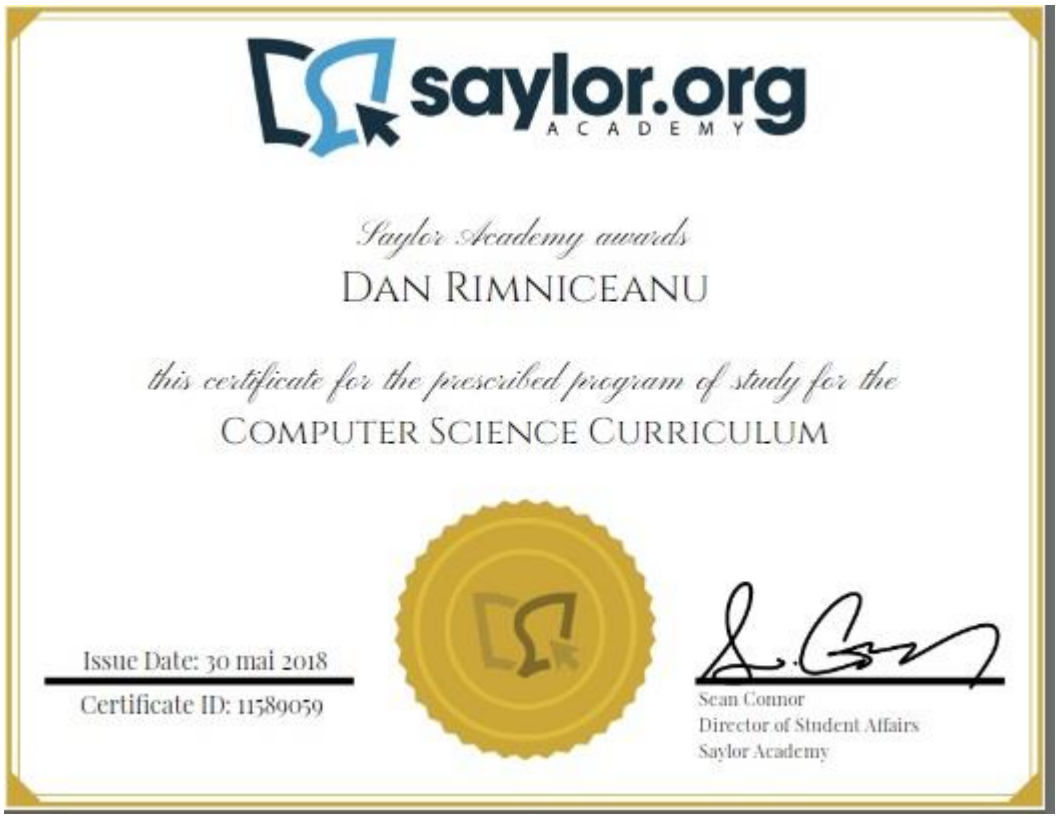
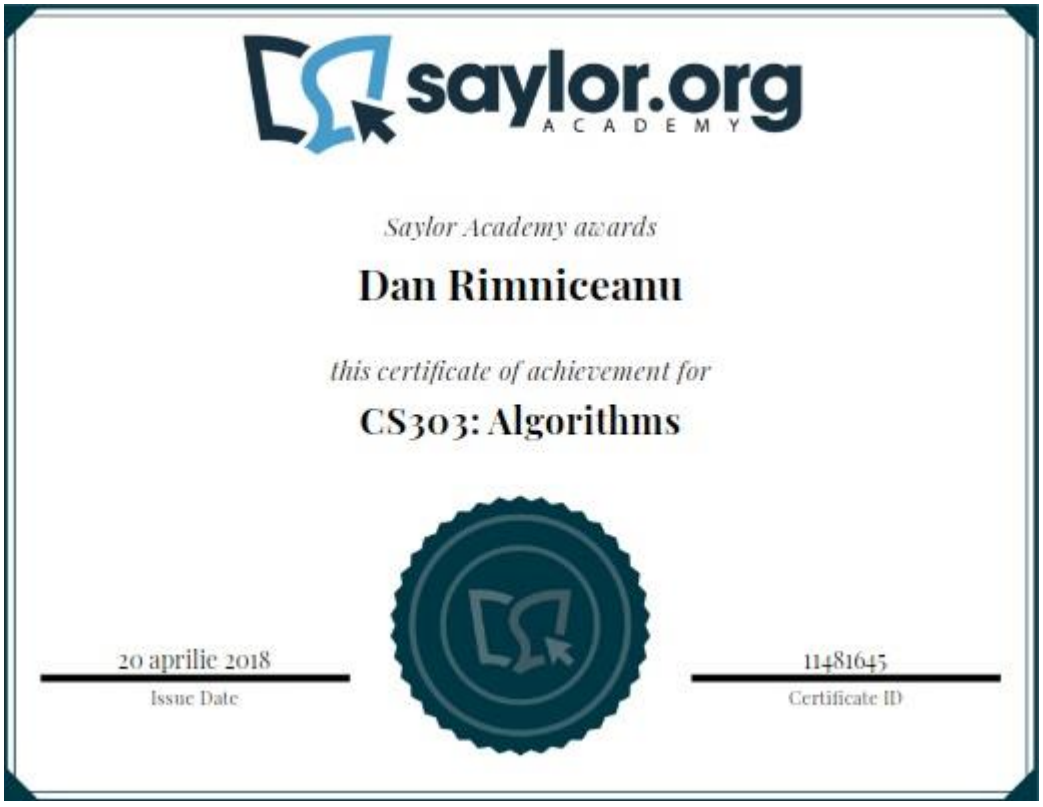

saylor.org
ACADEMY
Saylor Academy awards
DAN RIMNICEANU
this certificate for the prescribed program of study for the
COMPUTER SCIENCE CURRICULUM
Issue Date: 30 mai 2018
Certificate ID: 11589059
Sean Connor
Director of Student Affairs
Saylor Academy


saylor.org
ACADEMY
Saylor Academy awards
Dan Rimniceanu
this certificate of achievement for
CS303: Algorithms
20 aprilie 2018
Issue Date
11481645
Certificate ID

This course focuses on the fundamentals of computer algorithms, emphasizing methods useful in practice. We look into the algorithm analysis as a way to understand behavior of computer programs as a function of its input size. Using the big-O notation, we classify algorithms by their efficiency. We look into basic algorithm strategies and approaches to problem solving. Some of these approaches include the divide and conquer method, dynamic programming, and greedy programming paradigms. Sorting and searching algorithms are discussed in detail as they form part of a solution to a large number of problems solved using computers. We also provide an introduction to the graph theory and graph algorithms as they are also used in many computer-based applications today. We conclude the course with a look into a special class of problems called the NP-complete problems.

Unit 1: Introduction to Algorithms

This unit introduces what algorithms are and discusses their importance with the role that algorithms play compared to other technologies used in computers. We look into description of algorithms using pseudo-code and we use pseudo-code for algorithmic analysis. We will go through an introduction of algorithms using examples of sorting algorithms while discussing the importance of algorithm analysis in that context.

Completing this unit should take you approximately 5 hours.

- Upon successful completion of this unit, you will be able to:
  - Explain and identify the importance of algorithms in modern computing systems and their place as a technology in the computing industry.
  - Identify algorithms as a pseudo-code to solve some common problems.

- 1.1: Introduction to Algorithms
  - Massachusetts Institute of Technology: Dr. Charles E. Leiserson's "Merge Sort"Page

    Watch this video to learn the basics of the algorithm. You can skip the first 17 minutes of the video as they talk about MIT class related logistics for the course.

- 1.2: Introduction to Framework for Algorithm Analysis
  - Indian Institute of Technology, Bombay: Dr. Abhiram Ranade's "Framework for Algorithm Analysis"Page

    Watch this video to learn about the basics of the algorithm analysis and associated framework.

- 1.3: The Importance of Algorithms

- Topcoder: "Importance of Algorithms"URL
  Read this article for an overview of importance of algorithms as well as a listing of some of the key algorithm areas.

- 1.4: Control Instructions
  - *Algorithms*: "Introduction"URL
    Read this page to get an introduction to algorithms.
  -  Introduction to Algorithms AssignmentURL
    Complete all questions in this assignment. There are three questions on finding the complexity of the algorithm using the pseudo code and finding the number of instructions executed to solve the problem. Each instruction is associated with some constant cost for execution. You can check your answers against the Answer Key.

Unit 2: Introduction to Analysis of Algorithms

In this unit, we explore how we can express an algorithm's efficiency as a function of its input size. The order of growth of running time of an algorithm gives a simple characterization of algorithm's efficiency and allows us to relate performance of alternative algorithms. Asymptotic analysis is based on the idea that as the problem size grows, the complexity will eventually settle down to a simple proportionality to some known function. This idea is incorporated in the "Big Oh", "Big Omega", and "Big Theta" notations for asymptotic performance. These notations are useful for expressing the complexity of an algorithm without getting lost in unnecessary detail.

