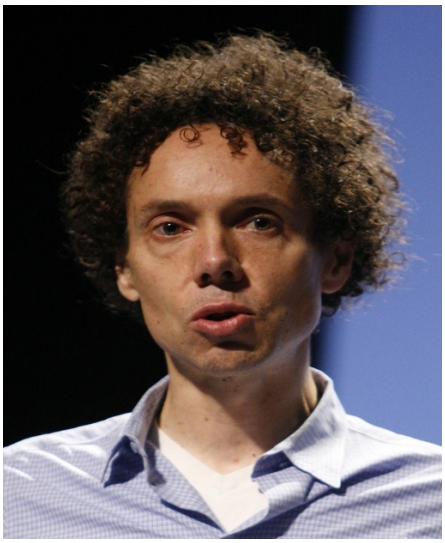
Outliers (Malcolm Gladwell, 2008)



CHAPTER TWO: The 10,000-Hour Rule

"IN HAMBURG, WE HAD TO PLAY FOR EIGHT HOURS."

1.
The University of Michigan opened its new Computer Center in 1971, in a brand-new building on Beal Avenue in Ann Arbor, with beige-brick exterior walls and a dark-glass front. The university's enormous mainframe computers stood in the middle of a vast white room, looking, as one faculty member remembers, "like one of the last scenes in the movie *2001: A Space Odyssey*." Off to the side were dozens of keypunch machines what passed in those days for computer terminals. In 1971, this was state of the art. The University of Michigan had one of the most advanced computer science programs in the world, and over the course of the Computer Center's life, thousands of students passed through that white room, the most famous of whom was a gawky teenager named Bill Joy.

Joy came to the University of Michigan the year the Computer Center opened. He was sixteen. He was tall and very thin, with a mop of unruly hair. Joy had been voted "Most Studious Student" by his graduating class at North Farmington High School, outside Detroit, which, as he puts it, meant that he was a "no-date nerd." He had thought he might end up as biologist or a mathematician. But late in his freshman year, he stumbled across the Computer Center and he was hooked.

From that point on, the Computer Center was his life. He programmed whenever he could. Joy got a job with a computer science professor so he could program over the summer. In 1975, he enrolled in graduate school at the University of California at Berkeley. There, he buried himself even deeper in the world of computer software. During the oral exams for his PhD, he made up a particularly complicated algorithm on the fly that, as one of his many admirers has written, "so stunned his examiners [that] one of them later compared the experience to 'Jesus confounding his elders.'"

Working in collaboration with a small group of programmers, Joy took on the task of rewriting UNIX, which was a software system developed by AT&T for mainframe computers. Joy's

version was very good. It was so good, in fact, that it became and remains the operating system on which literally millions of computers around the world run. “If you put your Mac in that funny mode where you can see the code,” Joy says, “I see things that I remember typing in twenty five years ago.” And do you know who wrote much of the software that allows you to access the Internet? Bill Joy.

After graduating from Berkeley, Joy cofounded the Silicon Valley firm Sun Microsystems, which was one of the most critical players in the computer revolution. There he rewrote another computer language Java and his legend grew still further. Among Silicon Valley insiders, Joy is spoken of with as much awe as someone like Bill Gates of Microsoft. He is sometimes called the Edison of the Internet. As the Yale computer scientist David Gelernter says, "Bill Joy is one of the most influential people in the modern history of computing.” The story of Bill Joy's genius has been told many times, and the lesson is always the same. Here was a world that was the purest of meritocracies. Computer programming didn't operate as an old-boy network, where you got ahead because of money or connections. It was a wide-open field in which all participants were judged solely on their talent and their accomplishments. It was a world where the best men won, and Joy was clearly one of those best men.

It would be easier to accept that version of events, however, if we hadn't just looked at hockey and soccer players. Theirs was supposed to be a pure meritocracy as well. Only it wasn't. It was a story of how the outliers in a particular field reached their lofty status through a combination of ability, opportunity, and utterly arbitrary advantage. Is it possible the same pattern of special opportunities operate in the real world as well? Let's go back over the story of Bill Joy and find out.

