

ĐẠI HỌC BÁCH KHOA HÀ NỘI

TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



BÁO CÁO

HỌC PHẦN: PROJECT I

Sinh viên	: Nguyễn Khánh Sơn
MSSV	: 20215471
Email	: Son.NK215471@sis.hust.edu.vn
Lớp	: 733501
Giảng viên hướng dẫn	: TS.Trần Vĩnh Đức

HÀ NỘI – 01/2024

Mục lục

Lời nói đầu.....	3
Phần 1: Lập trình cơ bản	4
1.1 Tuần 1: Lập trình cơ bản.....	4
1.2 Tuần 2: Độ quy.....	5
1.3 Tuần 3: Cấu trúc dữ liệu đơn giản	5
1.4 Tuần 4: Bảng băm (Hash tables).....	6
1.5 Tuần 5: Thuật toán trong đồ thị.....	6
1.6 Tuần 6: Tìm đường trong đồ thị	6
1.7 Tuần 7, 8: Tính toán và truy vấn	6
Phần 2: Dự án	8
2.1 Khóa học JavaScript Algorithms and Data Structures.....	8
2.2 Khóa học Front End Development Libraries	9
2.3 Thực hiện lại các bài code trên Hustack.....	10
Phụ lục	11

Lời nói đầu

Học phần Project I của trường Công nghệ thông tin và truyền thông cung cấp cho sinh viên các kỹ năng lập trình và phát triển ứng dụng. Trong quá trình phát triển project cá nhân của mình, sinh viên không chỉ ôn lại các kiến thức đã học mà còn có cơ hội áp dụng các kỹ thuật, các thuật toán vào trực tiếp các phần trong dự án đang phát triển. Phát triển cho sinh viên cả về tư duy lập trình cũng như cách phát triển một project cá nhân.

Trong quá trình học tập, bản thân em thông qua quá trình làm các bài tập thuật toán trên Hustack và học để lấy chứng chỉ trên FreeCodeCamp cũng đã đúc rút được cho mình các kinh nghiệm cá nhân quý giá, hỗ trợ cho việc học tập các học phần tiếp theo cũng như việc phát triển các dự án sau này.

Em xin chân thành cảm ơn giảng viên TS. Trần Vĩnh Đức đã hỗ trợ, hướng dẫn em hoàn thành học phần này. Nhờ sự định hướng của thầy giúp cho em có cơ hội tìm hiểu thêm về lập trình web và ngôn ngữ javascript, một kiến thức rất cần thiết cho tương lai và công việc phát triển phần mềm sau này của em.

Phần 1: Lập trình cơ bản

Phần đầu tiên của học phần được kéo dài trong 8 tuần đầu của học kỳ, nội dung thực hiện gồm hoàn thành 43 bài tập về lập trình được chia thành các chủ đề theo từng tuần với độ khó tăng dần. Các bài tập trong phần 1 được em thực hiện bằng ngôn ngữ C++.

1.1 Tuần 1: Lập trình cơ bản

Tuần đầu tiên của phần bao gồm 14 bài lập trình với đa dạng các chủ đề, mức độ phức tạp của các chủ đề dừng ở mức đơn giản, giúp sinh viên làm quen với các bài lập trình trong khóa học:

- Bài 1: Tính tổng của một dãy số nguyên
- Bài 2: Tìm các bội có ba chữ số của một số nguyên cho trước
- Bài 3: Bài toán giá điện
- Bài 4: Chuyển các chữ trong văn bản thành chữ hoa
- Bài 5: Kiểm tra định dạng ngày
- Bài 6: Tính bình phương của các số từ 1 đến n
- Bài 7: Đếm các số lẻ trong dãy
- Bài 8: Tính các phép tính cơ bản giữa hai số nguyên
- Bài 9: Đổi giờ sang giây
- Bài 10: Ôn tập về truy vấn
- Bài 11: Tìm số dãy con có tổng cố định
- Bài 12: Nối các dòng văn bản
- Bài 13: Thay thế các từ viết tắt
- Bài 14: Giải phương trình bậc 2

Qua 14 bài lập trình cơ bản trên, sinh viên sẽ được làm quen với lập trình thuật toán cơ bản, các cú pháp, truy vấn, xử lý sâu,... làm tiền đề cho các bài tập trong tuần tiếp theo

1.2 Tuần 2: Đề quy

Tuần thứ 2 của giai đoạn bao gồm 6 bài lập trình về đệ quy, giải các bài toán có thể sử dụng đệ quy để vét cạn các kết quả. Song song với đó, các bài toán trên hoàn toàn có thể sử dụng các phương pháp khử đệ quy giúp tối ưu tính toán.