Completing this unit should take you approximately 9 hours.

- Upon successful completion of this unit, you will be able to:
  - Describe asymptotic notations for bounding algorithm running times from above and below.
  - Explain methods for solving recurrences useful in describing running times of recursive algorithms.
  - Explain the use of Master Theorem in describing running times of recursive algorithms.

- 2.1: Introduction to Algorithms
  - Massachusetts Institute of Technology: Dr. Erik Demaine's "Introduction to Algorithms"Page
    Watch this video to learn about the basics of the algorithm.

- 2.2: Asymptotic Analysis

- University of California, Berkeley: Dr. Jonathan Shewchuk's "Asymptotic Analysis"Page

  Watch this video to learn about the ideas behind the use of asymptotic analysis in algorithms.

- 2.3: Introduction to Analysis of Algorithms
  - University of California, Berkeley: S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani's "Algorithms: Prologue"URL

    Read the Prologue of S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani's book *Algorithms*.

- 2.4: Master Theorem
  - Wikipedia: "Master Theorem"URL

    Read this article to learn about the use of Master Theorem in analyzing recursive problems.
  - Introduction to Analysis of Algorithms AssignmentURL

    Complete all questions in this assignment. There are six questions listed in two parts. The first part requires solving the problems using the Master Theorem discussed in the lectures. The second part requires solving for complexity of the algorithms using the first principles. You can check your answers against the Answer Key.

Unit 3: Divide and Conquer Method

In this unit, we will examine a popular technique called divide-and-conquer that is used in solving computer science problems. This technique solves the problem by breaking up the problem into smaller problems of same type and then recursively solving these smaller problems and combining their answers. We will also look into analysis of these algorithms through the use of recursion techniques.

Completing this unit should take you approximately 10 hours.

- Upon successful completion of this unit, you will be able to:
  - Explain methods for solving recurrences useful in describing running times of recursive algorithms.
  - Describe divide-and-conquer recursive technique for solving a class of problems.

- 3.1: Introduction to Divide and Conquer Algorithms

- University of California, Berkeley: S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani's "Divide-and-Conquer Algorithms"URL

  Read Chapter 2 of S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani's book *Algorithms*.

- 3.2: Recurrences in Algorithms
  - Massachusetts Institute of Technology: Dr. Erik Demaine's "Introduction to Algorithms"Page

    Watch this lecture to learn about the basics of the algorithm.

- 3.3: Recursion
  - Wikipedia: "Recursion"URL

    Read this article to learn about the mathematical concepts behind the idea of recursion.
  - Divide and Conquer AssignmentURL

    Complete all questions in this assignment. There are two questions on the divide-and-conquer strategy. The first one involves a search strategy, whereas the second one involves a multiplication strategy. You can check your answers against the Answer Key.

Unit 4: Sorting Algorithms

This unit introduces algorithms that sort real numbers. Algorithms often use sorting as a key subroutine. There is a wide variety of sorting algorithms, and they use a rich set of techniques. These algorithms have different runtime complexities and work better in certain conditions. Some of algorithms that we will study include Quick Sort, Insertion Sort, Bubble Sort, and Merge Sort.

Completing this unit should take you approximately 7 hours.

- Upon successful completion of this unit, you will be able to:
  - Describe sorting algorithms and their runtime complexity analysis.

- 4.1: Introduction to Sorting Algorithms
  - Knight School: "Algorithmic Thinking"Page

    Watch this video to learn about sorting algorithms.

  - "Google Interview with Barack Obama"Page

Watch this video to learn about sorting algorithms.

- 4.2: Sorting Algorithms - Part I
    - Massachusetts Institute of Technology: Dr. Charles E. Leiserson's "Quicksort, Randomized Algorithms"Page

      Watch this video to learn about the basics of sorting algorithms.

- 4.3: Sorting Algorithms - Part II
    - Massachusetts Institute of Technology: Dr. Erik Demaine's "Introduction to Algorithms"Page

      Watch this video to learn about various types of sorting algorithms.

- 4.4: Popular Sorting Algorithms
    - Wikipedia: "Sorting Algorithms"URL

      Read this article to learn about the popular sorting algorithms in use today.
    -  Sorting AssignmentURL

      Please complete all questions in this assignment. There are two questions on the sorting algorithms. The first one involves the merge-sort algorithm and requires you to work through the merging part of the algorithm. The second one then asks you to implement the merging work that you just completed in the first problem. You can check your answers against the Answer Key.

Unit 5: Dynamic Programming

In this unit, we will study another popular computer science algorithmic paradigm called the dynamic programming. Dynamic programming, like the divide-and-conquer method, solves problem by combining solutions to sub-problems. Dynamic programming typically applies to optimization problems in which a set of choices must be made in order to arrive at an optimal solution. It is effective when a given sub-problem may arise from more than one partial set of choices. We will look into problems, such as the longest common subsequence and the knapsack problem, to explain the key ideas behind dynamic programming.

Completing this unit should take you approximately 7 hours.

- Upon successful completion of this unit, you will be able to:
    - Describe the dynamic programming technique for solving a class of problems.

- 5.1: Introduction to Dynamic Programming

- University of California, Berkeley: S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani's "Dynamic Programming"URL

  Read Chapter 6 of S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani's book *Algorithms*.

- 5.2: Dynamic Programming
  - Massachusetts Institute of Technology: Dr. Charles E. Leiserson's "Dynamic Programming, Longest Common Subsequence"Page

    Watch this video to learn about dynamic programming concepts.

- 5.3: The Knapsack Problem
  - James Bedford's "Dynamic Programming - The Knapsack Problem"Page

    Watch this video series to learn about how the knapsack problem is solved using dynamic programming techniques.

- 5.4: Dynamic Programming Examples
  - Wikipedia: "Dynamic Programming"URL

    Read this article to learn about the different types of problems to which dynamic programming techniques are applied.
  - Dynamic Programming AssignmentURL

    Complete all questions in this activity. There is one question on the dynamic-programming paradigm that requires two implementations. The first one involves the use of the regular approach to matrix multiplication, and the second one requires the dynamic programming approach. You have to compare the runtimes for the two approaches. You can check your answers against the Answer Key.

Unit 6: Graph Theory and Graph Algorithms

In this unit, you will learn about graph theory and graph-based algorithms. Graphs are a pervasive data structure in computer science and algorithms working with them are fundamental to the subject. We will review basic concepts of graph and associated terminology. We will also see how we can represent graphs in computer algorithms and use these representations to solve some common problems, such as finding the shortest paths between any two places. You will also get an introduction to trees and a minimum weight spanning tree algorithm.

Completing this unit should take you approximately 10 hours.

- Upon successful completion of this unit, you will be able to:

- Describe concepts in graph theory, graph-based algorithms, and their analysis.
- Describe tree-based algorithms and their analysis.

- 6.1: Introduction to Graph Theory
  - University of California, San Diego: Edward A. Bender and S. G. Williamson's "Basic Concepts in Graph Theory"URL

    Read this chapter for an introduction to graph theory.

- 6.2: Paths in Graphs
  - University of California, Berkeley: S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani's "Paths in Graphs"URL

    Read Chapter 4 from S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani's book *Algorithms*.

- 6.3: Graph Data Structures
  - PatrickJMT: "Graph Theory - An Introduction"Page

    Watch this video to learn about data structures used in graph algorithms.

- 6.4: Graph Theory Algorithms - Part I
  - Massachusetts Institute of Technology: Dr. Charles E. Leiserson's "Greedy Algorithms, Minimum Spanning Trees"Page

    Watch this video to learn about graph theory concepts, greedy algorithms, and minimum spanning trees.

- 6.5: Graph Theory Algorithms - Part II
  - Massachusetts Institute of Technology: Dr. Charles E. Leiserson's "Shortest Paths I: Properties, Dijkstra's Algorithm, Breadth-first Search"Page

    Watch this video to learn about shortest path problems.
  - Graph Theory and Graph Algorithms AssignmentURL

    Complete all questions in this assignment. There is one question on the breadth-first search implementation and one on depth-first search implementation. You can check your answers against the Answer Key.

Unit 7: Greedy Algorithms

In this unit, we will look into a common computer science algorithm technique called the greedy algorithms. Like the dynamic programming paradigm, greedy algorithms typically

apply to optimization problems in which a set of choices must be made in order to arrive at an optimal solution. The idea of greedy algorithm is to make each choice in a locally optimal manner. We will explore some common greedy algorithms in use today as a way of explaining the topic in this unit.

Completing this unit should take you approximately 7 hours.

- Upon successful completion of this unit, the student will be able to:
  - Describe greedy algorithms and their applications.

- 7.1: Introduction to Greedy Algorithms
  - University of California, Berkeley: S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani's "Greedy Algorithms"URL

    Read Chapter 5 of S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani's book *Algorithms*.

- 7.2: Greedy Algorithms - Part I
  - Indian Institute of Technology, Bombay: Dr. Sunder Viswanathan's "Greedy Algorithms - I"Page

    Watch this video to learn about greedy algorithms.

