

Pages / ... / Software Programming Competency Test

# 1. Introduction

Created by TUNG DUC NGUYEN tung2.nguyen, last modified on 2021/05/09

The original is <http://collab.lge.com/main/x/plnWJ>

## Development background and purpose

Đây là một **bài kiểm tra** xác nhận một cách **khách quan** và **định lượng năng lực lập trình** của các nhà phát triển SW, và nhằm mục đích **thiết lập văn hóa học hỏi và phát triển bản thân**, cuối cùng là **giúp phát triển năng lực** của các lập trình viên:

- Đánh giá khách quan / định lượng về khả năng lập trình SW
- Tạo môi trường phát triển năng lực / phát triển bản thân và thiết lập văn hóa học tập
- Nâng cấp khả năng phát triển SW

## Development and operation organization

Nhóm Giáo dục SW của Trung tâm Nghiên cứu Kỹ thuật SW trong Bộ phận CTO là một nhóm giáo dục SW chuyên biệt cho tất cả các công ty LG Electronics.

Chúng tôi đang góp phần nâng cao chất lượng giáo dục và nâng cao năng lực SW.

Ngoài ra, chúng tôi cung cấp dịch vụ và giáo dục mới và sáng tạo bằng cách xác định nhu cầu của nhân viên và hợp tác.

Bài kiểm tra Chứng nhận Năng lực Lập trình SW là một **bài kiểm tra** có thể **đo lường một cách khách quan và định lượng** các **năng lực lập trình SW cơ bản** mà các **nhà phát triển phần mềm** của LG Electronics nên có.

## History

- 2011
  - LG Electronics SW thiết lập hệ thống chứng nhận năng lực phát triển
  - Bắt đầu bài kiểm tra với Bài kiểm tra Chứng nhận Năng lực Trung cấp'SW '
  - C và Java 2 môn để lựa chọn để làm bài kiểm tra
  - Viết và kiểm tra thực tế (Kỹ thuật SW là một môn học phổ biến để viết)
- 2012
  - Kiểm tra viết: Giới thiệu môi trường CBT (JCTest)
  - Kiểm tra thực tế: Giới thiệu môi trường phát triển phần mềm dựa trên ảo hóa (Đám mây)
  - Tham gia Viện nghiên cứu SW tại Ấn Độ (LGSI) tại chỗ
- 2013
  - Tham gia ứng tuyển địa phương cho Trung tâm Nghiên cứu MC ở Trung Quốc
- 2014
  - Tham gia kỳ thi LG Innotek
- 2015
  - Mở hệ thống ứng dụng thử nghiệm ([lset.lge.com](http://lset.lge.com)): ứng dụng thử nghiệm, kiểm tra kết quả thử nghiệm, v.v.
- 2016
  - SW Engineering bị loại khỏi bài kiểm tra viết
  - Được áp dụng như một điều kiện tiên quyết để lựa chọn ứng viên cho khóa học chuyên gia SW (SW Architect, SDET: Kỹ sư phát triển phần mềm trong thử nghiệm)
  - Tham gia kỳ thi LIG Nex1
- 2017
  - Đổi tên bài kiểm tra chứng chỉ năng lực lập trình'SW '
  - Hệ thống ứng dụng kỳ thi mới mở ([codepro.lge.com](http://codepro.lge.com))
  - : Viết mã dựa trên web, biên dịch, gửi, chấm điểm tự động
  - Tổ chức hệ thống kiểm tra (thay đổi số lượng câu hỏi kiểm tra, hình thức kiểm tra và loại câu hỏi kiểm tra)
  - Chỉ kỹ năng thực hành (không bao gồm kiểm tra viết)
  - C, JAVA, C ++, thí sinh chọn ngôn ngữ cho mỗi câu hỏi và thực hiện (không chọn môn C, JAVA)