2.
For almost a generation, psychologists around the world have been engaged in a spirited debate over a question that most of us would consider to have been settled years ago. The question is this: is there such a thing as innate talent? The obvious answer is yes. Not every hockey player born in January ends up playing at the professional level. Only some do--the innately talented ones. Achievement is talent plus preparation. The problem with this view is that the closer psychologists look at the careers of the gifted, the smaller the role innate talent seems to play and the bigger the role preparation seems to play.

Exhibit A in the talent argument is a study done in the early 1990s by the psychologist K. Anders Ericsson and two colleagues at Berlin's elite Academy of Music. With the help of the Academy's professors, they divided the school's violinists into three groups. In the first group were the stars, the students with the potential to become world-class soloists. In the second were those judged to be merely “good.” In the third were students who were unlikely to ever play professionally and who intended to be music teachers in the public school system. All of the violinists were then asked the same question: over the course of your entire career, ever since you first picked up the violin, how many hours have you practiced?

Everyone from all three groups started playing at roughly the same age, around five years old. In those first few years, everyone practiced roughly the same amount, about two or three hours a week. But when the students were around the age of eight, real differences started to emerge. The students who would end up the best in their class began to practice more than everyone else: six hours a week by age nine, eight hours a week by age twelve, sixteen hours a week by age fourteen, and up and up, until by the age of twenty they were practicing that is, purposefully and single-mindedly playing their instruments with the intent to get better well over thirty hours a week. In fact, by the age of twenty, the elite performers had each totaled ten thousand hours of

practice. By contrast, the merely good students had totaled eight thousand hours, and the future music teachers had totaled just over four thousand hours.

Ericsson and his colleagues then compared amateur pianists with professional pianists. The same pattern emerged. The amateurs never practiced more than about three hours a week over the course of their childhood, and by the age of twenty they had totaled two thousand hours of practice. The professionals, on the other hand, steadily increased their practice time every year, until by the age of twenty they, like the violinists, had reached ten thousand hours.
The striking thing about Ericsson's study is that he and his colleagues couldn't find any "naturals," musicians who floated effortlessly to the top while practicing a fraction of the time their peers did. Nor could they find any "grinds," people who worked harder than everyone else, yet just didn't have what it takes to break the top ranks. Their research suggests that once a musician has enough ability to get into a top music school, the thing that distinguishes one performer from another is how hard he or she works. That's it. And what's more, the people at the very top don't work just harder or even much harder than everyone else. They work much, much harder.

The idea that excellence at performing a complex task requires a critical minimum level of practice surfaces again and again in studies of expertise. In fact, researchers have settled on what they believe is the magic number for true expertise: ten thousand hours.

"The emerging picture from such studies is that ten thousand hours of practice is required to achieve the level of mastery associated with being a world-class expert in anything," writes the neurologist Daniel Levitin. "In study after study, of composers, basketball players, fiction writers, ice skaters, concert pianists, chess players, master criminals, and what have you, this number comes up again and again. Of course, this doesn't address why some people get more out of their practice sessions than others do. But no one has yet found a case in which true world class expertise was accomplished in less time. It seems that it takes the brain this long to assimilate all that it needs to know to achieve true mastery."

This is true even of people we think of as prodigies. Mozart, for example, famously started writing music at six. But, writes the psychologist Michael Howe in his book *Genius Explained*,

> by the standards of mature composers, Mozart's early works are not outstanding. The earliest pieces were all probably written down by his father, and perhaps improved in the process. Many of Wolfgang's childhood compositions, such as the first seven of his concertos for piano and orchestra, are largely arrangements of works by other composers. Of those concertos that only contain music original to Mozart, the earliest that is now regarded as a masterwork (No. 9, K. 271) was not composed until he was twenty-one: by that time Mozart had already been composing concertos for ten years.

The music critic Harold Schonberg goes further: Mozart, he argues, actually "developed late," since he didn't produce his greatest work until he had been composing for more than twenty years. To become a chess grandmaster also seems to take about ten years. (Only the legendary Bobby Fischer got to that elite level in less than that amount of time: it took him nine years.) And what's ten years? Well, it's roughly how long it takes to put in ten thousand hours of hard practice. Ten thousand hours is the magic number of greatness.

Here is the explanation for what was so puzzling about the rosters of the Czech and Canadian national sports teams. There was practically no one on those teams born after September 1, which doesn't seem to make any sense. You'd think that there should be a fair number of Czech

hockey or soccer prodigies born late in the year who are so talented that they eventually make their way into the top tier as young adults, despite their birth dates.