- 7.3: Greedy Algorithms - Part II
  - Indian Institute of Technology, Bombay: Dr. Sunder Viswanathan's "Greedy Algorithms - II"Page

    Watch this video to learn about greedy algorithms.

- 7.4: Greedy Algorithms - Part III
  - Indian Institute of Technology, Bombay: Dr. Sunder Viswanathan's "Greedy Algorithms - III"Page

    Watch this video to learn about greedy algorithms.

  - Wikipedia: "Recursion"URL

    Read this article to learn about the mathematical concept behind the idea of recursion.

  - Greedy Algorithms AssignmentURL

    Complete all questions in this activity. There is one question on the minimum spanning tree problem requiring two implementations using the Kruskal's

algorithm and the Prim's algorithm. You can check your answers against the Answer Key.

Unit 8: NP-Completeness

In this last unit, we will study a special class of problems called the NP-complete problems. Many interesting computational problems are NP-complete, but there are no polynomial-time algorithms known for solving any of them. The unit presents techniques for determining when a problem is NP-complete.

Completing this unit should take you approximately 11 hours.

- Upon successful completion of this unit, the student will be able to:
  - Explain the classification of difficult computer science problems as belonging to P, NP, and NP-hard classes.

- 8.1: Introduction to NP-Completeness
  - University of California, Berkeley: S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani's "NP-Complete Problems"URL

    Read Chapter 8 of S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani's book *Algorithms*.

- 8.2: NP-Completeness - Part I
  - Indian Institute of Technology, Bombay: Dr. Sunder Viswanathan's "NP-Completeness - Part I"Page

    Watch this video to learn about NP-Complete problems in the context of computer algorithms.

- 8.3: NP-Completeness - Part II
  - Indian Institute of Technology, Bombay: Dr. Sunder Viswanathan's "NP-Completeness - Part II"Page

    Watch this video to learn about NP-Complete problems in the context of computer algorithms.

- 8.4: NP-Completeness - Part III
  - Indian Institute of Technology, Bombay: Dr. Sunder Viswanathan's "NP-Completeness - Part III"Page

    Watch this video to learn about NP-Complete problems in the context of computer algorithms.

- 8.5: NP-Completeness - Part IV
    - Indian Institute of Technology, Bombay: Dr. Sunder Viswanathan's "NP-Completeness - Part IV"Page

      Watch this video to learn about NP-Complete problems in the context of computer algorithms.
- 8.6: NP-Completeness - Part V
    - Indian Institute of Technology, Bombay: Dr. Sunder Viswanathan's "NP-Completeness - Part V"Page

      Watch this video to learn about NP-Complete problems in the context of computer algorithms.
    -  NP-Completeness AssignmentURL

      Complete all questions in this assignment. There are two questions that are both related to proving NP-Completeness of a given problem. You can check your answers against the Answer Key.

Stanford | ONLINE
STATEMENT OF ACCOMPLISHMENT

April 1, 2017

Dan Rimniceanu

has successfully completed a free online offering of

Algorithms: Design and Analysis

This is an undergraduate level course on the design and analysis of algorithms. The main topics are: asymptotic analysis, divide and conquer algorithms, sorting and searching, basic randomized algorithms, graph search, shortest paths, heaps, search trees, and hash tables. In order to earn a Statement of Accomplishment, participants were required to score at least 70% on 6 problem sets, 6 programming assignments, and 1 final exam.

Tim Roughgarden
Associate Professor of Computer Science
Stanford University

PLEASE NOTE: SOME ONLINE COURSES MAY DRAW ON MATERIAL FROM COURSES TAUGHT ON-CAMPUS BUT THEY ARE NOT EQUIVALENT TO ON-CAMPUS COURSES. THIS STATEMENT DOES NOT AFFIRM THAT THIS PARTICIPANT WAS ENROLLED AS A STUDENT AT STANFORD UNIVERSITY IN ANY WAY. IT DOES NOT CONFER A STANFORD UNIVERSITY GRADE, COURSE CREDIT OR DEGREE, AND IT DOES NOT VERIFY THE IDENTITY OF THE PARTICIPANT.

Authenticity can be verified at **https://verify.lagunita.stanford.edu/SOA/15851fa9146449dd91e5e8e8937be8c3**

Stanford | ONLINE
STATEMENT OF ACCOMPLISHMENT

September 5, 2017

Dan Rimniceanu

has successfully completed a free online offering of

Algorithms: Design and Analysis, Part 2

This course covers greedy algorithms, including applications to minimum spanning trees and Huffman codes; dynamic programming, including applications to sequence alignment and shortest-path problems; and exact and approximation algorithms for NP-complete problems. In order to earn a Statement of Accomplishment, participants were required to score at least 70% on 6 problem sets, 6 programming assignments, and 1 final exam.

Tim Roughgarden
Associate Professor of Computer Science
Stanford University

PLEASE NOTE: SOME ONLINE COURSES MAY DRAW ON MATERIAL FROM COURSES TAUGHT ON-CAMPUS BUT THEY ARE NOT EQUIVALENT TO ON-CAMPUS COURSES. THIS STATEMENT DOES NOT AFFIRM THAT THIS PARTICIPANT WAS ENROLLED AS A STUDENT AT STANFORD UNIVERSITY IN ANY WAY. IT DOES NOT CONFER A STANFORD UNIVERSITY GRADE, COURSE CREDIT OR DEGREE, AND IT DOES NOT VERIFY THE IDENTITY OF THE PARTICIPANT.

Authenticity can be verified at **https://verify.lagunita.stanford.edu/SOA/e40649277d9e47d1a30950f36afa8dd9**

saylor.org
ACADEMY

Saylor Academy awards

DAN RIMNICEANU

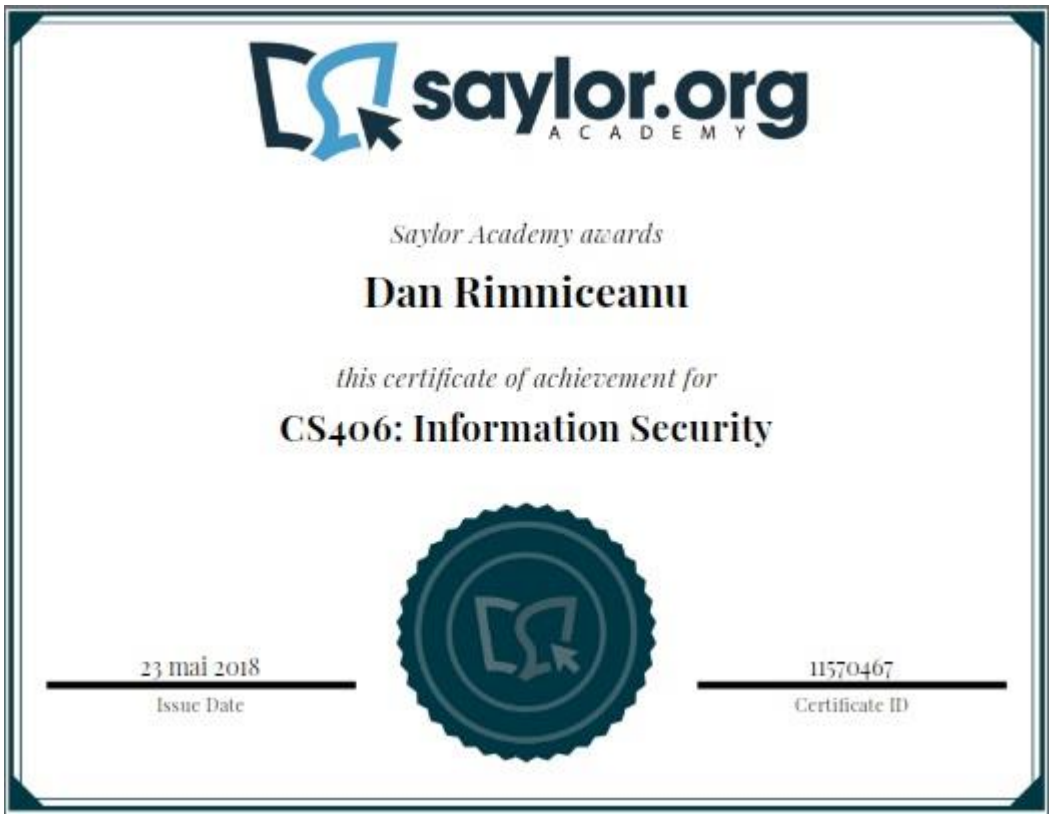
this certificate for the prescribed curriculum of study for the

MINOR IN COMPUTER SCIENCE

Issue Date: 24 mai 2018

Certificate ID: 11574603

Devon Ritter
Director of Education,
Saylor Academy



saylor.org
ACADEMY

Saylor Academy awards

Dan Rimniceanu

this certificate of achievement for

CS406: Information Security

23 mai 2018

Issue Date

11570467

Certificate ID

CS406: Information Security

Course Introduction

This course focuses on the fundamentals of information security that are used in protecting both the information present in computer storage as well as information traveling over computer networks. Interest in information security has been spurred by the pervasive use of computer-based applications such as information systems, databases, and the Internet. Information security has also emerged as a national goal in the United States and in other countries with national defense and homeland security implications. Information security is enabled through securing data, computers, and networks. In this course, we will look into such topics as fundamentals of information security, computer security technology and principles, access control mechanisms, cryptography algorithms, software security, physical security, and security management and risk assessment. By the end of this course, you will be able to describe major information security issues and trends, and advise an individual seeking to protect his or her data.