- 2018
  - Liên kết với hệ thống thăng tiến (yêu cầu về trình độ tối thiểu để được thăng chức)
  - : Chỉ dành cho các công việc SW, những người vượt qua Bài kiểm tra Chứng nhận Năng lực Lập trình SW có thể thay thế tất cả 6 Sigma Green Belt / Black Belt.
  - Áp dụng bổ sung các yêu cầu thiết yếu để lựa chọn ứng viên cho khóa học chuyên gia SW (Chuyên gia AI, Chuyên gia bảo mật SW)
  - Áp dụng tuyển dụng thường xuyên nhóm công việc SW của khối kinh doanh HE
- 2019
  - Tuyển dụng văn phòng chính và bộ phận CTO (Python bổ sung được cung cấp cho 3 ngôn ngữ hiện có trong quá trình kiểm tra tuyển dụng)
  - Phản ánh hệ thống thăng tiến LG Innotek (đạt được chứng nhận năng lực để thăng hạng người đầu tiên / có trách nhiệm)
- Năm 2020
  - Triển khai thử nghiệm trực tuyến đầu tiên: Chuyển đổi thử nghiệm ngoại tuyến hiện có thành trực tuyến
  - Tham gia Viện Nghiên cứu Việt Nam
  - Được áp dụng như một quy trình tiêu chuẩn để tuyển dụng toàn công ty

### Test configuration

Configuration	domain	Explanation	Number of questions	points
Practical skills	Understanding the code	<ul style="list-style-type: none"> <li>Understand the code to understand the meaning of the program</li> <li>Check the idiom of the language or the pattern of the code</li> </ul>	One	30
	problem solving (Data structure, algorithm)	<ul style="list-style-type: none"> <li>Derive the optimal algorithm using data structure</li> <li>Finding a solution that satisfies accuracy and efficiency</li> </ul>	2	70 (30+40)
Sum			3 problems	100 points

Cấu hình	miền	Giải trình	Số lượng các câu hỏi	điểm
Kỹ năng thực hành	Hiểu mã	<ul style="list-style-type: none"> <li>Hiểu mã để hiểu ý nghĩa của chương trình</li> <li>Kiểm tra thành ngữ của ngôn ngữ hoặc mẫu mã</li> </ul>	Một	30
	giải quyết vấn đề (Cấu trúc dữ liệu, thuật toán)	<ul style="list-style-type: none"> <li>Tìm ra thuật toán tối ưu bằng cách sử dụng cấu trúc dữ liệu</li> <li>Tìm giải pháp đáp ứng độ chính xác và hiệu quả</li> </ul>	2	70 (30 + 40)
Tổng			3 vấn đề	100 điểm

### Subject Field/Phạm vi câu hỏi

ability	Justice	Component	Detail	Metric	Details
Problem solving	A key	Problem definition	Ability to	Understanding	Accurate

<p>solution to a given problem is devised and implemented as a code, a well-known algorithmic technique is applied, or an optimal problem solution is derived using a data structure. Finding a solution that satisfies accuracy and efficiency.</p> <p>-Accuracy: Outputs the correct result value for a given input - Efficiency: Efficient use for a given resource</p>		comprehensively understand situations and information to find key problems	the problem	understanding of the requirements of the problem Distinguish between simple or easy problems and difficult to solve problems
			Problem definition	A concise summary of the listed requirements and conversion to another problem equivalent to the defined problem.
	Module reuse	Ability to reuse modules most appropriate to the situation Use empirical knowledge necessary for problem solving Understand and structure the characteristics of language	Module selection	Choosing the most rigorous module to solve your problem
			API utilization	Understanding the defined API Utilizing the API
	Understanding/using data structure	Ability to understand and utilize the characteristics of each data structure to solve a unit problem Define a data structure that considers performance and memory	Arrangement	Array insertion/deletion Multidimensional array utilization Lookup table utilization
			List	Understanding the structure of lists Inserting and deleting lists
			stack	Understanding the structure of the stack Inserts and deletes in the stack
			cue	Understanding the structure of queues Inserting and deleting queues
			Heap	Understanding the structure of the heap Insert and delete heaps Implementing priority queues using heap sort heaps
			Hash table	Understanding the structure of the hash table Insertion

