CSCI-SHU 101
Introduction to Computer Science

Instructor Information

- Instructors: Xianbin Gu, Promethee Spathis, Lihua Xu
- Office: S713, S738, S752
- Office hours: TBD
- Email: xg7@nyu.edu, ps147@nyu.edu, lx615@nyu.edu

Course Information

- CSCI-SHU 101
- Introduction to Computer Science
- This course introduces the core concepts of computer science, including how computers work, how data is represented, and how algorithms and programs solve problems. Students will gain experience with basic programming, data structures, operating systems, databases, and emerging topics like AI. Designed for students with no prior computing background, the course emphasizes computational thinking and hands-on problem solving.
- Prerequisite: CSCI-SHU 11 Introduction to Computer Programming
- 2 x 110’ class meeting per week
- Class room number and building: TBD

Course Overview and Goals

This course provides a broad introduction to the central ideas of computer science and how computing enables modern innovation. Students will learn how computers represent and process information, how software interacts with hardware, how data is structured and managed. The course also introduces fundamental problem-solving techniques, computational thinking, and the role of algorithms in automating solutions.

Through hands-on examples, demonstrations, and small exercises, students will explore the essential building blocks of computing—from logic and memory to the foundations of artificial intelligence. By the end of the course, students will understand how computers work at multiple levels and how core CS concepts apply to real-world systems and emerging technologies.

Key learning themes include:

- How computers encode and manipulate data
- How hardware, operating systems, and software interact
- Fundamental algorithms and data structures

- Networking, the Internet, and cloud computing
- Databases and data-driven applications
- AI and machine learning concepts

This course prepares students for further study in computer science and provides literacy in the essential principles that shape today's digital world.

Upon Completion of this Course, students will be able to:

By the end of the course, students will be able to:

- Explain core concepts in computer science including data representation, algorithms, and computer architecture.
- Apply computational thinking strategies to break down and model problems.
- Describe how software interacts with hardware through operating systems and system-level abstractions.
- Analyze basic algorithms and data structures for efficiency and appropriateness.
- Demonstrate an understanding of networking principles and how the Internet enables global connectivity.
- Identify how data is stored, queried, and managed in databases.
- Recognize the capabilities and limitations of artificial intelligence and machine learning methods.
- Evaluate computing systems by considering trade-offs in performance, scalability, and complexity.

Course Requirements

Class Participation

Active engagement in lectures and discussions is expected. Students should attend all class sessions, contribute to discussions, ask questions, support peer learning, and contribute positively to the learning environment. Participation includes in-class activities, collaborative exercises, and engagement in lab/recitation sessions.

Weekly Assignments

Short weekly assignments reinforce key concepts such as binary representation, system architecture, data structures, and networks. These may include short written responses, algorithm design exercises, or small programming tasks. One lowest assignment score will be dropped.

Labs

Weekly lab sessions provide hands-on practice applying theoretical ideas to real computing tools. Activities may include:

- Writing and running simple programs
- Visualizing how data is stored and processed

- Measuring algorithm performance
- Exploring operating system tools and network utilities
- Working with databases and AI sandbox demos

Labs are graded on completion and participation rather than correctness, promoting experimentation and skill-building.

Tests & Quizzes

Weekly take-home quizzes will check conceptual understanding throughout the semester. A midterm and final exam will assess cumulative knowledge of computing fundamentals, including terminology, problem-solving methods, and systems thinking. Exams will not require prior programming experience.

Capstone Project

In small teams, students will design and implement a small computing solution—such as a simple game, tool, or data-driven application—demonstrating an integrated understanding of hardware-software interaction, data structures, and algorithmic thinking.

Assigned Readings

Readings will come from the textbook(s) and supplemental short articles to illustrate connections between course concepts and real-world computing systems. Students are expected to complete assigned readings before class in order to contribute fully to discussions and activities.

