# 2020 U.S. Programming Skills Report

Insights into the programming ability of candidates in the U.S. and skills gaps directly impacting recruitment and on-the-job performance

SHL




.SHL.

SHL, the global leader in talent innovation, helps companies transform productivity through deeper people insight. Powering the future of business, our data and tools are proven to drive stronger, more sustainable business outcomes.

Our unrivalled assessment science, benchmark data and analytics empower leaders and their teams to make confident, data-driven people decisions, when it matters most – transforming the way organizations recruit, manage and develop talent. With 40-years of talent expertise, we're a trusted technology partner to more than 10,000 companies worldwide. We work with companies of all sizes in every industry across more than 150 countries, including 50% of the Fortune Global 500 and 80% of the FTSE 100.
For more information, visit shl.com.

# Executive Summary

The world has gone digital, and technology is an integral part of all businesses. This has led to a surge in demand for tech-resources across every industry. The Bureau of Labor Statistics expects a growth of 25.6% in jobs for software developers in the coming years.[1]

This highly competitive tech hiring market makes it difficult for recruiters to attract and hire skilled programmers. It takes 50% longer to fill a tech position compared to other roles[2], and recruiters are only able to fill 60% of tech openings in the U.S.[3] The difficulty to meet the high demand for tech resources is compounded by the challenge of finding qualified tech candidates. Recent research studies have shown that U.S. businesses could lose $162 billion in revenue unless they find the candidates they need.[4]

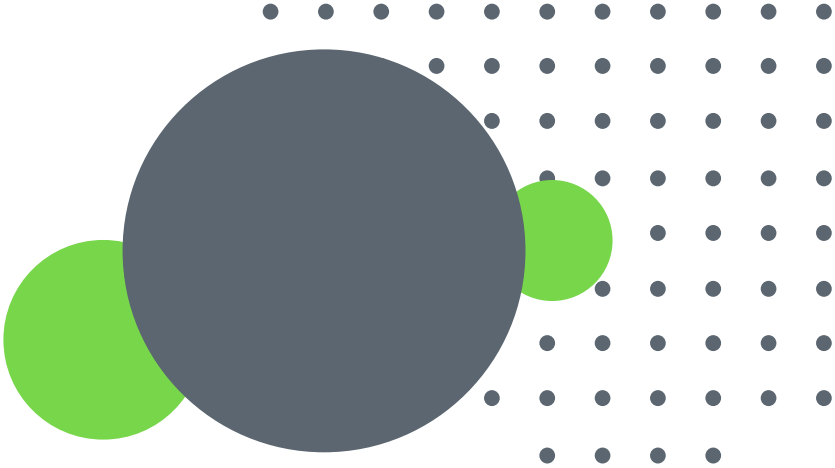
[1] www.bls.gov/emp/tables/occupations-most-job-growth.htm
[2] www.icims.com/hiring-insights/for-employers/how-to-win-tech-talent
[3] www.icims.com/hiring-insights/for-employers/how-to-win-tech-talent
[4] www.kornferry.com/insights/articles/talent-crunch-future-of-work

## Key findings of the report:

1. **About 40% of the candidates can write completely correct code, and another 25% candidates are able to write logically correct code, with some syntactical errors. However, this 25% pool of deserving candidates typically gets filtered out of the hiring process, a huge systemic leakage!**
   A tech hiring coding assessment typically requires the candidate to write the source code to solve the given problem, and it scores the submitted code based on the number of test cases passed. An issue with this kind of evaluation is that it does not factor in superficial or syntactical mistakes in the code. A minor error would be rectified so would not make a difference in the 'real-world', however the candidate who makes a silly mistake in the test would default some test cases and get penalized in the score. Thus, the organization loses out on a possibly good, valuable person in their candidate pool. Our study found that on average, 40% of candidates can write completely correct code. And another 25% have correct logic in their program (they contain the correct control structures and have incorporated the critical data dependencies), so they are deserving candidates, but they fail to clear the test. Hence tech talent pools can increase by up to 65% if the assessment identifies candidates who write code with correct logic rather than complete accuracy. This is a real opportunity to increase the talent pool.

2. **About 20% of candidates follow good programming practices**
   Only 20% of candidates follow good programming practices to write code. The rest of the candidates write code that is low on readability and program structure. Even more importantly, just 6% of the candidates that write correct code follow good programming practices (that make the program readable and maintainable). This is a key skills gap, and the candidates must be taught to write more maintainable and efficient code.