But to Ericsson and those who argue against the primacy of talent, that isn't surprising at all. That late-born prodigy doesn't get chosen for the all-star team as an eight-year-old because he's too small. So he doesn't get the extra practice. And without that extra practice, he has no chance at hitting ten thousand hours by the time the professional hockey teams start looking for players. And without ten thousand hours under his belt, there is no way he can ever master the skills necessary to play at the top level. Even Mozart the greatest musical prodigy of all time couldn't hit his stride until he had his ten thousand hours in. Practice isn't the thing you do once you're good. It's the thing you do that makes you good.

The other interesting thing about that ten thousand hours, of course, is that ten thousand hours is an enormous amount of time. It's all but impossible to reach that number all by yourself by the time you're a young adult. You have to have parents who encourage and support you. You can't be poor, because if you have to hold down a part-time job on the side to help make ends meet, there won't be time left in the day to practice enough. In fact, most people can reach that number only if they get into some kind of special program like a hockey all-star squad or if they get some kind of extraordinary opportunity that gives them a chance to put in those hours.

3.
So, back to Bill Joy. It's 1971. He's tall and gawky and sixteen years old. He's the math whiz, the kind of student that schools like MIT and Caltech and the University of Waterloo attract by the hundreds. “When Bill was a little kid, he wanted to know everything about everything way before he should've even known he wanted to know,” his father, William, says. “We answered him when we could. And when we couldn't, we would just give him a book.” When it came time to apply to college, Joy got a perfect score on the math portion of the Scholastic Aptitude Test. “It wasn't particularly hard,” he says matter-of-factly. “There was plenty of time to check it twice.”

He has talent by the truckload. But that's not the only consideration. It never is. The key to his development is that he stumbled across that nondescript building on Beal A venue.
In the early 1970s, when Joy was learning about programming, computers were the size of rooms. A single machine (which might have less power and memory than your microwave now has) could cost upwards of a million dollars and that's in 1970s dollars. Computers were rare. If you found one, if was hard to get access to it; if you managed to get access, renting time on it cost a fortune.

What's more, programming itself was extraordinarily tedious. This was the era when computer programs were created using cardboard punch cards. Each line of code was imprinted on the card using a keypunch machine. A complex program might include hundreds, if not thousands, of these cards in tall stacks. Once a program was ready, you walked over to whatever mainframe computer you had access to and gave the stack of cards to an operator. Since computers could handle only one task at a time, the operator made an appointment for your program, and depending on how many people were ahead of you in line, you might not get your cards back for a few hours or even a day. And if you made even a single error even a typographical error in your program, you had to take the cards back, track down the error, and begin the whole process again.

Under those circumstances, it was exceedingly difficult for anyone to become a programming expert. Certainly becoming an expert by your early twenties was all but impossible. When you

can “program” for only a few minutes out of every hour you spend in the computer room, how can you ever get in ten thousand hours of practice? “Programming with cards,” one computer scientist from that era remembers, “did not teach you programming. It taught you patience and proofreading.”

It wasn't until the mid-1960s that a solution to the programming problem emerged. Computers were finally powerful enough that they could handle more than one “appointment” at once. If the computer's operating system was rewritten, computer scientists realized, the machine's time could be shared; the computer could be trained to handle hundreds of tasks at the same time. That, in turn, meant that programmers didn't have to physically hand their stacks of computer cards to the operator anymore. Dozens of terminals could be built, all linked to the mainframe by a telephone line, and everyone could be working online all at once.

Here is how one history of the period describes the advent of time-sharing:

> This was not just a revolution. It was a revelation. Forget the operator, the card decks, the wait. With time-sharing, you could sit at your Teletype, bang in a couple of com mands, and get an answer then and there. Time-sharing was interactive: A program could ask for a response, wait for you to type it in, act on it while you waited, and show you the result, all in “real time.”

This is where Michigan came in, because Michigan was one of the first universities in the world to switch over to time-sharing. By 1967, a prototype of the system was up and running. By the early 1970s, Michigan had enough computing power that a hundred people could be programming simultaneously in the Computer Center. “In the late sixties, early seventies, I don't think there was anyplace else that was exactly like Michigan,” Mike Alexander, one of the pioneers of Michigan's computing system, said. “Maybe MIT. Maybe Carnegie Mellon. Maybe Dartmouth. I don't think there were any others.”