Grading of Assignments

The grade for this course will be determined according to the following formula:

Assignments/Activities	% of Final Grade
Weekly Problem Sets	15%
Labs / Hands-on Activities	15%
Midterm Exam	20%
Final Exam	25%
Capstone Project	25%
Participation & Engagement	5%

Letter Grades

Letter grades for the entire course will be assigned as follows:

Letter Grade	Points	Percent
A	4.00	92.5% and higher
A-	3.67	90.0 – 92.49%
B+	3.33	87.5% - 89.99%
B	3.00	82.5% - 87.49%
B-	2.67	80% - 82.49%
C+	2.33	77.5% - 79.99%
C	2.00	72.5% - 77.49%
C-	1.67	70% - 72.49%
D+	1.33	67.5% - 69.99%
D	1.00	62.5% - 67.49
D-	.67	60% - 62.49%
F	.00	59.99% and lower

View Grades

You can view your grades at any time through Brightspace under the Gradebook tab. Feedback and grades will be posted promptly as assignments are evaluated. If you have questions or believe there is an error in a posted grade, please contact the instructor within one week of receiving the grade.

Course Schedule

Topics and Assignments

Week	Topic	Goals	Assignment Due
1	What Is Computer Science?	- What is Computer Science? - History and impact of computing - Computational thinking and abstraction - Problem-solving techniques	Quiz 1: Basic CS concepts & history Lab 1: Create a simple flowchart or

		- Introduction to algorithms and pseudocode	pseudocode for a real-world problem
2	Binary Systems & Data Representation	- Binary, decimal, and hexadecimal number systems - Boolean logic and logic gates - Data representation: text (ASCII, Unicode), images, and sound - Simple arithmetic operations in binary	Quiz 2: Binary and data representation Lab 2: Convert numbers between binary, decimal, hex; simulate simple logic gates
3	How Computers Work: Architecture & Components	- Overview of computer architecture - The von Neumann model - Central Processing Unit (CPU): ALU, registers, control unit - Memory hierarchy: Cache, RAM, and storage (SSD/HDD) - Introduction to instruction sets and machine language	Quiz 3: Architecture & CPU concepts Lab 3: Visualize CPU components & memory hierarchy using simulation tools
4	From Hardware to Software	- Hardware–software relationship - OS layers - compilers vs. interpreters - programming paradigms overview	Quiz 4: Hardware-software relationships Lab 4: Explore compiling vs interpreting small code snippets
5	Introduction to Data Structures	- Arrays and linked lists - Stacks and queues - Basics of searching and sorting algorithms	Quiz 5: Data structures fundamentals Lab 5: Implement simple lists, stacks, and queues in Python or pseudocode
6	Algorithmic Thinking & Efficiency	- Algorithm comparison - Big-O intuition - Modular problem-solving	Quiz 6: Lab 6: Compare runtime of different algorithms; optimize small programs

7	Object-Oriented Thinking	- Objects, classes, inheritance, - encapsulation, polymorphism - modeling real systems	Quiz 7Model entities and relationships abstractly. Compare OOP to procedural design. Lab 7: Create a UML diagram for a mini system. Identify “objects” in real-world apps.
8	Midterm Exam		
9	Operating Systems & Process Management	- OS roles - process scheduling - threads and concurrency - file I/O	Quiz 9: OS and process concepts Lab 9: Observe process scheduling & file operations in a virtual OS environment
10	Memory & Storage Management	- Stack vs. heap - memory allocation - garbage collection - file systems and persistence.	Quiz 10: Memory & storage concepts Lab 10: Write small programs to test variable scope & memory usage
11	Databases & Information Management	- What is a database - relational model - SQL concepts - data integrity	Quiz 11: Database concepts & SQL Lab 11: Create and query a small relational database
12	Data, AI & Machine Learning	- AI concepts - ML vs. traditional programming - training data and bias - ethical implications	Quiz 12: AI/ML concepts Lab 12: Explore a simple ML dataset using pre-built notebooks
13	Networking & The Internet	- Networks (LAN/WAN/Internet) - IP, DNS, HTTP	Quiz 13: Networking fundamentals

		- cybersecurity basics	Lab 13: Ping & trace routes; basic packet inspection (safe tools)
14	Emerging Topics in Computing	- Cloud, IoT, edge - quantum & neuromorphic computing (overview)	Quiz 14: Describe current trends and their computational principles. Lab 14: Lab: Cloud sandbox exploration; IoT simulation demos.
15	Capstone Project & Final Exam	- Mini-project (algorithm visualization, network simulation, or database prototype). - Peer review & reflection.	Integrate multiple CS concepts into one coherent system. Communicate computational reasoning effectively.