Unit 1: Computer Security Concepts

This unit provides an overview of information security. First, we look at the basic concepts of confidentiality, integrity, and availability as discussed in the National Institute of Standards and Technology (NIST) standard Federal Information Processing Standards (FIPS) 199. We will discuss threats, attacks, and assets in the overall context of a security management model. We will also learn about the challenges of information security and its overall scope.

Completing this unit should take you approximately 6 hours.

- Unit 1 Learning OutcomesPage

- 1.1: Introduction to Information Security
  - The Open University: "An Introduction to Information Security"Page

    Read these sections, which introduce information security.

- 1.2: Introduction to Data and Network Security
  - George Mason University: Paul A. Strassman's "Information Assurance for Defense Security"Page

    Watch this lecture to learn about the methods for managing risks to information assets. IT practitioners seek to protect the confidentiality, integrity, and availability

of data and their delivery systems - whether the data are in storage, in processing, or in transit, and whether threatened by malice or accident.

- 1.3: Confidentiality, Integrity, and Availability
    - University of Miami School of Medicine: "Confidentiality, Integrity, and Availability"URL

      Read this page for an overview of the basic security concepts of confidentiality, integrity, and availability.

- 1.4: NIST FIPS 199 Standard
    - National Institute of Standards and Technology: "Standards for Security Categorization of Federal Information and Information Systems"File

      Read this document to gain a better understanding of the security objectives of confidentiality, integrity, and availability.

- 1.5: Assets and Threats
    - Robert J. Shimonski's "Threats and Your Assets: What Is Really at Risk?"URL

      Read this article for an introduction to the types of information assets and associated threats.

Unit 2: Basic Cryptographic Concepts

Encryption and decryption of data form the basis of information security. Cryptography is the science of encrypting data. In this unit, we will explore basic cryptography concepts and the purpose of cryptography. Also, we will look into the details of symmetric key encryption techniques. In particular, we will discuss the symmetric key algorithms called Data Encryption Standard (DES), Triple DES, and Advanced Encryption Standard (AES). DES is a block-cipher employing a 56-bit key that operates on 64-bit blocks. Triple-DES is a variant of DES that employs up to three 56-bit keys and makes three encryption/decryption passes over the block. In 1997, NIST initiated a very public, four-and-a-half-year process to develop a new secure cryptosystem for U.S. government applications. The result, the Advanced Encryption Standard, became the official successor to DES in December 2001.

Completing this unit should take you approximately 7 hours.

- Unit 2 Learning OutcomesPage

- 2.1: Basic Cryptography Concepts: Symmetric Encryption Algorithms
    - Steve Weis' "Theory and Practice of Cryptography"Page

Watch this video about the basics of information security and cryptographic concepts related to symmetric encryption algorithms like DES, Triple DES, and AES.

- 2.2: Purpose of Cryptography
    - Gary C. Kessler's "An Overview of Cryptography: The Purpose of Cryptography"URL

      Read section 2 about basic cryptographic concepts.

- 2.3: Data Encryption Standard (DES)
    - Gary C. Kessler's "An Overview of Cryptography: Some of the Finer Details of DES, Breaking DES, and DES Variants"URL

      Read section 5.4 about DES symmetric key cryptography algorithm.

- 2.4: Triple DES
    - Tropical Software: "Triple DES Encryption"URL

      Read this page about how Triple DES makes use of DES to improve on encryption-based security.

- 2.5: Advanced Encryption Standard (AES)
    - Gary C. Kessler's "An Overview of Cryptography: The Advanced Encryption Standard and Rijndael"URL

      Read section 5.9 about the advanced encryption standard (AES) algorithm.

Unit 3: Public-Key Encryption

In this unit, we explore basic cryptography concepts and look into the details of asymmetric key encryption techniques based on the concepts of a public-key. You will be able to compare and contrast public-key algorithms and symmetric algorithms discussed in Unit 2. In particular, we will discuss the public-key algorithms by Rivest, Shamir, and Adelman (RSA) and another algorithm by Diffie and Hellman.

Completing this unit should take you approximately 12 hours.

- Unit 3 Learning OutcomesPage

- 3.1: Introduction to Public-Key Cryptography
    - Whitfield Diffie's "Before, During, and After Public-Key Cryptography"Page

Watch this video about the origins of cryptographic concepts. Whitfield Diffie, a key figure in the discovery of public-key cryptography, traces the growth of information security through the 20th century and into the 21st. In the 1970s, the world of information security was transformed by public-key cryptography, the radical revision of cryptographic thinking that allowed people with no prior contact to communicate securely. Public-key solved security problems born of the revolution in information technology that characterized the 20th century and made Internet commerce possible. Security problems rarely stay solved, however. Continuing growth in computing, networking, and wireless applications have given rise to new security problems that are already confronting us.

- 3.2: Public-Key Encryption Algorithms
    - Naval Postgraduate School: "Public Key Cryptography"URL

      Watch these videos about cryptographic concepts related to public-key algorithms, such as the RSA algorithm and the Diffie-Hellman algorithm and how they are used in network security.

        - Cryptography Basics
        - Symmetric and Public Key Cryptography
        - Network Authentication through Cryptography
        - PKI
- 3.3: Public-Key Cryptography
    - Gary C. Kessler's "An Overview of Cryptography: Public-Key Cryptography"URL

      Read section 3.2 about the key concepts behind public-key cryptography. After reading this section, explain the history of public-key cryptography, the factorization problem, and describe how RSA works.

- 3.4: RSA Public-Key Algorithm
    - Gary C. Kessler's "An Overview of Cryptography: Some of the Finer Details of RSA Public-Key Cryptography"URL

      Read section 5.3 about the steps in the RSA Public-Key Algorithm. After reading, you should be able to describe a simple example of generating public/private keys for RSA systems and describe the process of encrypting and decrypting a message.

- 3.5: Diffie-Hellman Algorithm
    - Gary C. Kessler's "An Overview of Cryptography: Some of the Finer Details of Diffie-Hellman"URL

Read section 5.2 about the steps in the Diffie-Hellman Public-Key Algorithm.

- 3.6: Cryptography in Practice
    - CrypTool: http://www.cryptool.org/URL

      Download this software, try to use different methods to encrypt messages, and then try to use the analysis tools to analyze the entropy such as floating frequency, histogram, N-Gram, autocorrelation, and periodicity, etc. Also try to use symmetric key ciphers such as DES and asymmetric ciphers such as RSA, DH, etc.

Unit 4: Access Control Mechanisms

Access control is a system that enables an authority to control access to areas and resources in a given physical facility or computer-based information system. In this unit, we will explore the access control mechanisms for user authorization. By the means of access control, appropriate authorization to information is provided to different entities in an organization. The common mechanisms include discretionary access control (DAC) and role-based access control (RBAC). We look into each of these in the context of their current usage in a typical enterprise.

Completing this unit should take you approximately 7 hours.

- Unit 4 Learning OutcomesPage

- 4.1: Authentication
    - Open Web Application Security Project: "Authentication"URL

      Read this chapter about authentication, a process of determining if a user or entity is who he/she claims to be.

- 4.2: Access Control and Authorization
    - Open Web Application Security Project: "Access Control and Authorization"URL

      Read this chapter about discretionary access control (DAC) and role-based access control (RBAC), a technical means for controlling access to computer resources.

- 4.3: Role-Based Access Control
    - National Institute of Standards and Technology: "An Introduction to Role-Based Access Control"URL

Read this page about role-based access control (RBAC), a technical means for controlling access to computer resources.

- 4.4: Role-Based Access Control and Role Graph Model
  - Purdue University: Sylvia Osborn's "The Role Graph Model and Its Extensions"Page

    Watch this video about techniques used in context of Role-Based Access Control mechanism.

Unit 5: Security Solutions

In this unit, we explore some of the common solutions for security issues that are currently in use. For securing web-based applications such as e-Commerce, Secure Sockets Layer (SSL) is a commonly used solution to enable security of transactions. It makes use of public-key based encryption and symmetric encryption during the transaction to ensure security. We also look into a protocol called Internet Protocol Security (IPSec) to secure communications.

Completing this unit should take you approximately 5 hours.