This was the opportunity that greeted Bill Joy when he arrived on the Ann Arbor campus in the fall of 1971. He hadn't chosen Michigan because of its computers. He had never done anything with computers in high school. He was interested in math and engineering. But when the programming bug hit him in his freshman year, he found himself by the happiest of accidents in one of the few places in the world where a seventeen-year-old could program all he wanted. “Do you know what the difference is between the computing cards and time-sharing?” Joy says. “It's the difference between playing chess by mail and speed chess.” Programming wasn't an exercise in frustration anymore. It was fun.”

“I lived in the north campus, and the Computer Center was in the north campus,” Joy went on. "How much time did I spend there? Oh, a phenomenal amount of time. It was open twenty-four hours. I would stay there all night, and just walk home in the morning. In an average week in those years, I was spending more time in the Computer Center than on my classes. A ll of us down there had this recurring nightmare of forgetting to show up for class at all, of not even realizing we were enrolled.

“The challenge was that they gave all the students an account with a fixed amount of money, so your time would run out. When you signed on, you would put in how long you wanted to spend on the computer. They gave you, like, an hour of time. That's all you'd get. But someone figured out that if you put in 'time equals' and then a letter, like t equals k, they wouldn't charge you,” he said, laughing at the memory. “It was a bug in the soft ware. You could put in t equals k and sit there forever.”

Just look at the stream of opportunities that came Bill Joy's way. Because he happened to go to a farsighted school like the University of Michigan, he was able to practice on a time-sharing system instead of with punch cards; because the Michigan system happened to have a bug in it, he could program all he wanted; because the university was willing to spend the money to keep the Computer Center open twenty-four hours, he could stay up all night; and because he was able to put in so many hours, by the time he happened to be presented with the opportunity to rewrite UNIX, he was up to the task. Bill Joy was brilliant. He wanted to learn. That was a big part of it. But before he could become an expert, someone had to give him the opportunity to learn how to be an expert.

“At Michigan, I was probably programming eight or ten hours a day,” he went on. “By the time I was at Berkeley I was doing it day and night. I had a terminal at home. I'd stay up until two or three o'clock in the morning, watching old movies and programming. Sometimes I'd fall asleep at the keyboard” he mimed his head falling on the keyboard “and you know how the key repeats until the end, and it starts to go beep, beep, beep. After that happens three times, you have to go to bed. I was still relatively incompetent even when I got to Berkeley. I was proficient by my second year there. That's when I wrote programs that are still in use today, thirty years later.” He paused for a moment to do the math in his head which for someone like Bill Joy doesn't take very long. Michigan in 1971. Programming in earnest by sophomore year. Add in the summers, then the days and nights in his first year at Berkeley. “So, so maybe... ten thousand hours?” He said, finally. “That's about right.”

4.

Is the ten-thousand-hour rule a general rule of success? If we scratch below the surface of every great achiever, do we always find the equivalent of the Michigan Computer Center or the hockey all-star team some sort of special opportunity for practice? Let's test the idea with two examples, and for the sake of simplicity, let's make them as familiar as possible: the Beatles, one of the most famous rock bands ever; and Bill Gates, one of the world's richest men.

The Beatles, John Lennon, Paul McCartney, George Harrison, and Ringo Starr, came to the United States in February of 1964, starting the so-called British Invasion of the American music scene and putting out a string of hit records that transformed the face of popular music.

The first interesting thing about the Beatles for our purposes is how long they had already been together by the time they reached the United States. Lennon and McCartney first started playing together in 1957, seven years prior to landing in America. (Incidentally, the time that elapsed between their founding and their arguably greatest artistic achievements Sgt. Pepper's Lonely Hearts Club Band and The Beatles [White Album] is ten years.) And if you look even more closely at those long years of preparation, you'll find an experience that, in the context of hockey players and Bill Joy and world-class violinists, sounds awfully familiar. In 1960, while they were still just a struggling high school rock band, they were invited to play in Hamburg, Germany.