Tests and Quizzes

Assessment in this course includes both in-class and take-home components designed to reinforce weekly learning:

- Weekly In-Class Quizzes (30 min) Each lecture begins with a short quiz assessing understanding of the previous week's material.
- In-Class Participation Questions (5 min/question) Throughout the lecture, students will respond to 5–6 interactive questions that contribute to the participation grade.
- Weekly Take-Home Quizzes (120 min) Following each lecture, students will complete an online quiz on Brightspace before their assigned recitation session. These assignments extend and deepen the concepts introduced in class.
- Laboratory Assignments (75 min/lab) Labs are conducted during recitation sessions. Each lab is graded based on a brief report submitted on Brightspace, using short-answer and fill-in-the-blank responses to demonstrate completion and understanding.

Course Materials

Required Textbooks & Materials

- Computer Science Illuminated (7th Edition), Nell Dale, John Lewis. Jones & Bartlett Learning. 978-1284155617

Optional Textbooks & Materials

- Computer Science: An Overview 11th Edition, J. Glenn Brookshear. Addison-Wesley. 978-0132569033

Resources

- **Access your course materials**: NYU LMS - Brightspace (nyu.edu/it/brightspace)
- **Obtain 24/7 technology assistance:** IT Service Desk (NYU IT) (nyu.edu/it/servicedesk)

Course Policies

Attendance and Tardiness

Attendance to all lectures and recitations is expected. Students are allowed 2 excused absences per semester without penalty. Any additional absences may affect your participation grade. If you miss class due to illness, family emergencies, or university-approved events, please provide appropriate documentation.

Repeated absences, whether excused or unexcused, will result in a reduction in participation points as follows:

- 3 absences will lead to an F for your participation grade.
- 5 absences will lead to a 15% reduction in your final grade.
- 8 absences will lead to failure of the course.

Punctual arrival is mandatory for this class. Be on time. Please do not leave in the middle of class unless it is an emergency.

Late Assignment

One documented absence may be excused with appropriate verification. Any additional absences will result in a zero for the missed quiz(zes), unless exceptional circumstances apply and are approved in advance. Makeup quizzes will not be offered for either documented or undocumented absences.

Absence Exceptions

Observance of Religious Holidays: You may miss class for the observance of religious holidays. If you anticipate being absent because of religious observance, notify me in advance so we can create a plan for making up missed work. For more on this policy: https://www.nyu.edu/about/policies-guidelines-compliance/policies-and-guidelines/university-calendar-policy-on-religious-holidays.html

Competitions, Conferences, Presentations: You are permitted to be absent from classes to participate in competitions, conferences, and presentations, either at home or out of town, as approved by the Associate Provost for Academic Affairs. Review the Undergraduate Bulletin for the conditions you must meet to obtain approval for this kind of absence.

A student with an injury or medical condition that requires ongoing accommodations (temporary or permanent) should contact the NYU Moses Center for Student Accessibility (CSA). If an accommodation is recommended by the Moses Center, then Academic Affairs may communicate on behalf of students to advocate for excused absences/extensions. Reasonable

accommodations, considering the course objectives, student learning, and fair standards, are ultimately decided by the professor.

Students who, in the judgment of the instructor, have not substantially met the requirements of the course or who have been excessively absent are not considered to have withdrawn from the course if they remain on the roster and may be given the final grade of F.

You will be asked to leave the class if you do not comply with the rules, and that will count towards your attendance.

Electronic Devices

Mobile Devices: You may not use mobile devices (phones, smartphones, etc.) in class unless otherwise indicated. Laptops are allowed during the labs and for taking notes (WIFI OFF unless otherwise stated). iPads or tablets are allowed for taking notes (WIFI OFF unless otherwise stated).

Recording Class: To ensure the free and open discussion of ideas, you may not record classroom lectures, discussion and/or activities without my advance written permission; any such recording can be used solely for your own private use. If you have approved accommodations from the Office of Disability Resources permitting the recording of class meetings, you must present the accommodation letter to me in advance of any recording. Distribution or sale of class recordings is prohibited without the written permission of the instructor and other students who are recorded.