					deletion of the hash table Search operation of the hash table
		Understanding/Utilizing Algorithm	Understand various algorithms, select an algorithm that meets the requirements, identify limitations such as complexity, performance, and memory, and apply the optimal algorithm	Sorting and bipartite search	Sorting implementation-selection, insertion, bubble, merge, quick, linear time binary search implementation and utilization
				Division and conquest	Understanding of division and conquest technique Utilization of division and conquest
				Dynamic Programming (DP)	Understanding DP Using DP
				Greedy	Understanding the greedy strategy Using the greedy strategy
				BFS	Understanding of breadth-first search Use of breadth-first search
				DFS	Understanding of depth-first search Use of depth-first search method
				Flood Fill	Understanding the geometrical approach to space Using the spatial approach
		Solution verification	Verify that the requirements can be fully reflected Verify that the correct answer is answered and how effective it is	accuracy	Ensures that the correct answer is output for a given input range. Loop immutability
				efficiency	Time Complexity Analysis Spatial Complexity Analysis
<b>Understanding the code</b>	-Grasp the meaning of the program by grasping the existing code-Grasp the idiom of the	Understanding of programming language	Understanding the programming language itself Understanding the language grammar Understanding the syntactic	- Understanding the grammar of each programming language -Understand creation, use, and release of	-Understanding the primitive type variable system (declaration, assignment, use, release) -Understanding the extended type system (declaration,

language or the pattern of the code in an abstract meaning.		usage of programming languages	data- Understand how to operate on data- Understand logic flow control- Understand how to declare and use functions	assignment, use, release of classes, structures, objects, etc.) -Operators, loop statements, control statements, functions (declaration, call)
	Understanding the idiom/pattern of language	Can understand idiomatic expressions of language. Understand patterns applied to code and understand behavior	Ability to identify patterns frequently used in language	-Refer to design patterns -Evaluation of idioms by language
	Identification of code history	Understanding the history and issues related to the code		
	Understanding module-level abstraction	The purpose of the module can be understood and utilized. The purpose of the overall method function according to the flow of the legacy source code can be identified and the result of the semantic code fragment can be expected by comparing it with the design part .	-Able to predict and derive the purpose of each module - Able to understand the relationship between each module	-Algorithm derivation from the presented code- Deduction of the purpose of processing data from the presented code- Understanding the purpose of each presented code module (function, class, object, etc.) -Understanding the relationship between the presented modules

Khả năng	Sự công bằng	Thành phần	Chi tiết	Hệ mét	Chi tiết
<b>Giải quyết vấn đề</b>	Giải pháp quan trọng cho một vấn đề nhất định được nghĩ ra và thực hiện dưới dạng mã, áp dụng kỹ thuật thuật toán nổi	Định nghĩa vấn đề	Khả năng hiểu toàn diện các tình huống và thông tin để tìm ra các vấn đề chính	Hiểu vấn đề	Hiểu chính xác yêu cầu của bài toán Phân biệt bài toán đơn giản, bài toán dễ và bài toán khó.
				Định nghĩa vấn đề	Một bản tóm tắt ngắn gọn các yêu cầu được liệt kê và chuyển đổi sang một vấn đề khác tương đương với vấn đề đã xác định.
		Tái sử dụng mô-đun	Khả năng sử dụng lại các mô-đun phù hợp nhất với tình huống	Lựa chọn mô-đun	Chọn mô-đun nghiêm ngặt nhất để giải quyết vấn đề của bạn