- Bài 1: Tính tổ hợp
- Bài 2: Tìm các dãy nhị phân độ dài n
- Bài 3: Tìm các dãy nhị phân độ dài n mà không có bit 1 nào đứng cạnh nhau
- Bài 4: Tìm các hoán vị của n số
- Bài 5: Tìm số cách giải sudoku
- Bài 6: Tính phần tử thứ n của dãy Fibonacci

Các bài toán trên thường được sử dụng phương pháp đệ quy tuy nhiên có thể áp dụng phương pháp khử đệ quy bằng vòng lặp hay cây nhị phân để giải quyết

1.3 Tuần 3: Cấu trúc dữ liệu đơn giản

Ở tuần này, sinh viên sẽ được rèn luyện về các cấu trúc dữ liệu cơ bản như stack, queue hay linked list, cây dữ liệu qua 8 bài lập trình

- Bài 1: Kiểm tra trật tự của đóng mở ngoặc
- Bài 2: Bài toán đổ nước
- Bài 3: Xây dựng danh sách liên kết (Linked list)
- Bài 4: Xây dựng cây nhị phân tìm kiếm (Binary search tree)
- Bài 5: Luyện tập các cách duyệt cây nhị phân
- Bài 6: Mô phỏng Stack
- Bài 7: Mô phỏng Queue
- Bài 8: Xây dựng cây gia phả

Các cấu trúc dữ liệu trên đều là các cấu trúc dữ liệu cơ bản, có thể áp dụng được trong nhiều tình huống, có thể áp dụng để xây dựng các cấu trúc dữ liệu phức tạp hơn và biến đổi để phù hợp với bài toán cụ thể của nhà phát triển phần mềm.

1.4 Tuần 4: Bảng băm (Hash tables)

4 bài lập trình trong tuần 4 có độ khó tăng dần giúp sinh viên từng bước ôn lại về cấu trúc dữ liệu phức tạp hơn là bảng băm (Hash tables)

- Bài 1: Chuyển xâu sang hash code của bảng băm
- Bài 2: Kiểm tra số đã xuất hiện trong dãy
- Bài 3: Lưu và tìm kiếm các phần tử của bảng băm
- Bài 4: Tìm kiếm các cặp số có tổng bằng m

1.5 Tuần 5: Thuật toán trong đồ thị

Ở tuần 5, sinh viên được ôn lại các cách duyệt đồ thị đã được học trong học phần toán rời rạc, áp dụng và xử lý các bài toán đồ thị bằng máy tính

- Bài 1: Đếm số chu trình Hamilton
- Bài 2: Duyệt đồ thị theo DFS
- Bài 3: Duyệt đồ thị theo BFS
- Bài 4: Tìm cây khung nhỏ nhất trong đồ thị

1.6 Tuần 6: Tìm đường trong đồ thị

Tuần thứ 6 đem đến những bài tập về tìm đường đi ngắn nhất trong đồ thị:

- Bài 1: Tìm đường đi ngắn nhất từ một đỉnh đến các đỉnh còn lại
- Bài 2: Tìm đường đi ngắn nhất giữa hai đỉnh của đồ thị
- Bài 3: Tìm luồng cực đại giữa hai đỉnh của đồ thị

Để tìm đường đi ngắn nhất của đồ thị có thể áp dụng các thuật toán Dijkstra, Bellman-Ford, Floyd-Warshall,... tuy nhiên các đồ thị trong bài toán đều có trọng số giữa các cạnh là dương nên ta ưu tiên sử dụng Dijkstra để tối ưu tính toán

1.7 Tuần 7, 8: Tính toán và truy vấn

Tuần 7 và 8 của phần lập trình cơ bản gồm các bài toán ứng dụng thực tế trong kinh doanh, thống kê. Ở các bài toán này, sinh viên thực hiện lập trình nhận các lệnh truy vấn và thực hiện yêu cầu truy vấn theo mục đích của bài toán:

- Bài 1 tuần 7: Thống kê giao dịch ngân hàng
- Bài 2 tuần 7: Thống kê, phân tích đơn đặt hàng qua mạng
- Bài 1 tuần 8: Thống kê nộp bài của sinh viên trong một cuộc thi
- Bài 2 tuần 8: Thống kê dữ liệu dân cư

Phần 1 của học phần mang đến cho sinh viên một giai đoạn lập trình xây dựng nền tảng trong suốt 8 tuần. Từ những bài tập lập trình cơ bản đến những khái niệm đối với đệ quy, cấu trúc dữ liệu, và thuật toán trong đồ thị, sinh viên không chỉ học được kiến thức mà còn phát triển kỹ năng lập trình và tư duy logic.

