



CS 150

Introduction to Computer Science 1

MWF 1:00 - 2:00 pm

Professor: Shereen Khoja
shereen@pacificu.edu

Summary of Course



- ❖ CS150 is a first course in computing and programming fundamentals.
- ❖ The goal of this course is to introduce you to problem solving through programming a computer. No previous computer experience of any type is required, but a deep interest in using one is.
- ❖ In this course, you will learn to program in C++. You will learn the basics of data types, selection structures, repetition structures, functions and arrays.
- ❖ By the end of this course you should be able to write a program to do anything you want, given enough time and patience.



Important Information

- ❖ **Co-requisite:** Math125 Precalculus
- ❖ **Professor:** Shereen Khoja
- ❖ **Email:** shereen@pacificu.edu
- ❖ **Office:** Strain 203C
- ❖ **Phone:** 352-2008
- ❖ **Office hours:**
 - Monday 2:00 pm - 3:00 pm
 - Wednesday 11:00 am - 12:00 pm
 - Friday 10:00 am - 11:00 am
- ❖ **Text:** Problem Solving, Abstraction, and Design Using C++ (3rd Edition) by Friedman & Koffman

Grading



❖ Grading (Lecture only):

- 7 programming assignments 30%
- 3 Exams30%
- 1 Final20%
- Unscheduled quizzes.....5%
- **TOTAL****85%**

❖ The lecture portion of your grade is 85% of your grade for this class. The lab portion accounts for the remaining 15%.



Grading (cont.)

❖ Program Grading:

- 70% successful execution
- 20% acceptable structure and style
- 10% documentation

❖ Percent Breakdown:

- | | | | | |
|----------|----|-------|----|----------|
| ➤ 92-100 | A | 90-92 | A- | |
| ➤ 88-90 | B+ | 82-88 | B | 80-82 B- |
| ➤ 78-80 | C+ | 72-78 | C | 70-72 C- |
| ➤ 68-70 | D+ | 60-68 | D | |
| ➤ 0-60 | F | | | |

Course Schedule



- ❖ Chapter 1 Introduction (Week 1)
- ❖ Chapter 2 Overview of C++ (Weeks 1 & 2)
- ❖ Chapter 4 Selection Structures (Weeks 3 & 4)
- ❖ Chapter 5 Repetition and Loop Statements (Weeks 4, 5, 6, 7 & 8)
- ❖ Chapter 3 Top-Down Design with Functions & Classes (Week 8)
- ❖ Chapter 6 Modular Programming (Weeks 8, 9, & 10)
- ❖ Chapter 7 Simple Data Types (Weeks 11 & 12)
- ❖ Chapter 8 Streams and Files (Weeks 11 & 12)
- ❖ Chapter 9 Arrays and Structures (Weeks 13 & 14)
- ❖ Chapter 12 Modeling Data with Arrays, Structs, and Classes (Weeks 13 & 14)

Exams



❖ Tentative dates for Midterms:

- Midterm 1 Friday, September 19 (Week 4)
- Midterm 2 Friday, October 17 (Week 8)
- Midterm 3 Friday, November 14 (Week 12)

❖ Date of Final:

- Monday, December 8, 8:30 AM - 11:00 AM

University Policies



- ❖ Attendance at every class is critical to your success in this class. I expect you to be on time and ready to go once it's 1:00 pm. Any missed lecture is your responsibility to make up. Do not expect me to repeat missed material in class.
- ❖ Computers are not to be used during lecture time. Once I begin lecturing, the keyboards should be pushed under the desk and the monitors should be ignored. Failure to do so will result in you being excused from class.
- ❖ Cell phones are to be turned off and put away during class. Any cell phone that rings during class will be confiscated. Leaving during the middle of class to answer a page/call is extremely rude, too.

Policies (cont.)



- ❖ Assignments are to be turned in by the beginning of class on the day they are due. An assignment may be turned in at the beginning of the next class with a penalty of 20% of the assignment points. Any assignment turned in later will not be graded and will be recorded as a zero.
- ❖ Make sure to test your program before you turn it in. You may turn in your program only once.
- ❖ A program that does not successfully compile or produces no output loses 70%.
- ❖ No early or late exams/finals will be given under any circumstance. Do not make early Thanksgiving or end of term arrangements.

Policies (cont.)



