

An Overview of Current Cloud-Based Online Programming Platforms

N. SreeRam

Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, India, sriramnimmagadda@gmail.com

Abstract.

There are several online cloud-based programming platforms available, and coding is becoming more and more important in all fields. There are several stages involved in programming, including requirements analysis, design solution development, coding, and testing. Any program's primary goal is to satisfy the needs of the user. After development is finished, the application has to be tested against the user specifications. To test the programme, it is therefore necessary to comprehend and create a variety of test cases based on the user requirements specification. An overview of identifying, comprehending, and resolving enforced test cases in programming using contemporary cloud-based compilers like Codechef, Hackerrank, and Earth is provided in this article.

Keywords: user requirements, cloud base compilers, online programming platforms, design of test cases, enforcing test cases in to program.

1. Introduction

These days, internet services are crucial, and new technological advancements have made cloud computing accessible over the internet everywhere. Early computers were used for programming, and installing the necessary computing resources and infrastructure—such as editors and compilers—was necessary. Cloud computing has made all computer resources available as software, platforms, and infrastructure as services. Therefore, programming on any platform without the need for a specially configured system is convenient. Programming is now possible on any system, regardless of setup, thanks to modern cloud-based compilers. The process of turning an original mathematical model design into executable computer programmes is called programming. Analysing, comprehending, creating algorithms,

confirming the accuracy of the requirements for algorithms, allocating resources, and executing algorithms in a programming language are all part of the process of producing software. Most programmers use an edit-compile-test cycle while writing code [2]. This cycle occurs in IDEs. To keep with the large and quickly evolving code base and high levels of reprocess coders practice a cloud-based build system. Writing code for any problem involves understand of problem statement, understanding constraints and framing a solution. In this regard this survey paper discusses about the programming task on various cloud-based compilers by considering some of already existing programming platforms. The main challenge while programming on modern online programming platforms is to meet the given constraints by resolving various test cases. Usually test cases are used to automated evaluation of code. The general criteria involved in programming over online platforms is first problem statement need to read, after reading the problem statement the programmer has given sample input and the resulting expected output.

The execution time of any solution depends on the size of the input is known as order of growth [6]. Order of growth is very useful to compute the running time of a solution. There are different notations to represent the order of growth. These notations are known as asymptotic notations.

O-Notation: It denotes upper limit. For a given function $g(n)$, we denote by $O(g(n))$ the set of functions: $O(g(n)) = \{f(n): \text{there exist positive constants } c \text{ and } n_0 \text{ such that } 0 \leq f(n) \leq c * g(n) \text{ for all } n \geq n_0\}$.

Ω -Notation: It denotes lower limit. For a given function $g(n)$, we denote by $\Omega(g(n))$ the set of functions: $\Omega(g(n)) = \{f(n): \text{there exist positive constants } c \text{ and } n_0 \text{ such that } 0 \leq c * g(n) \leq f(n) \text{ for all } n \geq n_0\}$

Θ -Notation: It denotes tight bound. For a given function $g(n)$, we denote by $\Theta(g(n))$ the set of functions: $\Theta(g(n)) = \{f(n): \text{there exist positive constants } c_1, c_2 \text{ and } n_0 \text{ such that } 0 \leq c_1 * g(n) \leq f(n) \leq c_2 * g(n) \text{ for all } n > n_0\}$

This paper lets an overview on programming and understanding the test cases over the modern cloud-based programming platforms. In further sections each section gives outline of programming task on variety of platforms such as hackers rank, hackers earth, code chef etc.

2. Hacker Rank

One of the most well-known cloud-based online communities, Hacker Rank [7], brings together programmers from all over the world to collaborate on solving problems in a variety of computer science domains, including artificial intelligence, machine learning, and algorithms. It also allows programmers to practice different programming paradigms, including structural, object-oriented, and functional programming [4]. This website allows users to learn programming, make money programming, set challenges, and organize competitions. The problems span many different areas, including databases, encryption, security, algorithms, data structures, and programming languages including C, C++, and Java. Every task on this platform has been debugged and examined using a range of test scenarios. A test case is a collection of factors under which a tester will determine the success or failure of a programmer. These test cases come in two flavors: hidden test cases and basic test cases. As a result, test case driven programming is another name for the programming done on this platform [8]. Before beginning to write code, it is necessary to comprehend each challenging problem statement. The issue statement, sample input, and sample output come next. It does not follow that all test cases will be resolved with the supplied inputs and outputs. The code must be tested against each test case after it has been written. The problem description itself may or may not reveal in all test situations. There may be some hidden test cases which require some out of box thinking about the problem. When run the code against hidden test cases, depending on the output generated by code it will result the following judgements

Accepted: When the code passed all the test cases. Executed successfully

Wrong Answer: When the output produced by code did not match the expected output. Then there is need to rethink about the solution because the solution approach may not fit to challenge

Terminated due to time out: When code doesn't solve problem efficiently in stipulated period. For example, when the written solution is $O(2n)$ when $n=100$, it will surely timeout so that there is need to optimize the algorithm

Compile Time Error: When there is syntactical error in code, it will result compile time error

Runtime error: If the code terminated unexpectedly, attempt to divide by zero. Then code must re-examine and avoid conditional statements that causes run time errors.