Các bài tập mang đến sự đa dạng với độ khó tăng dần, giúp sinh viên vượt qua từng thách thức và hiểu sâu hơn về cách áp dụng kiến thức vào thực tế. Việc ôn tập và luyện tập với các cấu trúc dữ liệu như Stack, Queue, Linked list, Binary search tree và Hash tables giúp sinh viên xây dựng nền tảng vững chắc cho công việc lập trình phức tạp hơn trong tương lai.

Các bài toán thực tế ở những tuần cuối cùng là cơ hội tốt để sinh viên ứng dụng những kiến thức đã học vào các tình huống thực tế, từ việc thống kê giao dịch ngân hàng đến phân tích đơn đặt hàng qua mạng. Qua học phần này, sinh viên không chỉ nắm vững kiến thức cơ bản mà còn phát triển kỹ năng giải quyết vấn đề và sự sáng tạo trong lập trình.

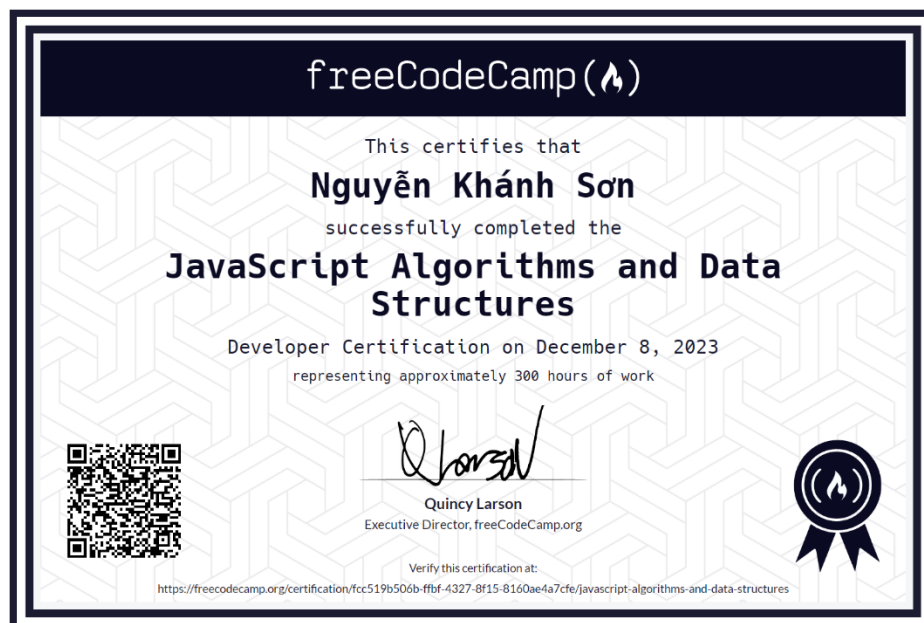
Phần 2: Dự án

Ở phần hai của học phần, dưới sự hướng dẫn, định hướng của TS. Trần Vĩnh Đức, em đã hoàn thiện các khóa học lấy chứng chỉ bao gồm JavaScript Algorithms and Data Structures và Front End Development Libraries trên trang FreeCodeCamp đồng thời thực hiện lại các bài tập trong phần 1 bằng ngôn ngữ JavaScript.

2.1 Khóa học JavaScript Algorithms and Data Structures

Khóa học JavaScript Algorithms and Data Structures trên FreeCodeCamp bao gồm các bài tập giúp lập trình viên làm quen với ngôn ngữ lập trình JavaScript và các thuật toán và cấu trúc dữ liệu của nó. JavaScript là một ngôn ngữ có các cách tiếp cận thuật toán đặc biệt hơn so với các ngôn ngữ bậc thấp và trung như C hay C++, JavaScript có thể thực hiện các truy vấn, tìm kiếm và xử lý thông tin qua regex, thực hiện xử lý các dãy thông qua các function đặc biệt giúp việc lập trình dễ dàng hơn và vô số các tính năng khác.

Sau khi hoàn thành khóa học này, em đã có thêm kiến thức căn bản về JavaScript và rút ra được một số kinh nghiệm lập trình cho bản thân làm tiền đề cho các bước tiếp theo trong phần dự án. Đồng thời thấy được rằng JavaScript là một ngôn ngữ lập trình bậc cao có ứng dụng lớn trong xây dựng phần mềm đặc biệt là phần mềm web, tuy nhiên đây là một ngôn ngữ dễ tiếp cận với các cú pháp đơn giản, cách tổ chức dữ liệu dễ làm quen, giúp các lập trình viên dễ dàng sử dụng và ứng dụng cho các sản phẩm của mình.