3. **About 25% of candidates can solve complex programming problems (typically required by IT product companies) by writing completely correct code and another 28% are able to write logically correct code with some minor errors**
   Many organizations want to evaluate a candidate's conceptual clarity (for e.g. his/ her ability to solve a complex programming problem like binary search tree, dynamic programming), as they need coders who can constantly innovate and develop new algorithms. Our study found that only 25% can write completely correct code for such problems. However, we can add another 28% if we look at the logic of the code. These are deserving candidates, but a traditional automated assessment tool would not identify them.



4. **Around 7% of candidates are able to write the correct logic but are not able to get their code to compile in the limited time. Hence, they score NIL in the test. These might be good candidates, but the companies miss them.**
   Around 7% of candidates write logically correct code, but they are unable to get their codes to compile within the limited assessment time. These candidates secure a nil score in assessments that score based on number of test cases passed, irrespective of the logical correctness of their algorithm. So companies miss out on this quality talent, primarily due to the limited time allotted for coding tests. There is technology available now to help your organization increase the efficiency of your tech hiring process, providing your candidates with the tools they need to show their skills, and increase the talent pool significantly.
5. **Code readability and maintainability**
   On average, software engineers spend 42% of their time dealing with technical debt and maintenance issues[5] that result from a code with low readability and maintainability. Candidates do not follow best practices implemented across industries while writing code, which result in hundreds of precious productive coding hours going waste, while maintaining and enhancing software products. Some of the common stylistic errors were using a bad variable name, writing long lines of code, not using defined variables and using hardcoded values.
6. **JAVA is the most preferred language amongst candidates**
   Java was the most preferred language amongst candidates followed by Python and C++. This is in line with the market demand as the majority of job openings are in Java and Python programming languages.

[5] https://www.pullrequest.com/blog/cost-of-bad-code/

# Report Methodology

A random sample of 165,000 candidates was used for the study. These candidates were either fresh graduates or young professionals with less than six years of professional experience. The candidates undertook SHL's AI-powered Coding Simulation assessment, which is conducted in a virtual environment. The assessment sends coding problems to the candidate and they get around 40 minutes to write code. The assessment helps to comprehensively evaluate candidate's coding skills. Refer to the Appendix to learn more about SHL's Coding Simulation.

The scores provided by the Coding Simulation assessment were then analyzed by SHL's Research & Development team, Subject Matter Experts and IT industry experts to draw key insights on candidate's programming abilities and skill gaps that can hamper their performance on the job.

# Report Findings

## Programming Skills of candidates

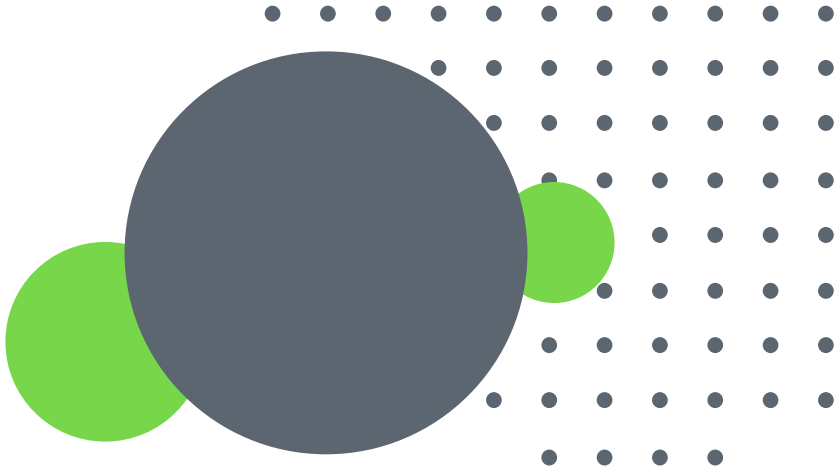
As per Bureau of Labor Statistics, there will be 1.4 million more software development jobs and nowhere near enough applicants who can fill them[6]. On average, it takes 66 days to hire a new tech employee compared to the average of 43 days for all other types of hires.[7] This makes tech hiring quite challenging for recruiters.

80% of revenue growth in 2022 is expected to come from digital products and solutions.[8] COVID-19 has also escalated digital initiatives and led to demand for tech resources. Therefore, organizations need to quickly build a high-quality tech workforce if they want to stay relevant in the new world of work. The demand for tech talent is no longer limited to tech companies but applies to all organizations in various industries.

On an average, 40% of candidates are able to write completely correct code (i.e. one that is logically correct, compiles and passes all test cases). Candidates were found to be more comfortable in solving simple coding problems – around 50% of them could write correct code for Sort/Search, math-based and HashMap data structure problems - but, we see a sharp decline to just 25% when it comes to solving complex programming problems like tree search, dynamic programming. Most companies require proficiency in solving complex programming problems and thus, they find it hard to identify highly skilled programmers.

