COMP 264: Introduction to Computer Systems (Section 002) Handout 1
Fall 2026 Course Information & Syllabus

Instructor: R. I. Greenberg

Mailing Address: Comp. Sci. Dept., Loyola U., 1052 W. Loyola Ave., Chicago, IL

Email: mailto:rig@cs.luc.edu **Phone:** (773)508-3782 **Web page:** https://rig.cs.luc.edu/~rig

Office Hours: In Doyle Center Room 216 : Monday and Wednesday 1:00–2:30pm.
These are guaranteed times to find me except as announced in advance or at http://bit.ly/RIGcal. You should also feel free to look for me at other times or make appointments. Email any time is generally the best way to reach me.

Departmental tutoring schedule: See http://www.luc.edu/cs/resources/tutoring.

Lectures: Monday/Wednesday/Friday 2:45–3:35pm in Mundelein 303.
Lecture notes and handouts are generally available on the course web site. The handouts are numbered sequentially starting with handout 1.

Course Objectives: This course studies the hierarchy of abstractions and implementations that constitute a modern computer system, with a particular focus on issues of interest to programmers, typically including some systems programming instruction.
Outcomes: Understanding of system issues that affect the performance, correctness, or utility of user-level programs.

Prerequisites: Prerequisites: COMP 141 and (COMP 170 or MATH/COMP 215) and (coreq or prereq of COMP 163 or MATH 201)
COMP 170 and 163 are preferred over the alternatives. (It is also helpful if you have taken COMP 150.)

Textbook: Following is a good traditional textbook but that you should not need to buy (new edition is in the works):
Randal E. Bryant and David R. O'Halloran. *Computer Systems: A Programmer's Perspective.* Pearson Prentice Hall, 3rd edition, 2015.
Lectures and notes will do a good job of covering the key ideas from this text, and there is now also a free, online text that provides similar coverage and that you could also get in print form if you wish:
Suzanne J. Matthews, Tia Newhall, and Kevin C. Webb. *Dive into Systems.* No Starch Press, 2022. Also a free online version through https://diveintosystems.org updated 8/25/2023.

Course Requirements: There will be several assignments, several in-class quizzes, two 50 minute in-class tests, and a 120 minute final exam. The weightings within the semester grade will be: Assignments: 20%, Quizzes 10%, Tests 1–2: 20% each, and Final exam 30%.

Homework: Only homework turned in by the due date is guaranteed to be graded. Any special circumstances that cause difficulty in meeting the deadlines should be brought to the attention of the instructor in advance. Homework must be handed in by the start of class on any due date, since solutions may be discussed in the same class on occasion. Homework will generally be submitted through a specified online mechanism.

Exams: The midterm exams, tentatively scheduled for week 6 and week 11, are 50 minutes long. The final exam is scheduled for Friday, December 11 at 4:15–6:15pm.

Collaboration: *No* collaboration is permitted on exams. *Collaboration* on homework is acceptable, but *copying* is not! (Safeguard your files and printouts.) You may discuss solution techniques with other students, but you must write up your solutions independently. If you obtain a solution through research, e.g., in the library or online, cite your source completely and write up the solution in your own words. Students should keep in mind the University's Academic Integrity policy (see below).

University Policies: Note the policies regarding • Academic Integrity: Open 3rd foldout at https://catalog.luc.edu/academic-standards-regulations/undergraduate, • Religious Holidays: https://www.luc.edu/academicaffairs/stories/archive/honoringstudentreligiousobservances.shtml
• Accessibility Support: See "Student Support" section at
https://www.luc.edu/media/lucedu/online/officeofonlinelearning/OOL%20Syllabus%20Update_v5.docx
• Title IX: https://www.luc.edu/equity/otherresources/resourcesforfacultystaff/syllabuslanguage

Tentative Course Outline and Approximate Weekly Schedule:

(Note the University's academic calendar at `http://www.luc.edu/academics/schedules`.)

In brackets are recommended sections of "B&O" (Randal E. Bryant and David R. O'Halloran.) and "Dive" (*Dive into Systems*; Chapters 17 and 18 only in online version). In any case, my lecture notes sometimes add something not in either source and seek to cover the key material in a reasonably self-contained way.