<p>tiếng hoặc giải pháp vấn đề tối ưu được đưa ra bằng cách sử dụng cấu trúc dữ liệu.</p> <p>Tìm kiếm một giải pháp đáp ứng độ chính xác và hiệu quả.</p> <p>-Độ chính xác: Đưa ra giá trị kết quả chính xác cho một đầu vào nhất định - Hiệu quả: Sử dụng hiệu quả cho một tài nguyên nhất định</p>		Sử dụng kiến thức thực nghiệm cần thiết để giải quyết vấn đề	Sử dụng API	Hiểu API đã xác định Sử dụng API
	Hiểu / sử dụng cấu trúc dữ liệu	Khả năng hiểu và sử dụng các đặc điểm của từng cấu trúc dữ liệu để giải quyết các vấn đề của đơn vị	Sắp xếp	Chèn / xóa mảng Sử dụng mảng đa chiều Sử dụng bảng tra cứu
			Danh sách	Hiểu cấu trúc của danh sách Chèn và xóa danh sách
			cây rơm	Hiểu cấu trúc của ngăn xếp Chèn và xóa trong ngăn xếp
			gợi ý	Hiểu cấu trúc của hàng đợi Chèn và xóa hàng đợi
			Đống	Hiểu cấu trúc của heap Chèn và xóa heap Thực hiện các hàng đợi ưu tiên bằng cách sử dụng heap sắp xếp heap
			Bảng băm	Hiểu cấu trúc của bảng băm Chèn xóa bảng băm Thao tác tìm kiếm bảng băm
	Hiểu / Sử dụng thuật toán	Hiểu các thuật toán khác nhau, chọn một thuật toán đáp ứng các yêu cầu , xác định các giới hạn như độ phức tạp, hiệu suất và bộ nhớ và áp dụng thuật toán tối ưu	Sắp xếp và tìm kiếm hai bên	Sắp xếp lựa chọn triển khai, chèn, bong bóng, hợp nhất, triển khai và sử dụng tìm kiếm nhị phân theo thời gian tuyến tính, nhanh chóng
			Phân chia và chinh phục	Hiểu biết về kỹ thuật phân chia và chinh phục Sử dụng kỹ thuật phân chia và chinh phục
			Lập trình động (DP)	Hiểu DP bằng cách sử dụng DP
			Tham	Hiểu chiến lược tham lam Sử dụng chiến lược tham lam
			BFS	Hiểu biết về tìm kiếm theo chiều rộng Sử dụng tìm kiếm ưu tiên theo chiều rộng
			DFS	Hiểu biết về tìm kiếm theo chiều sâu Sử dụng phương pháp tìm kiếm theo chiều sâu
	Xác minh giải pháp	Xác minh rằng bạn có thể phản ánh đầy đủ các yêu cầu của mình	Lắp đầy lỗ	Hiểu cách tiếp cận hình học trong không gian Sử dụng cách tiếp cận không gian
			sự chính xác	Đảm bảo rằng câu trả lời đúng được xuất cho một phạm vi đầu vào nhất định. Tính bất biến của vòng lặp

			Xác minh rằng bạn đưa ra câu trả lời chính xác và mức độ hiệu quả của câu trả lời đó	hiệu quả	Phân tích độ phức tạp theo thời gian Phân tích độ phức tạp theo không gian
<b>Hiểu mã</b>	-Tìm hiểu ý nghĩa của chương trình bằng cách nắm được mã hiện có - Nắm được thành ngữ của ngôn ngữ hoặc mẫu mã theo nghĩa trừu tượng.	Hiểu biết về ngôn ngữ lập trình	Hiểu chính ngôn ngữ lập trình Hiểu ngữ pháp ngôn ngữ Hiểu cách sử dụng cú pháp của ngôn ngữ lập trình	- Hiểu ngữ pháp của từng ngôn ngữ lập trình -Hiểu việc tạo, sử dụng và giải phóng dữ liệu- Hiểu cách hoạt động trên dữ liệu- Hiểu điều khiển luồng logic- Hiểu cách khai báo và sử dụng các hàm	-Hiểu hệ thống biến kiểu nguyên thủy (khai báo, gán, sử dụng, giải phóng) -Hiểu hệ thống kiểu mở rộng (khai báo, gán, sử dụng, giải phóng các lớp, cấu trúc, đối tượng, v.v.) -Các bộ điều hành, câu lệnh lặp, câu lệnh điều khiển, hàm (khai báo, gọi điện)
		Hiểu thành ngữ / mẫu ngôn ngữ	Có thể hiểu các cách diễn đạt thành ngữ của ngôn ngữ. Hiểu các mẫu được áp dụng cho mã và hiểu hành vi	Khả năng xác định các mẫu thường được sử dụng trong ngôn ngữ	-Yêu cầu thiết kế mẫu -Đánh giá thành ngữ theo ngữ liệu
		Nhận dạng lịch sử mã	Hiểu lịch sử và các vấn đề liên quan đến mã		
		Hiểu trừu tượng cấp mô-đun	Mục đích của mô-đun có thể được hiểu và sử dụng. Mục đích của	-Có thể đoán và suy	Nguồn gốc -Algorithm từ giới mã Trích của mục đích xử lý dữ liệu từ giới số- Hiểu mục đích của mỗi mô-đun mã trình bày (chức năng, lớp, đối tượng, vv)