Segmentation Fault: When code is running overrun the memory or not enough memory is allocated. Then

there is need to examine the variable declarations and their sizes that are equal to given input and outputs.

Abort called: When code is using maximum resources, may be large amount of memory used exceeds the memory limit. Then code must modify to reduce the usage of resources

3. Hacker Earth

Hacker Earth [9], which conduct coding challenges for coders that help to identify good talent, will compete with 19 other start-ups throughout the world. It is a competitive programming platform which support over 32 programming languages include c, c++, java, etc. and it was developed by Like hacker rank, hacker earth is also a cloud based online programming platform on which people throughout the world together can solve the problems from different domains such programming, data structures, machine learning and etc. Hacker earth compiler works in Linux environment so whenever code didn't pass test case it will generate the following signals as runtime errors.

SIGSEGV: This is the most common runtime error may be caused by an out of scope array index, buffer overflow and incorrectly initialized pointer.

SIGXFSZ: When output limit exceeds hacker earth compiler re- turns this status signal

SIGFPE: This status signal will generate by hacker earth compiler due to floating point error or attempt to divide by zero.

SIGABRT: This status signal will produce whenever program abnormally terminate due to insufficient resources such as not availability of enough memory.

NZEC: This signal means program return a value different from 0 i.e it will happen due forgot to write return 0.

MLE: This signal generates when the program trying to allocate memory beyond the indicated size.

Wrong Answer: When the code did not give expected output, this compile produces this message. Then there is need to understand input and output of program and modify the code according to that

4. Code Chef

Codechef is one of the most popular platforms to help programmers to solve and practice variety of problems from different domains such as algorithms and computer programming

and participate in contests. Program should read from and write to standard output. After submitting a solution code chef IDE will give below possible results [10]

1. Accepted: If code ran successfully and gave expected output. If there is a score for the problem, this will be displayed in parenthesis next to the checkmark.
2. Time Limit Exceeded: If code was compiled successfully, but it didn't stop before time limit. Try optimizing solution approach.
3. Wrong answer: If code compiled and ran successfully but the output did not match with expected output. Then there is need to change solution approach.
4. Runtime Error: If code compiled and ran but encountered an error. The most common reasons are using too much memory or dividing by zero. Then there is need to change conditional statements in the code.
5. Compilation Error: If there are syntactical mistakes in code, it will return compilation error. Then there is need to check and modify the syntax of the statements

5. Conclusions

The main moto of this article is to give an outline about the programming over online cloud-based compilers by resolving test cases. This article was prepared by working on above discussed platforms. In addition to above mentioned cloud-based programming platforms few more also available due to limitations we could not cover all of them. Due to availability of this kind of cloud-based programming platforms, one need not to buy and install different programming language compilers explicitly but can use them as software as service.

References

- [1]https://www.sas.com/content/dam/SAS/sv_se/doc/Presentation/SAS-Forum-SAS-in-cloud.pdf
- [2] Hyunmin Seo, Caitlin sadowski, Sebastain Elbaum, Robert Bowdidge “Programmers' Build Errors: A Case Study (at Google) ICSE '14, May 31 - June 7, 2014, Hyderabad, India ACM 978-1- 4503-2756-5/14/06
- [3] Judy C.R.Tseng, Chih-Hsiang WU “An expert system approach to improving stability and reliability of web service Expert Systems with Applications “ Volume 33, Issue 2, August 2007, Pages 379- 388

- [4] <https://www.hackerrank.com/faq>
- [5] https://en.wikipedia.org/wiki/Test_case
- [6] <https://www.hackerearth.com/practice/basic-programming/complexity-analysis/time-and-space-complexity/tutorial/>
- [7] <https://www.hackerrank.com/faq>
- [8] <https://technologyconversations.com/2013/12/20/test-driven-development-tdd-example-walkthrough/>
- [9] <https://www.gadgetsnow.com/tech-news/HackerEarth-in-finals-of-global-startup-competition/articleshow/29526978.cms>
- [10] <https://www.codechef.com/problems/school>
- [11] Reema Tarija “Introduction to C programming” Second Edition
university of Delhi, OXFORD