1. (8/24) Administrivia. Overview of big ideas.
   Computer representation of information ["B&O" 1–1.1 or "Dive" Chap. 4 introduction].
   Program translation ["B&O" 1.2–1.3 or "Dive" 1–1.1,2.9.5(middle portion headed "Compilation Steps"),17.1–17.2,17.4,17.12,17.14].
2. (8/31) Additional overview ["B&O" 1.4–1.8 or "Dive" 5.2,5.6.2,11.1,2.1].
   "B&O" Chap. 1 remainder (esp. Amdahl's Law) ["B&O" 1.9–1.10 or "Dive" 14.4.1 but noting that "Dive" uses different notation and looks at a more specific type of program enhancement through parallelization].
   Quiz 1 (9/2–4) through "Memory Hierarchy" section of notes in Administrivia/Intro unit in Sakai.
3. (9/9) Information storage/representation, boolean algebra, bit and logical operations ["B&O" 2–2.1 or "Dive" 4.1–4.2,2.2,4.7,1.5–1.5.2,1.5.4,1.2–1.3,4.6,5.3].
   Quiz 2 (9/11–9/14) on "Information Storage" unit in Sakai.
4. (9/14) Integer representation and arithmetic ["B&O" 2.2–2.3 or "Dive" 4.3–4.5].
   Quiz 3 (9/16–18) on integer representations and arithmetic.
   Floating-point representation ["B&O" 2.4.1–2.4.3 or "Dive" 4.8].
5. (9/21) "B&O" Chap. 2 practice problems (like 2.1, 3, 4, 8, 14, 16).
   Quiz 4 (9/21–23) on floating-point representation.
   Floating point operations and rounding, etc. ["B&O" 2.4.4–2.5].
6. (9/28) Review of sample exam, etc. for Exam 1.
   Exam 1 (9/30) on integer and floating-point representations and Amdahl's Law.
   "Machine-Level Programs Part 1": introduction ["B&O" 3–3.2 or "Dive" 6].
7. (10/7) "Machine-Level Programs Part 1": data manipulation ["B&O" 3.3–3.5 or "Dive" 1.4,2.3,7–7.3].
   Quiz 5 (10/7–9) on "Machine-Level Programs Part 1".
   "B&O" Chap. 3 practice problems like 3.1, 3.6, and 3.8.
8. (10/12) "Machine-Level Programs Part 2": control ["B&O" 3.6 or "Dive" 7.4 and 2.9.1 in the middle portion headed "Switch statements"].
   "Machine-Level Programs Part 3": procedures ["B&O" 3.7 or "Dive" 7.5–7.6].
9. (10/19) "Machine-Level Programs Part 3": arrays ["B&O" 3.8 or "Dive" 1.5,2.4–2.5,2.9.4,7.7–7.8].
   "B&O" Chap. 3 practice problems like 3.36 and 3.37.
   "Machine-Level Programs Part 4": heterogeneous data structures and alignment ["B&O" 3.9 or "Dive" 1.6,2.7,7.9].
   "B&O" Chap. 3 practice problems like 3.44.
   Quiz 6 (10/23–26) on "Machine-Level Programs" Parts 2–4 as covered so far.
10. (10/26) "Machine-Level Programs Part 4" pointers, GDB, buffer overflow ["B&O" 3.10,3.12 or "Dive" 3.1,7.10,10].
    More on working with pointers, especially linked lists ["Dive" 2.9.2].
    More on memory allocation and memory bugs. [Selected content from 9.9–9.12; reading most recommended is 9.9 through 9.9.2 and 9.11. Also related is "Dive" 3.3]
11. (11/2) Review for Exam 2.
    Exam 2 (11/4) on Machine-Level Programs.
    Program performance optimization: Intro ["B&O" 5.1–5.2 or "Dive" 12–12.2].
12. (11/9) Example program and optimizations ["B&O" 5.3–5.6].
    Quiz 7 (11/11–13) on optimization so far.
    Instruction scheduling and pipelining ["B&O" 5.7 or "Dive" 5.6–5.9].
13. (11/16) More on performance optimization ["B&O" 5.8–5.12].
    Storage technologies ["B&O" 6.1 or "Dive" 11.1–2].
    Locality, memory hierarchy, cache memories ["B&O" 6.2–6.4 or "Dive" 11.3–11.4].
    Quiz 8 (11/20–11/23) on optimization remainder and memory hierarchy so far.
14. (11/23) Cache memories continued ["B&O" 6.2–6.4 or "Dive" 11.3–11.4 and 11.7].
15. (11/30) Cache-friendly code ["B&O" 6.5 or "Dive" 11.3.1].
    "B&O" Chap. 6 practice problems (like 6.9, 6.16, 6.38, 6.39).
    If time, Optimization addendum, esp. profiling example. ["B&O" 5.13–15 or "Dive" 12.1] and/or discussion of virtual memory ["B&O" 9.1–9.6 or "Dive" 13.3].