- Unit 5 Learning OutcomesPage

- 5.1: Security Protocols and Solutions
  - Indian Institute of Technology, Kharagpur: Indranil Sengupta's "Basic Cryptographic Concepts"Page

    Watch this video about cryptographic concepts related to Secure Sockets Layer (SSL), Secure Shell (SSH), and Internet Protocol Security (IPSec).

- 5.2: Internet Protocol Security
  - Gary C. Kessler's "An Overview of Cryptography: IP Security (IPSec) Protocol"URL

    Read section 5.6 about Internet Protocol for securing communications. After reading this section, describe the two modes for IPSec: AH and ESP. Also try to explain how to use AH and ESP to build VPN (tunnel mode and transport mode).

- 5.3: Secure Sockets Layer
  - Gary C. Kessler's "An Overview of Cryptography: The SSL Family of Secure Transaction Protocols for the World Wide Web"URL

Read section 5.7 about the SSL family of protocols for securing transactions over the Internet. When reading this section, please pay special attention to the diagram in Figure 5. You need to be able to explain the message flows in Figure 5 for SSL/TLS.

- 5.4: Pretty Good Privacy
  - Gary C. Kessler's "An Overview of Cryptography: Pretty Good Privacy (PGP)"URL

    Read section 5.5 about Pretty Good Privacy (PGP), one of today's most widely used public-key cryptography programs.

Unit 6: Firewalls, Intrusion Detection, and Intrusion Prevention

In this unit, we will explore the use of security tools such as firewalls and intrusion prevention systems. Following a quick introduction to the concepts of intranet and extranet systems that are frequently used for information exchange by enterprises today, we will look into cryptographic concepts related to securing communication using firewalls. We will explore how firewalls work and will also study different types of intrusion detection systems including host-based and network-based systems.

Completing this unit should take you approximately 9 hours.

- Unit 6 Learning OutcomesPage

- 6.1: Security Protocols and Solutions
  - Indian Institute of Technology, Kharagpur: Indranil Sengupta's "Intranet, Extranet, Firewall"Page

    Watch this video. Following a brief introduction to intranets and extranets used frequently today by businesses, Sengupta explains cryptographic concepts related to securing communication using firewalls.

- 6.2: Firewall
  - The Open University: "Firewalls - An Overview"Page

    Read this page.

  - Jeff Tyson's "How Firewalls Work"URL

    Read this page.

- 6.3: Host-Based IDS vs. Network-Based IDS

- Ricky M. Magalhaes' "Host-Based IDS vs. Network-Based IDS"URL

  Read this article about host-based and network-based intrusion detection systems.

- 6.4: Network Attacks and Defense
  - University of Cambridge: Ross Anderson's "Network Attack and Defense"URL

    Read this chapter. While you read, try to explain various attacks, the skills that are needed for carrying out these attacks, and how to defend your system against these attacks.

Unit 7: Physical Security

Physical security is the protection of personnel, hardware, programs, networks, and data from physical circumstances and events that could cause serious losses or damage. This unit explains the importance of physical security as a significant item in providing overall security. Without physical security, all other sophisticated techniques can become useless. Specifically, we will study the NASA physical security program, which is a result of extensive research on physical security. We will also look into different types of attacks that are possible in the context of physical security.

Completing this unit should take you approximately 7 hours.

- Unit 7 Learning OutcomesPage

- 7.1: Physical Security
  - University of Cambridge: Ross Anderson's "Monitoring Systems"URL

    Read this chapter.

  - University of Cambridge: Ross Anderson's "Physical Protection"URL

    Read this chapter. After reading these chapters, explain the relationship between threat models and attacks. Take the communication examples in the chapters and try to explain different attacks based on different assumption of threat models.

- 7.2: NASA's Physical Security Program
  - National Aeronautics and Space Administration: "Physical Security Program"URL

    Read this page about NASA's physical security program. Physical security in IT context requires most of the ideas discussed here, even though they were developed in the context of NASA's requirements.

- 7.3: Types of Attacks
    - Sarah Granger's "Social Engineering Fundamentals, Part I: Hacker Tactics"URL

      Read this page. After you read, explain how the attacker identifies the target/victim and how to carry out social engineering via various approaches (phones, online chatting, Dumpster diving, reverse engineering etc.).

Unit 8: Malicious Software and Software Security

In this unit, we will explore malicious software, also known as malware. We will also study common software security issues such as buffer overflow, used by several malware to exploit systems' vulnerability. At the end of the unit, we will learn how to use these malware and other security attacks to carry out information warfare.

Completing this unit should take you approximately 8 hours.

- Unit 8 Learning OutcomesPage

- 8.1: Malicious Web
    - University of Washington: Giovanni Vigna's "From Badware to Malware: Taming the Malicious Web"Page

      Watch this video to learn about how malicious actors leverage legitimate websites for the delivery of attacks that target vulnerabilities in client-side software.

- 8.2: Internet Security Issues
    - Talks at Google: "Vint Cerf"Page

      Watch this video to learn about security issues on the Internet, and what could have been done differently had we realized this was going to be the global information exchange infrastructure of the 21st century.

- 8.3: Types of Internet Security Issues
    - Carnegie Mellon University: "Denial of Service"URL

      Read this page. While you read, try to explain the modes of DoS attacks, such as consumption of scarce resources, configuration information alternation, and physical destruction. For DDoS attacks, describe the tools that are used for DDoS, why the DDoS attacks are possible, and the protocol vulnerabilities that are used in DDoS attacks.

    - Bennett Todd's "Distributed Denial of Service Attacks"URL

Read this page.

- 8.4: Secure Coding
  - Carnegie Mellon University: Robert Seacord's "Top Ten Secure Coding Practices"URL

    Read this page. After you read, describe the top 10 best practices for secure coding and describe the principles for secure coding (e.g., separation of duties, least privilege).

  - Open Web Application Security Project: "Secure Coding Principles"URL

    Read this page.

- 8.5: Electronic and Information Warfare
  -  University of Cambridge: Ross Anderson's "Electronic and Information Warfare"URL

    Read this chapter. After you read, describe the different attacks on communication systems and how one could use these attacks to carry out information warfare (in particular, based on the interaction between civil and military uses).

Unit 9: Security Risk Management

In this unit, we will explore risk management, which is the process of identifying vulnerabilities in an organization's information systems and taking appropriate steps to ensure confidentiality, integrity, and availability of various components of the information systems. Risk assessment is an essential element of risk management, and we will identify the steps of the risk assessment process using case studies for four different types of enterprises.

Completing this unit should take you approximately 13 hours.

- Unit 9 Learning OutcomesPage

- 9.1: How Much Security Do You Really Need?
  - Open Web Application Security Project: "How Much Security Do You Really Need?"URL

    Read this page to learn about the basics of risk assessment.

- 9.2: Risk Management

- Purdue University: Jack Jones' "Shifting Focus: Aligning Security with Risk Management"URL

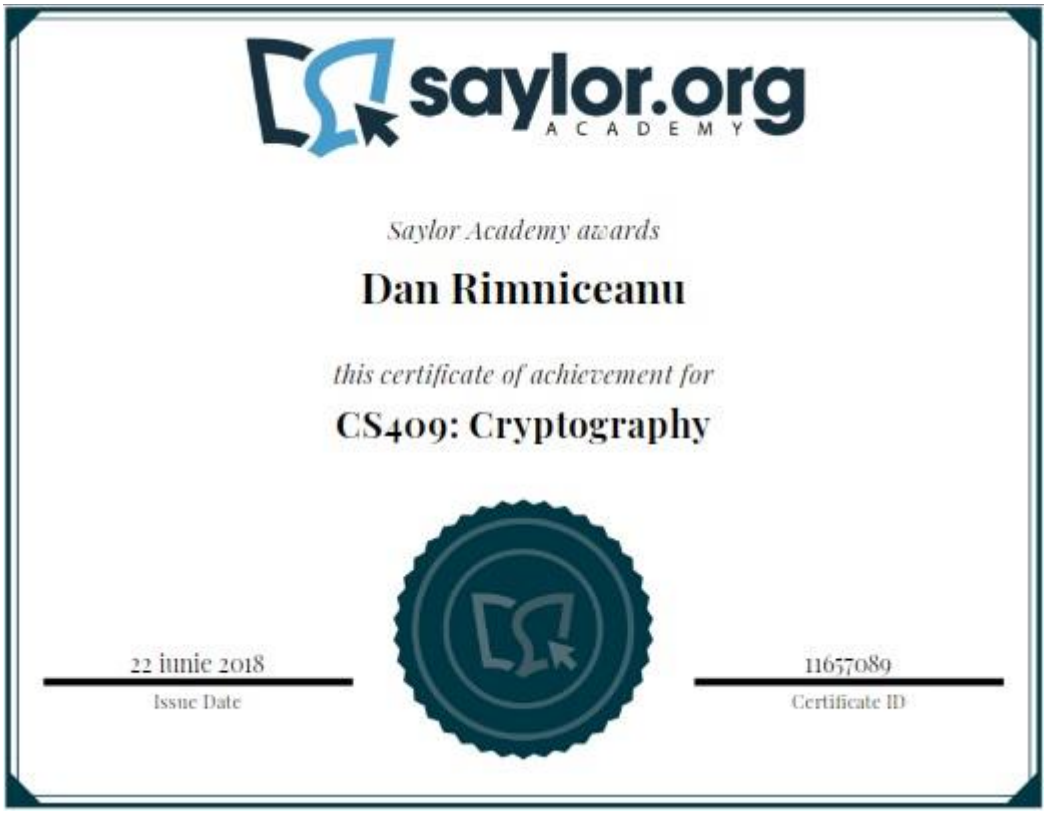
  Watch this video about security and the risk management process.