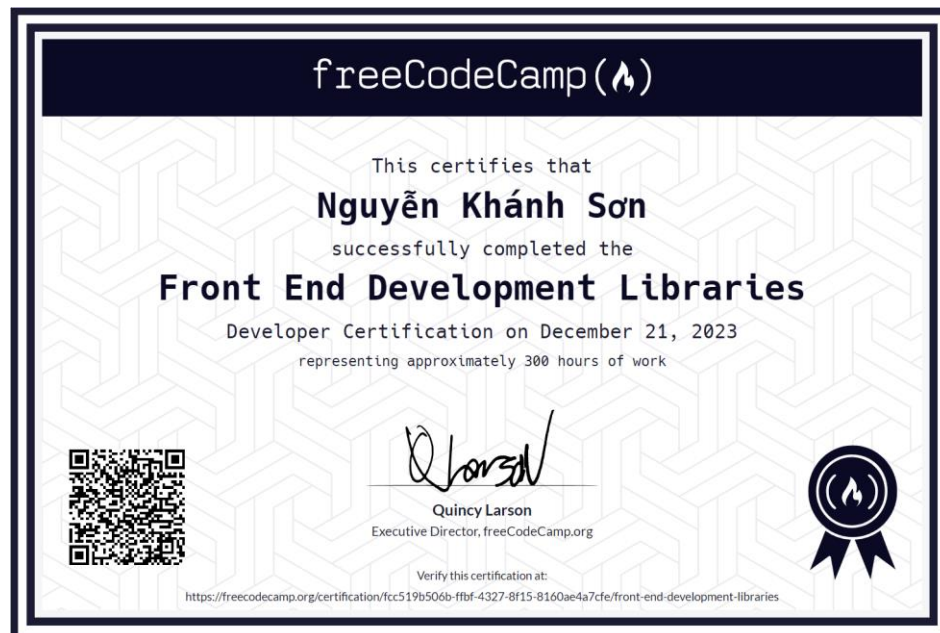


2.2 Khóa học Front End Development Libraries

Khóa học Front End Development của FreeCodeCamp cung cấp cho người học các kiến thức về xây dựng một giao diện web bằng HTML, CSS và JavaScripts. Trong khóa học, em được học về các framework và thư viện ứng dụng trong xây dựng web như Bootstrap, jQuery, React và Redux.

- Bootstrap: là một thư viện thông dụng trong việc xây dựng responsive web và mobile-first web.
- jQuery: là thư viện giúp đơn giản hóa sự tương tác giữa JavaScript và HTML, CSS, các sự kiện, hoạt ảnh,...
- React và Redux thường được sử dụng kết hợp để quản lý trạng thái của các web app. Trong khi giúp cho việc xây dựng các ứng dụng web trở nên dễ dàng hơn bằng cách tách các thành phần UI thành các phần nhỏ hơn, dễ quản lý hơn thì Redux giúp quản lý trạng thái cho các ứng dụng JavaScript một cách nhất quán

Đặc biệt, kết thúc khóa học, em được hoàn thành các mini project là các web app giúp ứng dụng các kiến thức đã học về quản lý và chuyển trạng thái web, quản lý thời gian trên web, ...



2.3 Thực hiện lại các bài code trên Hustack

Trong giai đoạn cuối cùng của phần 2, em thực hiện lập trình lại 43 bài code trong phần 1 bằng JavaScript, và xây dựng một giao diện đơn giản để test các bài code đó.

Khác với C++, JavaScript đôi khi có các cách tiếp cận và giải quyết bài toán khác với C++. JavaScript tiện lợi hơn C++ ở các cách truy vấn trong dãy, sử dụng regex, việc triển khai Object trong JavaScript cũng dễ dàng hơn. Tuy nhiên, JavaScript không hỗ trợ thư viện của các cấu trúc dữ liệu như queue, stack, hashtable,... nên phải xây dựng lại các cấu trúc dữ liệu đó nếu muốn sử dụng, việc xây dựng lại có thể hạn chế trong đơn giản hóa code nhưng cũng giúp em ôn tập lại kiến thức về các cấu trúc dữ liệu đó và cách để triển khai cấu trúc dữ liệu.

Project I Nguyễn Khánh Sơn 20215471	Description An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed). However, if an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling. Note: Absolute positioned elements are removed from the normal flow, and can overlap elements. Here is a simple example:	Input: <div></div> <div>Solve</div> Output: <div></div>
Week 1		
Week 2		
Week 3		
Week 4		
Week 5		
Week 6		
Week 7		
Week 8		

Phụ lục

Link chứng chỉ [JavaScript Algorithm and DataStructures](#)



Link chứng chỉ [Front End Development Libraries](#)



Link github: [Sonyaaa211/Project-I-IT3150 \(github.com\)](#)