Organizations who want more qualified tech candidates must revisit their hiring approach, adopt digital hiring, and move away from traditional hiring, that will help to quickly identify best-fit tech talent and build a world-class tech team. In addition, they need to look beyond their traditional talent sources to fill the huge demand for tech resources. Consider candidates that are part of bootcamps, MOOCs and alternative degree courses, as they have also been found to possess students with sound programming skills. Gartner's Recruiting for Tech Talent 2019 report explained that the key to increasing your talent pool is to expand your hiring criteria which better locates and attracts top technical talent.[9]

[6] https://www.usatoday.com/story/tech/talkingtech/2017/03/22/tech-where-jobs/99496462/
[7] https://www.zdnet.com/article/us-companies-facing-a-huge-tech-talent-deficit-in-2020/
[8] https://www.cio.com/article/3237066/5-things-that-are-dragging-down-your-digital-transformation.html
[9] https://www.gartner.com/en/documents/3909099/recruiting-for-tech-talent



## Shortcomings of Traditional Coding Assessments

Traditional coding assessments evaluate candidates' coding ability in rudimentary ways – giving a score only when a candidate codes correctly. They penalize candidates who make minor mistakes in their code which makes it uncompilable or fails to pass all test cases. Limited assessment time often contributes to an increase in minor errors for most candidates. If candidates are given more time, it is quite likely that they will not make these mistakes.

Coding assessments should focus on evaluating coding ability which rests heavily on the logical correctness of the code (i.e. whether the candidate was able to apply the right logic to solve the problem at hand), but traditional assessments are often not equipped to score candidates' programming ability. They work as a Boolean system: giving a straight 0 to candidates with logically correct but uncompilable code.

This seriously reduces the size of shortlisted candidates and limits the chances of identifying good tech talent, and, given that only 60% tech positions are filled, shortages introduced by the evaluation process are surely not healthy for any organization.

Based on data of over 165,000 candidates' code it was found that:

- Around 38% of candidates have written correct code with basic and advanced functionality and partially correct handling of corner test cases. The source code passed all the basic and advanced test cases in the test suite and a percentage of the edge test case(s) as well.
- Nearly 24% of the candidates' code were logically sound code, but its test case failed because of minor errors.

The data clearly shows that these candidates possess strong coding skills but a traditional coding assessment would have rejected them due to the restricted capabilities of the assessment. These candidates surely deserve a chance to be considered.

## Skills Gaps

Due to COVID-19, business models have changed rapidly from traditional, in-person testing to digital-first approaches. Social distancing has forced people to stay indoors and work remotely.

This has led to a surge in demand for video software, online movie streaming, and remote working tools; there has also been a huge demand for resources in tech areas like data science, analytics, cloud computing, mobile developing and software engineering. In 2020 alone, there has been an increase of 64,000 job openings for data science and analytics alone.[10]

The U.S. is not able to serve this huge demand for tech talent. As per a recent research done by Korn Ferry, "The United States, currently the world's leading technology market, can expect to lose out on $162 billion by 2030 due to sector skills shortages. These talent deficits may be perilous to America's status as the global tech center."

Our data and analysis also share the same situation, only 20% of candidates were able to write a readable and maintainable code, and only 6% of the codes written were functionally correct, maintainable and of optimized complexity. This is a key metric that directly impacts speed and performance of any software solution. In the ever-increasing digital age, there is a clear gap in the ability to write optimized code.

[10] https://analyticsindiamag.com/reasons-why-there-is-a-shortage-of-data-scientists-in-the-industry/

## Most Common Code Readability Errors

Code readability refers to a code that is easily comprehensible by other programmers and directly impacts the maintainability of the code (i.e. the ease of maintaining and enhancing the code). Code with poor readability makes it hard to understand, debug, maintain and extend. All software needs to be constantly updated and maintained to support new features and improve performance. It is critical for coders to follow language best practices to write quality code.