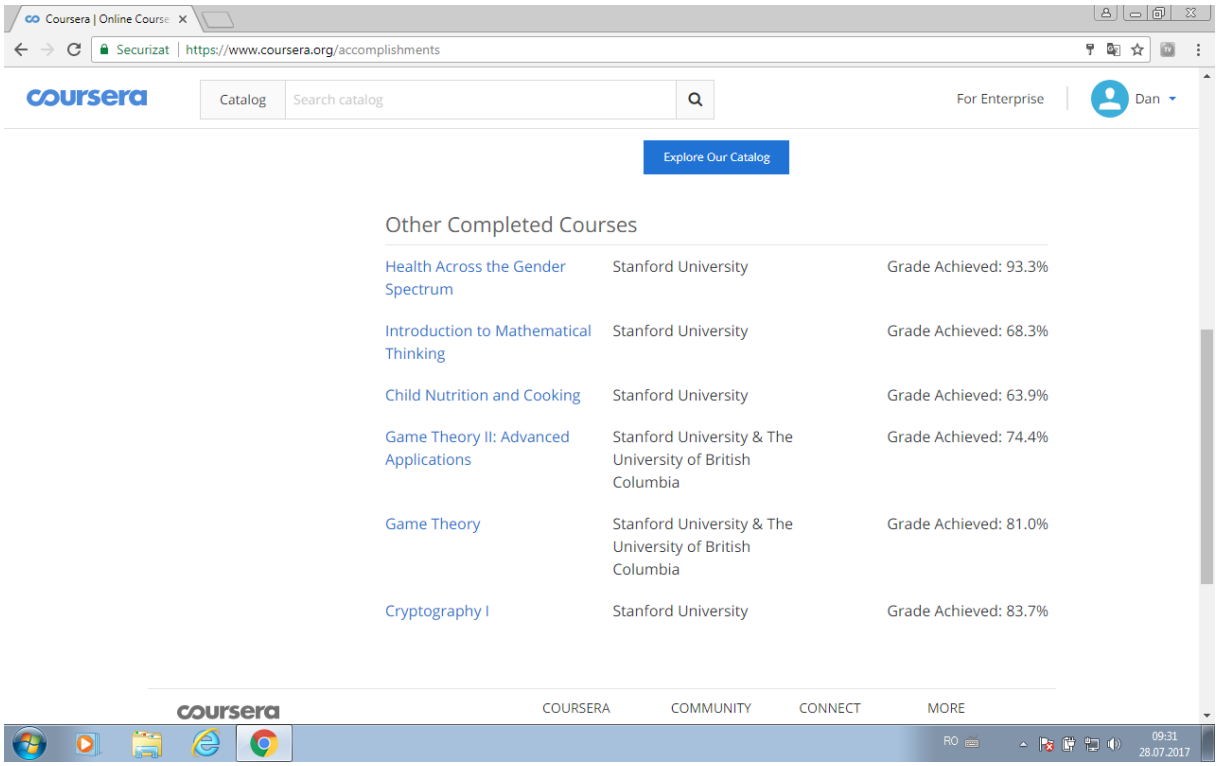
- 9.3: Information Security Risk Assessment Case Studies
  - U.S. Government Accountability Office: "Information Security Practices of Leading Organizations"File

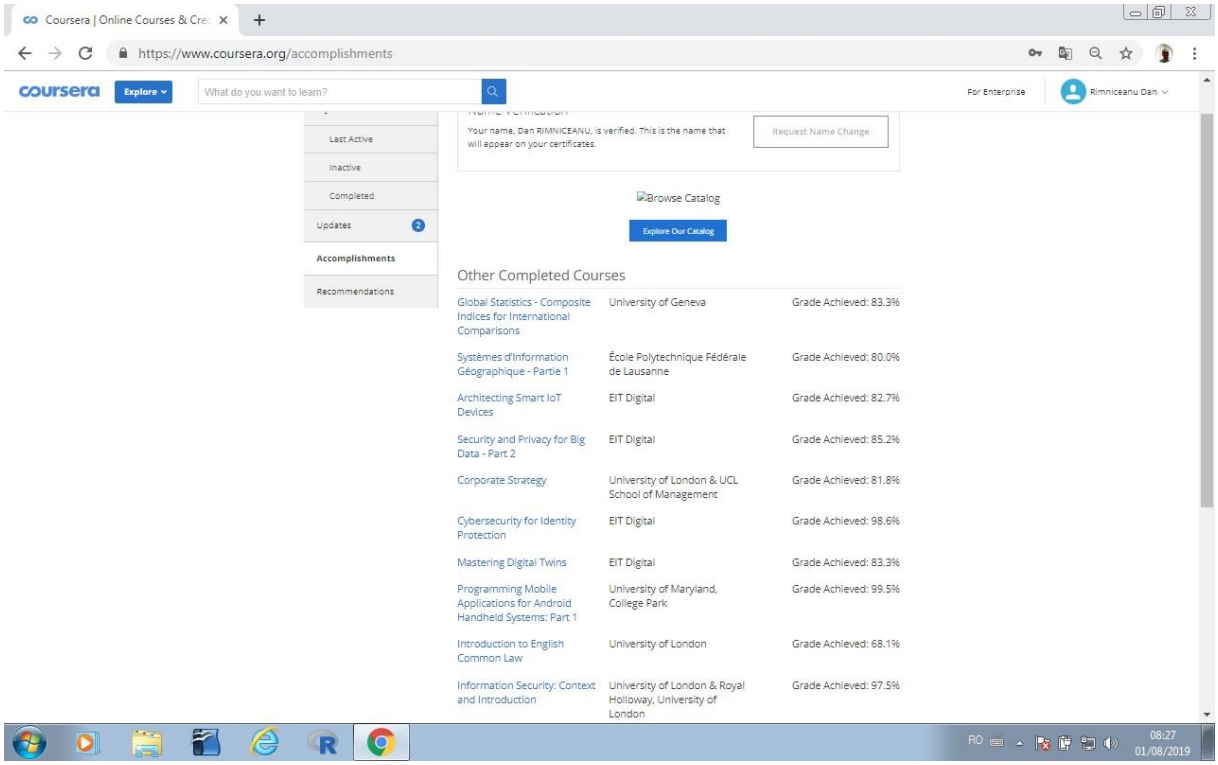
    Read the introduction to this report. After you read, describe the recommended process for risk assessment including the different roles involved. Then, read each of the case studies. As you read, try to map these two case studies to the risk assessment processes in the introduction.

- 9.4: Risk Assessment in Practice
  - Microsoft Security Assessment ToolURL

    Download and install this program. Use some simple cases to carry out a business risk profile assessment and defense in depth assessment.




Coursera | Online Course
Securizat | https://www.coursera.org/accomplishments
coursera
Catalog
Search catalog
For Enterprise
Dan
Explore Our Catalog
Other Completed Courses
Health Across the Gender Spectrum
Stanford University
Grade Achieved: 93.3%
Introduction to Mathematical Thinking
Stanford University
Grade Achieved: 68.3%
Child Nutrition and Cooking
Stanford University
Grade Achieved: 63.9%
Game Theory II: Advanced Applications
Stanford University & The University of British Columbia
Grade Achieved: 74.4%
Game Theory
Stanford University & The University of British Columbia
Grade Achieved: 81.0%
Cryptography I
Stanford University
Grade Achieved: 83.7%
coursera
COURSERA
COMMUNITY
CONNECT
MORE
RO
09:31
28.07.2017


Coursera | Online Courses & Cre
https://www.coursera.org/accomplishments
coursera
Explore
What do you want to learn?
For Enterprise
Rimniceanu Dan
Last Active
Inactive
Completed
Updates
Accomplishments
Recommendations
Your name, Dan RIMNICEANU, is verified. This is the name that will appear on your certificates.
Request Name Change
Browse Catalog
Explore Our Catalog
Other Completed Courses
Global Statistics - Composite Indices for International Comparisons
University of Geneva
Grade Achieved: 83.3%
Systèmes d'Information Géographique - Partie 1
École Polytechnique Fédérale de Lausanne
Grade Achieved: 80.0%
Architecting Smart IoT Devices
EIT Digital
Grade Achieved: 82.7%
Security and Privacy for Big Data - Part 2
EIT Digital
Grade Achieved: 85.2%
Corporate Strategy
University of London & UCL School of Management
Grade Achieved: 81.8%
Cybersecurity for Identity Protection
EIT Digital
Grade Achieved: 98.6%
Mastering Digital Twins
EIT Digital
Grade Achieved: 83.3%
Programming Mobile Applications for Android Handheld Systems: Part 1
University of Maryland, College Park
Grade Achieved: 99.5%
Introduction to English Common Law
University of London
Grade Achieved: 68.1%
Information Security: Context and Introduction
University of London & Royal Holloway, University of London
Grade Achieved: 97.5%
RO
08:27
01/08/2019

CS409: Cryptography

Course Introduction

Cryptography is essentially the science of writing in secret code. In data and telecommunications, cryptography has specific security requirements, such as authentication, privacy or confidentiality, integrity, and non-repudiation. To meet these security requirements, we employ secret key (or symmetric) cryptography, public-key (or asymmetric) cryptography, and hash functions.