			chức năng phương pháp tổng thể theo luồng của mã nguồn kế thừa có thể được xác định và kết quả của đoạn mã ngữ nghĩa có thể được dự đoán bằng cách so sánh nó với phần thiết kế .	ra mục đích của từng mô-đun - Có thể hiểu mối quan hệ giữa từng mô-đun	-Understanding mối quan hệ giữa các module giới
--	--	--	---	--	---

### Thời khóa biểu kiểm tra

#### ◆ Test Composition and Acceptance Criteria

- Area : Understanding Code, Problem-Solving
- Language of Test : C , C++, Java, Python
- Test Time : AM 09:00 – 13:00, 4hours
- Acceptance Criteria : 70 points or more

Test	Area	Explanation	The Number of Questions	Point	Acceptance Criteria	Time
Practical Test Only	Understanding Code	Identify code to understand program meaning, Determine the idiom or code pattern of a SW language	1	30	70 points or more	09:00~13:00
	Problem-Solving	Write solution that satisfy optimal algorithms, accuracy and efficiency	2	70 (30+40)		
Total			3	100	70	240min.

### Tiêu chí chấp nhận

- Đạt được hơn 70 điểm (100 điểm)

### Tiêu chí chấm điểm

- **Tính điểm tự động trực tuyến** dựa trên các trường hợp thử nghiệm
- Kết quả (điểm) của bài thi sẽ **được công bố riêng** sau khi bài thi kết thúc.
- Chúng tôi không tiết lộ tất cả các trường hợp thử nghiệm cho mỗi vấn đề. **Chỉ các trường hợp thử nghiệm được xác định** theo vấn đề mới **được tiết lộ** .
- **Điểm số có thể khác nhau đối với** từng trường hợp thử nghiệm .
- **Điểm được tính cho số lượng bài kiểm tra đã vượt qua** (điểm một phần được công nhận).



- **Chạy thời gian** và với **bộ nhớ** kiểm tra và có thể được bao gồm trong các trường hợp thử nghiệm.

## Thực hành

[codepro.lge.com](https://codepro.lge.com) có thể được truy cập cả bên trong và bên ngoài công ty.

Mật khẩu: codepro \$\$ %% (Mã được viết trong trải nghiệm sẽ không được lưu.)

## Trang web được đề xuất

<a href="#">Tập chí phần mềm</a>	Nếu bạn chưa biết, chúng tôi sẽ chia sẻ các mẹo về phát triển lỗi và kiệt sức.
<a href="#">Tem mã hóa</a>	Là nơi sinh viên tập hợp và giải quyết vấn đề hai tuần một lần để phát triển kỹ năng viết mã tốt.
<a href="#">Baekjoon Online Judge</a>	Có rất nhiều vấn đề trang web có thể giải quyết các vấn đề thuật toán, vì vậy bạn có thể giải quyết các vấn đề khác nhau.
<a href="#">Algospot</a>	Trang web giải quyết các vấn đề liên quan đến thuật toán / cấu trúc dữ liệu
<a href="#">Leetcode</a>	Trang web giải quyết vấn đề thuật toán trực tuyến
<a href="#">Dự án Euler</a>	Cung cấp nhiều cấp độ toán học khác nhau
<a href="#">CodeKata</a>	Một bộ sưu tập của chương trình đơn giản và vui vẻ các vấn đề
<a href="#">Visualgo</a>	Một trang web nơi bạn có thể bắt đầu học các thuật toán

No labels