Some of the most common errors made by candidates were:

1. Using a name that is too long or short for a variable -- not following recommended character limits. 28% of the candidates who undertook the coding assessment made this error. This was the most common error made by candidates across all languages. 67% of candidates coding in JavaScript made the error, followed by 34% of candidates using Java, 30% using C, and 30% using C++.
2. Writing long lines of code – when length of the code exceeds the recommended number of characters. Around 20% of candidates made this error. The maximum number of candidates coding in JavaScript made this error (37%), followed by Python (36%).
3. Bad code formatting – Candidates that were coding in Java were more likely to write badly formatted code, followed by Ruby, JavaScript and Python

**Table 1: Most common code readability errors made by candidates**

| Readability errors | Java | C | C# | C++ | JavaScript | Python | Ruby | Overall |
|---|---|---|---|---|---|---|---|---|
| Bad Variable Name | 34% | 30% | 8% | 30% | 67% | 10% | 7% | 28% |
| Line over maximum character limit | 14% | 4% | 22% | 12% | 37% | 36% | 13% | 20% |
| Better code formatting can be done | 11% | 2% | 6% | 3% | 5% | 4% | 7% | 8% |

Organizations assume that candidates already possess the skills to write a good, maintainable code, but research has found that teaching these skills is not a top priority for colleges. Thus, organizations need to invest in training the selected candidates to maintain a high-quality code.

[10] https://analyticsindiamag.com/reasons-why-there-is-a-shortage-of-data-scientists-in-the-industry/

[11] https://www.researchgate.net/publication/299412540_Code_Readability_Testing_an_Empirical_Study

## Language Preference

Java has shown to be the most preferred programming language amongst candidates followed by Python and C++. This corresponds to the job market as Java is one of the most sought-after coding skills, as most enterprise solutions, mobile apps, etc. are written in Java language. It frequently figures in the list of Top Programming Languages and many reports claim it to be the most popular primary programming language available. Popular mobile apps like Twitter, Spotify, Pinterest, Uber and more have built their Android applications using Java.

Python is a close second. It was one of the most in-demand programming language in the U.S. with the highest number of jobs posting (74,000) in January 2020. Python is extensively used in artificial intelligence (AI), machine learning, big data, and robotics that has led to its immense popularity amongst the developer community and employers.

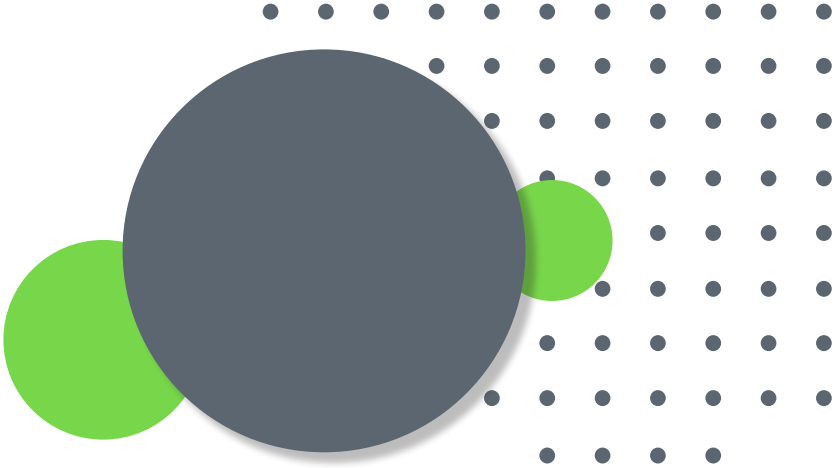
Leading software giants like Google use Python for almost all solutions including their AI algorithms and App Engine Cloud. The same can be said for Dropbox – 99.9% of its code has been written in Python.

Over the past 5 years, we have seen a number of new programming languages appear in the coding arena like React, Typescript, Kubernetes, Docker and more. Recruiters and hiring managers should not only evaluate knowledge of the desired programming language required for the job role, but also the candidate's ability to cross-skill from one programming language to another. You can rely on a technical-skills-assessment to evaluate candidate's knowledge of other programming languages and use the technical interview as an opportunity to also evaluate candidate's aptitude and approach towards learning a new language.

**Table 2: Programming languages used by candidates for coding assessments**

| Language | Number of codes | Percentage |
|---|---|---|
| Java | 88889 | 53.75% |
| Python | 23221 | 14.04% |
| C++ | 20911 | 12.64% |
| C# | 18184 | 11.00% |
| JavaScript | 9537 | 5.77% |
| C | 2658 | 1.61% |
| Ruby | 1331 | 0.80% |
| Swift | 639 | 0.39% |
| Total | 165370 | 100.00% |

[12] https://analyticsindiamag.com/reasons-why-there-is-a-shortage-of-data-scientists-in-the-industry/



### Ease of using languages to solve problems

Candidate's preference of using JAVA or Python to solve coding problems over languages like C, is driven not only by demand in the job market, but also due to the ease of using programming applications like Java or Python to write code. Languages like C do not come with ready-to-use data structures that make it easier to write code. For instance, HashMap is not available in C, while languages like Python come with "batteries include", which provides lots of ready-to-use functionalities to the programmer.