“Hamburg in those days did not have rock-and-roll music clubs. It had strip clubs,” says Philip Norman, who wrote the Beatles biography *Shout!* "There was one particular club owner called Bruno, who was originally a fairground showman. He had the idea of bringing in rock groups to play in various clubs. They had this formula. It was a huge nonstop show, hour after hour, with a lot of people lurching in and the other lot lurching out. And the bands would play all the time to catch the passing traffic. In an American red-light district, they would call it nonstop striptease. “Many of the bands that played in Hamburg were from Liverpool,” Norman went on. “It was an accident. Bruno went to London to look for bands. But he happened to meet an entrepreneur from Liverpool in Soho who was down in London by pure chance. And he arranged to send

some bands over. That's how the connection was established. And eventually the Beatles made a connection not just with Bruno but with other club owners as well. They kept going back because they got a lot of alcohol and a lot of sex."

And what was so special about Hamburg? It wasn't that it paid well. It didn't. Or that the acoustics were fantastic? They weren't. Or that the audiences were savvy and appreciative? They were anything but. It was the sheer amount of time the band was forced to play.

Here is John Lennon, in an interview after the Beatles disbanded, talking about the band's performances at a Hamburg strip club called the Indra:

> We got better and got more confidence. We couldn't help it with all the experience playing all night long. It was handy them being foreign. We had to try even harder, put our heart and soul into it, to get ourselves over. In Liverpool, we'd only ever done one-hour sessions, and we just used to do our best numbers, the same ones, at everyone. In Hamburg, we had to play for eight hours, so we really had to find a new way of playing.

Eight hours?

Here is Pete Best, the Beatles' drummer at the time: "Once the news got out about that we were making a show, the club started packing them in. We played seven nights a week. At first we played almost nonstop till twelve-thirty, when it closed, but as we got better the crowds stayed till two most mornings."

Seven days a week?

The Beatles ended up traveling to Hamburg five times between i960 and the end of 1962. On the first trip, they played 106 nights, five or more hours a night. On their second trip, they played 92 times. On their third trip, they played 48 times, for a total of 172 hours on stage. The last two Hamburg gigs, in November and December of 1962, involved another 90 hours of performing. All told, they performed for 270 nights in just over a year and a half. By the time they had their first burst of success in 1964, in fact, they had performed live an estimated twelve hundred times. Do you know how extraordinary that is? Most bands today don't perform twelve hundred times in their entire careers. The Hamburg crucible is one of the things that set the Beatles apart.

"They were no good onstage when they went there and they were very good when they came back," Norman went on. "They learned not only stamina. They had to learn an enormous amount of numbers cover versions of everything you can think of, not just rock and roll, a bit of jazz too. They weren't disciplined onstage at all before that. But when they came back, they sounded like no one else. It was the making of them."

5

Let's now turn to the history of Bill Gates. His story is almost as well known as the Beatles'. Brilliant, young math whiz discovers computer programming. Drops out of Harvard. Starts a little computer company called Microsoft with his friends. Through sheer brilliance and ambition and guts Gates builds it into the giant of the software world. That's the broad outline. Let's dig a little bit deeper.

Gates's father was a wealthy lawyer in Seattle, and his mother was the daughter of a well-to-do banker. As a child Bill was precocious and easily bored by his studies. So his parents took him out of public school and, at the beginning of seventh grade, sent him to Lakeside, a private

school that catered to Seattle's elite families. Midway through Gates's second year at Lakeside, the school started a computer club.

“The Mothers' Club at school did a rummage sale every year, and there was always the question of what the money would go to,” Gates remembers. “Some went to the summer program, where inner-city kids would come up to the campus. Some of it would go for teachers. That year, they put three thousand dollars into a computer terminal down in this funny little room that we subsequently took control of. It was kind of an amazing thing.”

It was an “amazing thing,” of course, because this was 1968. Most colleges didn't have computer clubs in the 1960s. Even more remarkable was the kind of computer Lakeside bought. The school didn't have its students learn programming by the laborious computer-card system, like virtually everyone else was doing in the 1960s. Instead, Lakeside installed what was called an ASR-33 Teletype, which was a time-sharing terminal with a direct link to a mainframe computer in downtown Seattle. “The whole idea of time-sharing only got invented in nineteen sixty five,” Gates continued. “Someone was pretty forward-looking.” Bill Joy got an extraordinary, early opportunity to learn programming on a time-share system as a fresh man in college, in 1971. Bill Gates got to do real-time programming as an eighth grader in 1968.