- ❖ The cheating policy is defined in Pacific Stuff & the Pacific Catalog as well as the Academic Policy that each of you signed. Be sure you read this policy carefully. Every piece of code written for CS150 is to be an original design and an original implementation. The Web, textbooks and any other references are simply references for you. This means that copying code from any source is prohibited. Further, source code is not to exchange hands in any form or by any medium except when sending your solutions to the instructor. It is OK to share high level ideas during your design phase, help someone fix a bug occasionally, and share information dealing with the computer system (compiling, using editors, etc.) that does not involve code writing.

Policies (cont.)



- ❖ All code in any form generated from this course becomes the intellectual property of Pacific University. You may not share this code with anyone at any time (including after this course is over) without obtaining written permission from Pacific University.

Policies (cont.)



- ❖ Computer failure, software failure, and lack of computer access are not accepted as excuses for late programs; therefore, start work on the programs as soon as they are assigned, don't put them off until the last minute. Make sure to keep backup copies of your assignment. Corrupting or accidentally deleting your programs is also not an acceptable excuse for late programs.
- ❖ The instructor reserves the right to raise or lower a student's grade based on class participation and attendance.



Important Dates

- ❖ September 1 Labor Day Holiday
- ❖ September 8 Last day to add courses, Last day to drop courses with no record
- ❖ October 3 No Classes (Arts & Sciences)
- ❖ November 3 Last day to withdraw from courses
- ❖ November 26(noon)-28 Thanksgiving break
- ❖ December 2 Last day of classes
- ❖ December 3 Reading Day
- ❖ December 8 (Monday) 8:30 am – 11:00 am, Final Exam

Basics



- ❖ You will be able to use any computer on campus (including your own) for homework
- ❖ You should already have a PUnetID and password for your Pacific email account
- ❖ Make sure to test this out
- ❖ Go to the TIC if it doesn't work

Goals



- ❖ Understand the basics of computers
- ❖ Learn problem solving skills
- ❖ Learn how to program in C++
- ❖ Learn how to use some program development tools (in lab)

Expectations



- ❖ Attend class
 - Take notes
 - Participate
 - Ask questions

- ❖ Keep up outside of class
 - Read the text book
 - Start assignments early
 - Work on your own
 - Follow directions

What is a computer?



- ❖ Several definitions
- ❖ PC's, Macs, embedded systems
- ❖ Composed of:
 - hardware
 - software