Instructional Technology Tools and Assistance: If you need background on specific instructional technology tools, such as NYU LMS (Brightspace), check the RITS Student Toolkit. You may also email shanghai.rits@nyu.edu for assistance.

Use of Generative AI Tools

ChatGPT, GitHub Copilot, and other AI-assisted coding tools can be useful for learning and development. However, their use must align with the course's learning objectives and ethical guidelines. This policy outlines when and how these tools may or may not be used in this course.

Permitted Uses

Students may use AI tools in the following ways:

- Debugging Assistance: You may use AI tools to suggest solutions for debugging, but you must understand and explain any corrections you apply.
- Syntax and Documentation Help: AI tools can be used to clarify syntax, library usage, and general programming concepts.
- Code Style Suggestions: AI tools may be used to suggest formatting improvements and best practices for clean, readable code.
- Brainstorming and Learning: AI can be used to explore concepts, but students must critically evaluate responses for accuracy and completeness.

Prohibited Uses

Students may NOT use AI tools in the following ways:

- Submitting AI-Generated Code as Original Work: You must write your own code and demonstrate your own problem-solving process.
- Using AI for Home Quizzes: AI assistance is strictly prohibited on any graded assessments unless explicitly permitted.
- Bypassing Learning Objectives: Over-reliance on AI-generated solutions without understanding them defeats the purpose of learning to program.
- Generating Code Without Attribution: If AI-generated code is used as a reference, you must cite the tool used and describe any modifications made.

Any misuse of AI that violates this policy will be considered academic misconduct and may result in penalties, including a reduced grade, a failing grade on the assignment, or disciplinary action under university policies.

Instructional Technology

Email Communication: I will contact you regularly via the NYU LMS (Brightspace). You should check for emails from me (reminders, logistics, updates, etc.) at the email address connected to NYU LMS (Brightspace). Please note that I will try to respond to your emails within 24 hours. Do not expect immediate responses to emails sent late at night or on the weekends.

Assignment Notification: All quizzes will be posted to our class NYU LMS (Brightspace) Site. You are responsible for looking at NYU LMS (Brightspace) after each class period to learn about the next homework assignment.

Academic Honesty/Plagiarism

Carefully read NYU Shanghai's Statement on Academic Integrity (in the Undergraduate Bulletin). Breaches of academic integrity could result in failure of an assignment, failure of the course, or other sanctions, as determined by the Academic Affairs office.

Disability Disclosure Statement

NYU Shanghai is committed to providing equal educational opportunity and participation for students with disabilities. It is NYU Shanghai's policy that no student with a qualified disability be excluded from participating in any NYU Shanghai program or activity, denied the benefits of any NYU Shanghai program or activity, or otherwise subjected to discrimination with regard to any NYU Shanghai program or activity. Any student who needs a reasonable accommodation based on a qualified disability should register with the Moses Center for Student Accessibility for assistance. Students can register online through the Moses Center and can contact the Academic Accommodations Team at shanghai.academicaccommodations@nyu.edu with questions or for assistance.

Title IX Statement

Title IX of the Education Amendments of 1972 (Title IX) prohibits discrimination on the basis of sex in educational programs. It protects victims of sexual or gender-based bullying and

harassment and survivors of gender-based violence. Protection from the discrimination on the basis of sex includes protection from being retaliated against for filing a complaint of discrimination or harassment. NYU Shanghai is committed to complying with Title IX and enforcing University policies prohibiting discrimination on the basis of sex. Shakera Turi (shakera.turi@nyu.edu), Executive Director of the Office of Equal Opportunity, serves as the University's Title IX Coordinator. The Title IX Coordinator is a resource for any questions or concerns about sex discrimination, sexual harassment, sexual violence, or sexual misconduct and is available to discuss your rights and judicial options. University policies define prohibited conduct, provide informal and formal procedures for filing a complaint and a prompt and equitable resolution of complaints.

Links to the Title IX Policy and related documents:

- Sexual Misconduct, Relationship Violence, and Stalking Policy
- Procedures for Complaints Against Students
- Procedures for Complaints Against Employees
- Resource Guide for Students
- Resource Guide for Employees

Henry and Lucy Moses Center for Student Accessibility
Telephone: 212-998-4980
Website: https://www.nyu.edu/csa
Email: mosescsa@nyu.edu