From that moment forward, Gates lived in the computer room. He and a number of others began to teach themselves how to use this strange new device. Buying time on the mainframe computer the ASR was hooked up to was, of course, expensive even for a wealthy institution like Lakeside and it wasn't long before the $3,000 put up by the Mothers' Club ran out. The parents raised more money. The students spent it. Then a group of programmers at the University of Washington formed an outfit called Computer Center Corporation (or C-Cubed), which leased computer time to local companies. As luck would have it, one of the founders of the firm Monique Rona had a son at Lakeside, a year ahead of Gates. Would the Lakeside computer club, Rona wondered, like to test out the company's software programs on the weekends in exchange for free programming time? Absolutely! After school, Gates took the bus to the C-Cubed offices and programmed long into the evening.

C-Cubed eventually went bankrupt, so Gates and his friends began hanging around the computer center at the University of Washington. Before long, they latched onto an outfit called ISI (Information Sciences Inc.), which agreed to let them have free computer time in exchange for working on a piece of software that could be used to automate company payrolls. In one seven-month period in 1971, Gates and his cohorts ran up 1,575 hours of computer time on the ISI mainframe, which averages out to eight hours a day, seven days a week.

“It was my obsession,” Gates says of his early high school years. “I skipped athletics. I went up there at night. We were programming on weekends. It would be a rare week that we wouldn't get twenty or thirty hours in. There was a period where Paul Allen and I got in trouble for stealing a bunch of passwords and crashing the system. We got kicked out. I didn't get to use the computer the whole summer. This is when I was fifteen and sixteen. Then I found out Paul had found a computer that was free at the University of Washington. They had these machines in the medical center and the physics department. They were on a twenty-four-hour schedule, but with this big slack period, so that between three and six in the morning they never scheduled anything.” Gates laughed. “I'd leave at night, after my bedtime. I could walk up to the University of Washington from my house. Or take the bus. That's why I'm always so generous to the University of Washington, because they let me steal so much computer time.” (Years later, Gates's mother said, “We always wondered why it was so hard for him to get up in the morning.”)

One of the founders of ISI, Bud Pembroke, then got a call from the technology company TRW, which had just signed a contract to set up a computer system at the huge Bonneville Power station in southern Washington State. TRW desperately needed programmers familiar with the particular software the power station used. In these early days of the computer revolution, programmers with that kind of specialized experience were hard to find. But Pembroke knew exactly whom to call: those high school kids from Lakeside who had been running up thousands of hours of computer time on the ISI mainframe. Gates was now in his senior year, and somehow he managed to convince his teachers to let him decamp for Bonneville under the guise of an independent study project. There he spent the spring writing code, supervised by a man named John Norton, who Gates says taught him as much about programming as almost anyone he'd ever met.

Those five years, from eighth grade through the end of high school, were Bill Gates's Hamburg, and by any measure, he was presented with an even more extraordinary series of opportunities than Bill Joy.

Opportunity number one was that Gates got sent to Lakeside. How many high schools in the world had access to a time-sharing terminal in 1968? Opportunity number two was that the mothers of Lakeside had enough money to pay for the school's computer fees. Number three was that, when that money ran out, one of the parents happened to work at C-Cubed, which happened to need someone to check its code on the weekends, and which also happened not to care if weekends turned into weeknights. Number four was that Gates just happened to find out about ISI, and ISI just happened to need someone to work on its payroll software. Number five was that Gates happened to live within walking distance of the University of Washington. Number six was that the university happened to have free computer time between three and six in the morning. Number seven was that TRW happened to call Bud Pembroke. Number eight was that the best programmers Pembroke knew for that particular problem happened to be two high school kids. And number nine was that Lakeside was willing to let those kids spend their spring term miles away, writing code.

And what did virtually all of those opportunities have in common? They gave Bill Gates extra time to practice. By the time Gates dropped out of Harvard after his sophomore year to try his hand at his own software company, he'd been programming practically nonstop for seven consecutive years. He was way past ten thousand hours. How many teenagers in the world had the kind of experience Gates had? “If there were fifty in the world, I'd be stunned,” he says. “There was C-Cubed and the payroll stuff we did, then TRW all those things came together. I had a better exposure to software development at a young age than I think anyone did in that period of time, and all because of an incredibly lucky series of events.”