In the first part of the course, we will review a number of different ciphers that were used before World War II. These ciphers would be easily broken nowadays, since cryptography has advanced quickly over the past couple of decades with the advent of modern computers. We will cover block cipher algorithms and describe the advanced encryption standard for a symmetric-key encryption adopted by the U.S. government. We will also learn about the important MD5 and SHA-1 hash functions as well as the message authentication code.

This course will focus on public key cryptography, which is best exemplified by the RSA algorithm (named after the algorithm inventors Rivest, Shamir, and Adleman). The RSA algorithm is considered particularly strong due to the fact that it relies on prime factorization, a computationally difficult process. We will take a careful look at this algorithm in this course. We will also learn about elliptic curves, another important mathematical function in cryptography, as well as the Diffie-Hellman key exchange and the elliptic curve discrete logarithm problem.

In the final part of the course, we will cover key exchange methods, study signature schemes, and provide an overview and discussion of public key infrastructure.

Unit 1: Introduction to Cryptography

This unit provides an overview of cryptography, the study of information-hiding and verification. Cryptography ensures the confidentiality/privacy, message integrity, authentication, and non-repudiation of information. There are two basic types of ciphers used: the symmetric key cipher, which uses the same key for the same message, and the asymmetric key cipher, which uses different keys for encoding and decoding the same message.

This unit will also go over the basics of information theory so that students can get a feel for message encoding before addressing various classical ciphers, which can now be easily cryptanalyzed and broken. Lastly, we will take a look at the methods and techniques used to cryptanalyze any algorithm that enciphers text.

Completing this unit should take you approximately 16 hours.

- Unit 1 Learning OutcomesPage

- 1.1: Introduction
  - *Cryptography*: "Introduction"URL

    Read this general introduction to cryptography. Learn the terms in bold font and familiarize yourself with the common goals and forms of cryptography.

- 1.2: History of Cryptography
  - *Cryptography*: "History"URL

    Read this article to familiarize yourself with cryptography's history.

- 1.3: Basics of Information Theory
  - Carnegie Mellon University: David Touretzky's "Basics of Information Theory"URL

    Read this page to learn about information theory and get a feel for how messages can be encoded. Make sure you do the exercise listed on the page under "Variable Length Codes" section.

- 1.4: Basic Cryptanalysis
  - Practical Cryptography: "Cryptanalysis"URL

    Read this page to learn about the basic cryptanalysis techniques.

  - Practical Cryptography: "Monogram, Bigram, and Trigram Frequency Counts"URL

    Read this page to learn about various cryptanalysis techniques.

- 1.5: Cryptography, Politics, and Policy
  -  Princeton University: Edward Felten's "Alice and Bob Go to Washington: A Cryptographic Theory of Politics and Policy"URL

    Watch this lecture about cryptography and politics. It discusses the choices national leaders are making about technology.

- 1.6: Stats in Code
  -  John Russell's "Frequency Counting"URL

    If you do not have java compiler on your computer, install Java on your computer via www.java.com. Then, create a java language program that performs a

frequency counting. You can see how to compile a java code via the java tutorials provided here. One possible solution can be found via the link above, under the Frequency Counting section. Study the solution code only after you have solved the problem or spent a substantial amount of time working on it.

Unit 2: Classical Cryptography

In this unit, we will learn to describe and analyze the following classical ciphers: ADFGVX, Affine, Beaufort, Bifid, Caesar, Columnar Transposition, Four-Square, Hill, Playfair, Polybius Square, Rail-fence, Simple Substitution, Straddle Checkerboard, Vigenere, Autokey, Enigma, and Lorenz ciphers.nz ciphers. These ciphers are intuitively easy to understand and seem to encrypt the message well, but they have many shortcomings, which we will discuss as we work through this unit. By studying these classical ciphers, you will learn to avoid poor cipher design.

Completing this unit should take you approximately 25 hours.

- Unit 2 Learning OutcomesPage

- 2.1: Classical Ciphers and Their Cryptanalysis
  - Practical Cryptography: "ADFGVX Cipher"URL

    Read this page, which discusses the ADFGVX Cipher. Learn the algorithm, go through the JavaScript example, and read through the cryptanalysis to learn how you would break this cipher.

  - Practical Cryptography: "Affine Cipher"URL

    Read this page, which discusses the Affine Cipher. Learn the algorithm, go through the JavaScript example, and read through the cryptanalysis.

  - Practical Cryptography: "Beaufort Cipher"URL

    Read this page, which discusses the Beaufort Cipher. Go through the algorithm and JavaScript example.

  - Practical Cryptography: "Bifid Cipher"URL

    Read this page, which discusses the Bifid Cipher. Learn the algorithm, go through the JavaScript example, and read through the cryptanalysis.

  - Practical Cryptography: "Caesar Cipher"URL

Read this page, which discusses the Caesar Cipher. First, go through the mathematical description and JavaScript example. Then, read through the cryptanalysis to learn how to break the cipher.

- Practical Cryptography: "Columnar Transposition Cipher"URL

Read this page, which discusses the Columnar Transposition Cipher. Go through the written example as well as the JavaScript example, and then read about the cryptanalysis.

- Practical Cryptography: "Four-Square Cipher"URL

Learn the algorithm and cryptanalysis of Four-Square Cipher. Go through the Javascript example.

- Practical Cryptography: "Hill Cipher"URL

Go through the written example and JavaScript example of Hill Cipher and then read about its cryptanalysis.

- Practical Cryptography: "Playfair Cipher"URL

Read about Playfair's algorithm and cryptanalysis and then go through the JavaScript example.

- Practical Cryptography: "Polybius Square Cipher"URL

Go through both examples (written and JavaScript) of the Polybius Square Cipher, and then read about its cryptanalysis.

- Practical Cryptography: "Rail-fence Cipher"URL

Go through both examples (written and JavaScript) of the Rail-fence Cipher and then read about its cryptanalysis.

- Practical Cryptography: "Simple Substitution Cipher"URL

Go through both examples (written and JavaScript) of the Simple Substitution Cipher and then read about its cryptanalysis.

- Practical Cryptography: "Straddle Checkerboard Cipher"URL

Learn the algorithm and check out the JavaScript example.

- Practical Cryptography: "Vigenere and Gronsfeld Cipher"URL

Read the page to learn the algorithm for the Vigenere, Gronsfeld and Autokey Cipher. Then go through the JavaScript example and read the cryptanalysis.

- 2.2: Mechanical Ciphers
    - Practical Cryptography: "Enigma Cipher"URL

      Read this page, which discusses the Enigma Cipher. Go through the JavaScript example and read about mathematical description.

    - Practical Cryptography: "Lorenz Cipher"URL

      Read this page, which discusses the Lorenz Cipher and go through the Javascript example.

- 2.3: Ciphers in Code
    - John Russell's "Cryptology Programs"URL

      If you do not have java compiler on your computer, install Java on your computer via www.java.com. Then, create Caesar cipher with java. You can see how to compile a java code via the java tutorials provided here. One possible solution can be found via the link above, under the Caesar section. Study the solution code only after you have solved the problem or spent a substantial amount of time working on it.

Unit 3: Block Ciphers

In this unit, we will start with an explanation of the substitution-permutation network, which works through the series of linked mathematical operations used in block cipher algorithms. Note that substitution-permutation networks take a block of plain text and the key as inputs and then apply several alternating rounds of substitution and permutation boxes to encipher the data.

This unit also uses the complete mathematical algorithm to describe the data encryption standard before finishing with a description of the advanced encryption standard for a symmetric-key encryption adopted by the U.S. government.

Completing this unit should take you approximately 14 hours.

- Unit 3 Learning OutcomesPage

- 3.1: Substitution-Permutation Network
    - Wikipedia: "Substitution-Permutation Network"URL

      Read this article to learn about substitution-permutation networks.

- 3.2: Data Encryption Standard

- National Institute of Standards and Technology: "Data Encryption Standard"URL

  Read this article to learn about the encryption standards given by National Institute of Standards and Technology. Study the material under the "Data Encryption Algorithm" section in detail. Spend time on this to make sure you know how the encryption algorithm works.