Java provides the best Integrated Development Environment, which makes it enjoyable for programmers to code in Java. C and C++ are also considered as complex programming languages by software engineers. As a result, we observe a high number of candidates opting for Java, Python, and C# to write their codes.

[12] https://analyticsindiamag.com/reasons-why-there-is-a-shortage-of-data-scientists-in-the-industry/

# Conclusion

Tech companies like Apple, Microsoft, Amazon, Facebook, Google have led the demand for software programmers, but as digital transformation has taken center stage for almost all businesses (especially due to COVID-19), we observe a strong demand for tech resources emerging from almost every sector. This will make the already tight labor market for high-quality tech talent even more competitive, but the strong demand is not supported by the supply of good talent. Recruiters struggle to find and hire good programmers. Organizations need to change the way they source, evaluate and nurture tech talent.

A different approach is required to source talent – one that is fair, broad, and builds a diverse talent pool for organizations to choose from. It is important for recruiting strategies to move beyond traditional qualifications and experiences and consider talent from an increasingly diverse and non-standard range of sources like bootcamps, MOOCs, alternative degree courses.

Invest in scientific and objective assessments that can help you evaluate a candidate's coding capabilities and competencies, and that do not rely on resumes or traditional coding assessments that weed out good talent from your talent funnel. The tool should use latest technologies (i.e. AI) and provide rich data insights to help you make the best hiring decisions.

According to the World Economic Forum, 50% of new technical knowledge acquired by students will be outdated by graduation.[16] For that matter, organizations need to partner with the government and universities to prepare young talent for the skills they need for future job roles.

Digital transformation is now strategic for almost all businesses, so it is important to invest in reskilling your workforce through regular training and development programs. By 2022 around 54% of the workforce will need major reskilling to be future ready.[17] Organizations like Salesforce have developed programs like Trailhead to give their employees what they need to take learning in their own hands and develop the skills they will required for the future. Two-thirds of U.S. hiring managers believe that training and developing the internal workforce will help to prepare for future innovations or disruptions but most organizations have been slow to adopt these necessary programs.

Organizations need to invest in training their new, as well as existing workforce to reduce the tech skills gap. Udemy's survey of over 50% of Learning & Development leaders said that training for technical skills is their top 2020 priority.[18] Organizations also need to work at grassroot level and align with institutions and universities to ensure that the graduating computer science students possess job-ready skills to quickly become productive.

With the above in mind, HR and talent acquisition professionals must revisit the way they look at their hiring and talent development strategies, to put the right processes and tools in place to attract and nurture the best tech talent.

SHL's Technology Hiring Solution delivers a highly fair, scientific, and fast tech evaluation process. It uses the power of AI to score codes that are not compilable. By identifying such additional candidates, we are able to improve throughput by at least 60%. This results into tiding over the talent shortage and build a high-quality and diverse tech workforce in half the time. Leading organizations across technology, banking, automobile, education and more rely on our Technology Hiring Solution to identify top-class programmers.

[16] www3.weforum.org/docs/WEF_Future_of_Jobs.pdf

[17] www.weforum.org/reports/the-future-of-jobs-report-2018

[18] https://learntocodewith.me/posts/tech-skills-in-demand/

# Appendix

## SHL Coding Simulation

SHL Coding Simulation is an online coding assessment to evaluate coding abilities of candidates. A coding problem is sent to candidates and they are provided with an intuitive and real-world Integrated Development Environment, 50+ programming languages, and built-in compiler to showcase their programming skills. The predictive test comprehensively analyzes candidates' code to grade their programming skills across three parameters:

- Programming Ability – evaluates the implementation of correct algorithms using the control-structures, data dependencies, and relevance of the solution.
- Programming Practices – evaluates the code for the best practices implemented across the industry while developing code, which enhances its readability and usability.
- Functional correctness – measures the degree of code correctness based on the test suite coverage by the developed code. The evaluation tests are: Basic, Advanced, and Edge.

The Coding Simulation uses AI models to score incomplete and uncompilable code on logical correctness and quality. This helps to reduce false rejects and improve throughput for recruiters by identifying more programmers compared to traditional coding assessments.

The Coding Simulation is available for a wide variety of roles and latest technologies like Data Science, Front-end, SQL, Selenium, DevOps, Diagram & Architecture and more. It can be combined with Technology Skills assessment to evaluate candidate's technical concepts and knowledge for 250+ IT skills including Front-end, Back-end, DevOps, Cloud, Networking, Data Science, QA and more.

Live coding interview enables hiring managers to conduct highly structured and efficient technical interviews remotely in a realistic coding environment. They can remotely interact and evaluate candidates in real-time and efficiently collaborate with evaluators post-interview to hire quality tech talent.

SHL