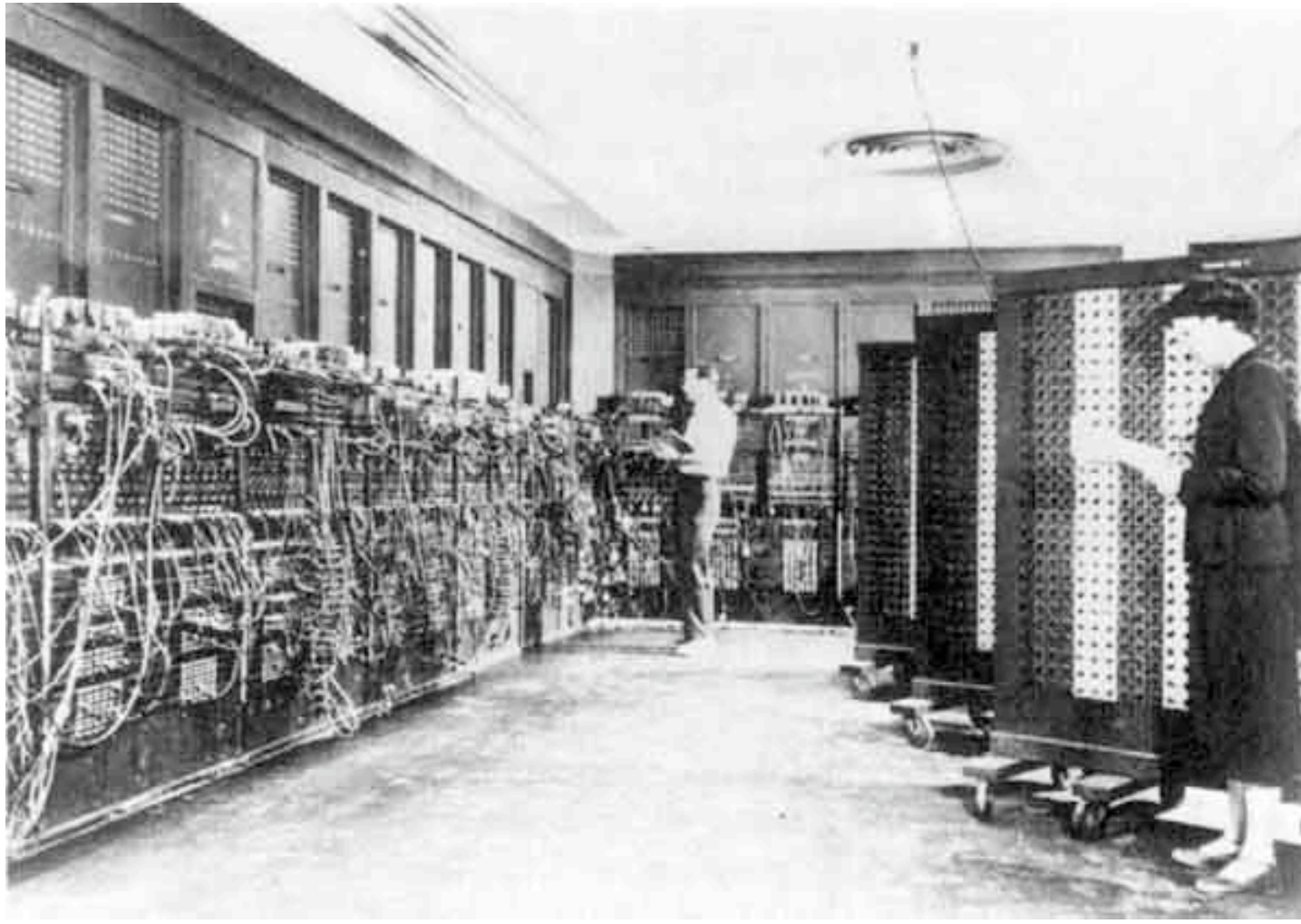
History



- ❖ First electronic digital computer
 - Late 1930's at Iowa State
 - Dr. John Atanasoff and Clifford Berry
 - Mathematical computations for nuclear physics

- ❖ First large-scale, general purpose computer
 - ENIAC in 1946 at U. Penn. for US Army
 - J. Presper Eckert and John Mauchley
 - Weighed 30 tons and occupied 1500 sq. ft.
 - Cost \$500,000 to develop and build
 - Used for calculating ballistics tables, predicting weather and making atomic energy calculations

Picture of ENIAC



Von Neumann Architecture



- ❖ Dr. John von Neumann proposed the concept of a stored-program computer
- ❖ In ENIAC data is stored in memory, so why not a program
- ❖ The von Neumann architecture is the basis of the digital computers we know today

Today



- ❖ Most of us use microcomputers
 - First developed in 70's
 - Small processor
 - Mac's and PC's are examples
- ❖ Large companies use mainframes
- ❖ Researchers, government, large companies use supercomputers

Hardware



- ❖ Main Memory
- ❖ Secondary Storage
- ❖ CPU (central processing unit)
- ❖ Input Devices
- ❖ Output Devices
- ❖ Network Connections

Memory



Address	Contents
0	-27.2
1	354
2	0.05
3	-26
4	H
5	400
6	RTV 001
7	ADD 003
8	STO 005
9	X
10	1005

- Memory is a sequence of storage cells
- Each memory cell has unique address
- Contents can be data or instruction
- Memory cell composed of bytes
- Bytes are groups of bits (8 usually)
- Bits are 0 or 1
- Everything stored as strings of 0s & 1s

Memory



❖ Main Memory

- RAM
- ROM

❖ Secondary Storage

- Hard disks
- Floppy disks
- CD ROMs

CPU



- ❖ Its job is to coordinate all operations
- ❖ Perform arithmetic operations
- ❖ Today, CPU's are integrated circuits

Software



- ❖ Operating System
- ❖ Application Software
- ❖ Programming Language/Compiler

Question



- ❖ Can computers think?
- ❖ No
- ❖ Computers are dumb
- ❖ Computers need a list of instructions to perform operations
- ❖ These instructions are *programs*

Programming Language



- ❖ Machine Language
 - 0010 0000 0000 0100
 - Each computer has its own unique machine language
 - Programming different ones is a pain
- ❖ High Level languages make programming easier
 - Compiler translates program into machine language
- ❖ Examples of languages?

Summary



- ❖ Today we have looked at:
 - The history of computers
 - The hardware of computers
 - The software of computers

- ❖ Next time we will:
 - Learn about the software development cycle
 - Start coding

- ❖ Completed sections 1.1 - 1.5 from the book