- 3.3: Advanced Encryption Standard
  - Wikipedia: "Advanced Encryption Standard"URL

    Read this article about the advanced encryption standard.

- 3.4: Abstraction in Cryptography
  - ETH Zurich: Ueli Maurer's "Abstraction in Cryptography"URL

    Watch this lecture about layers of abstraction in cryptography.

- Unit 4: Hash Functions

  This unit will introduce the concept of "hash" and then present the important MD5 and SHA-1 hash functions. (MD5 is a widely used cryptographic hash function with a 128-bit hash value, and SHA-1 is a cryptographic hash function designed by the National Security Agency.) We will finish the unit with a look at message authentication code, sometimes called a "keyed hash function."

  Completing this unit should take you approximately 19 hours.

  - Unit 4 Learning OutcomesPage
- 4.1: Cryptographic Hash
  - Steve Friedl's "Illustrated Guide to Cryptographic Hashes"URL

    Read this page about cryptographic hashes. Understand what hash is, how hash works, how to use it with UNIX, and issues related to collisions.

- 4.2: Cryptographic Hash Functions
  - 4.2.1: MD5
    - Wikipedia: "MD5"URL

      Read about the MD5, which is a widely used cryptographic hash function. Make sure you understand the MD5 algorithm.

  - 4.2.2: SHA-1
    - Wikipedia: "SHA-1"URL

Read about the cryptographic hash function designed by the National Security Agency. Make sure you understand SHA-1 pseudo code.

- 4.2.3: Message Authentication Code
  - Wikipedia: "Message Authentication Code"URL

    Read the linked article about message authentication code. Make sure you understand the examples in the article.

- 4.3: Cryptographic Hashing Function in Code
  - Open Web Application Security Project: "Cryptographic Hashing Function"URL

    Create a Java language program that runs cryptographic hashing function. One possible solution can be found in this article. Study the solution code only after you have solved the problem or spent a substantial amount of time working on it.

Skip Other students also took...

Other students also took...

Unit 5: The RSA Cryptosystem and Factoring Integers

In this unit, we will learn the basic idea behind public key cryptography and explain in detail RSA as the most important example of public key cryptography. Next, we will discuss the algorithms used to determine whether an input number is prime. As noted earlier, these algorithms are important in public key cryptography because encryption depends on the factorization of prime numbers. This unit will present the mathematical background you need in order to understand these algorithms and in turn get a better picture of public key cryptography.

Completing this unit should take you approximately 23 hours.

- Unit 5 Learning OutcomesPage

- 5.1: Introduction
  - Andrew Ross' "Public Key Cryptography"URL

    Read this introduction to public key cryptography.

- 5.1.1: Example of Public Key Cryptography
  - Andrew Ross' "RSA"URL

    Read the linked page above to learn about RSA. Take for granted the Chinese Remainder theorem, which is explained later.

- 5.1.2: Primality Testing
    - Wikipedia: "Primality Test"URL

      Read this article on primality testing, which is crucial to the security of public-key cryptography. Make sure you understand the naive tests, probabilistic tests, and fast deterministic tests.

- 5.2: Math Background
    - 5.2.1: Euclid's Algorithm
        - Robert Milson's "Euclid's Algorithm"URL

          Read this page about Euclid's algorithm. Work through the given example.

    - 5.2.2: Chinese Remainder Theorem
        -  Chi Woo's "Chinese Remainder Theorem"URL

          Read this page to learn how the Chinese Remainder theorem works.

    - 5.2.3: Legendre Symbol
        - Alvaro Lozano Robledo's "Legendre Symbol"URL

          Read this definition of the Legendre symbol.

    - 5.2.4: Calculating the Jacobi Symbol
        - Christoph Bergemann's "Jacobi Symbol"URL

          Read this page.

        - Christoph Bergemann's "Calculating the Jacobi Symbol"URL

          Read this page.

    - 5.2.5: Subgroup
        - Yann Lamontagne's "Subgroup"URL

          Read this definition of a subgroup.

- 5.3: Prime Factorization Algorithms (More Math)
    - Eric Weisstein's "Prime Factorization Algorithms"URL

      Read this introduction to prime factorization.

    - 5.3.1: Integer Factorization

- John Smith's "Integer Factorization"URL

  Read this to learn about integer factorization.

- 5.3.2: The Pollard p-1 Algorithm
  - John Smith's "Pollard p-1 Algorithm"URL

    Read this to learn how to factor an integer with Pollard's p-1 algorithm.

- 5.3.3: The Pollard Rho Algorithm
  - John Smith's "Pollard Rho Algorithm"URL

    Read this to learn how to factor an integer with Pollard's Rho algorithm.

- 5.3.4: Shanks' Square Forms Factorization
  - Wikipedia: "Shanks' Square Forms Factorization"URL

    Read this article to learn about Shanks' square forms factorization. Make sure you understand the algorithm and the examples given.

- 5.3.5: The Solovay-Strassen Algorithm
  - Christoph Bergemann's "Solovay-Strassen Test"URL

    Read this to learn the how Solovay-Strassen test works.

- 5.3.6: Strong Pseudoprimes
  - Wikipedia: "Strong Pseudoprime"URL

    Read this article to learn the definitions and properties of pseudoprime numbers. Go through the examples.

- 5.3.7: Miller-Rabin Prime Test
  - Christoph Bergemann's "Miller-Rabin Prime Test"URL

    Read this to learn the how Miller-Rabin prime test works.

- 5.4: Miller-Rabin Primality Test in Code
  - LiteratePrograms: "Miller-Rabin Primality Test"URL

    Create a java language program that performs Miller-Rabin primality test. One possible solution can be found via the link above. Study the solution code only after you have solved the problem or spent a substantial amount of time working on it.

Unit 6: Elliptic Curve Cryptography

This unit will cover elliptic curve cryptography. This approach to public-key cryptography is based on the algebraic structure of elliptic curves over finite fields. This unit includes examples of elliptic curves over the field of real numbers. The next unit will explain the Diffie-Hellman key exchange as the most important example of cryptographic protocol for symmetric key exchange. In the last part of this unit, we will learn about the elliptic curve discrete logarithm problem, which is the cornerstone of much of present-day elliptic curve cryptography.

Completing this unit should take you approximately 12 hours.

- Unit 6 Learning OutcomesPage

- 6.1: Elliptic Curve Cryptography
  - Wikipedia: "Elliptic Curve Cryptography"URL

    Read this article.

- 6.2: Elliptic Curves
  - David Jao's "Elliptic Curves"URL

    Read this article to learn about elliptic curves. Make sure you understand the examples given. Depending on your mathematics background, you may need to click on the additional links explaining the terminology used.

  - 6.2.1: Diffie-Hellman Key Exchange
    - Cameron McLeman's "Diffie-Hellman Key Exchange"URL

      Read this page to learn how to exchange the keys with Diffie-Hellman key exchange technique.

  - 6.2.2: Elliptic Curve Discrete Logarithm Problem
    - Cameron McLeman's "Elliptic Curve Discrete Logarithm Problem"URL

      Read this page, which introduces elliptic curve discrete logarithm problems. Make sure you know what the problems ask you to solve.

Unit 7: Digital Signature and Entity Authentication

This unit begins with a general discussion of key exchange methods, or methods designed to distribute keys securely so that they can be later used in a cryptographic algorithm. This unit also describes the difficult problem of computing the discrete

logarithm, which is of greatly interest to cryptologists by virtue of its ElGamal signature scheme.

The unit will then cover five additional schemes (trusted certificates, private certificates, a modified Schnorr algorithm, a modified Guillou-Quisquater algorithm, and a modified Mu-Varadharajan algorithm) before ending with an overview and discussion of public key infrastructure and a lecture by James Massey.

Completing this unit should take you approximately 12 hours.

- Unit 7 Learning OutcomesPage

- 7.1: Key Exchange
  - Wikipedia: "Key Exchange"URL

    Read this article to learn how to exchange cryptographic keys between users so a cryptographic algorithm can be used.

- 7.2: Discrete Logarithm
  - Christoph Bergemann's "Discrete Logarithm"URL

    Read about the discrete logarithm problem, which is of great interest in the field of cryptography.

- 7.3: The ElGamal Signature Schemes
  - Wikipedia: "ElGamal Signature Schemes"URL

    Read this article to learn about the system parameters, key and signature generation, verification, correctness, and security of the ElGamal signature scheme.

- 7.4: Autokey Identity Schemes
  - University of Delaware: David Mills' "Autokey Identity Schemes"URL

    Read this page to learn about autokey identity schemes.

  - Wikipedia: "Public Key Infrastructure"URL

    Read this article for general overview of public key infrastructure.

- 7.5: Cryptography - Science or Magic?

- Massachusetts Institute of Technology: James Massey's "Cryptography - Science or Magic?"URL

  Watch this lecture about cryptography. Professor Massey presents a number of topics, including No-break Cryptography, No-leak Secret Sharing, No-key Cryptography, and No-watch Coin Tossing,