Chapter 1
Introduction to Computers and Programming

- Reading: Chapter 1 (1.1 to 1.5)
- Good Problems to Work: p. 13 [1.11, 1.17] p. 18 [1.18, 1.21, 1.22, 1.23]
Chris Lane
Mathematics and Computer Science

- Web Page: http://zeus.cs.pacificu.edu/lanec/
- Email: lanec@pacificu.edu
- Office: Price 211
- Office Hours:
  - Monday, Friday 10:30am-11:30pm and Tuesday 10-11:30am or by appointment

Fall 2016  CS150 - Intro to CS I  3

What is CS 150

- CS 150 is a programming course
- You will learn
  - Syntax (Grammar)
    - The mechanics of writing a C++ program
  - Design
    - Logical reasoning
    - How do I solve this problem with a program?
    - How do I break this problem into smaller, solvable tasks?
- No previous skills required!
How to succeed in CS150

- Don’t miss class.
  - Take notes
  - Bring book and notes to lab, lecture, and office hours

- Try and read ahead
  - Bring questions to class!

- Start programming assignments early
  - They take much longer than you think

- Do as much on your own as possible. If you get stuck, come see me or one of the TAs

How to succeed in CS150

- Read the assignments carefully and follow all directions

- See me as soon as possible about any questions!

- Don’t forget that you are at a small school!
How to send an effective email

To: lanec@pacificu.edu
From: smit1234@pacificu.edu
Subject: CS150 - exam question

Hello Prof. Lane,

I’m studying for the exam and I ran across switch statements in the reading. Since we did not cover switch statements in class, I was wondering if switch statements could be on the exam?

Thanks,
Lesley

First Homework Assignment

- Fill out the survey on the class web page
- Print it out
- Turn it in on Tuesday at the start of lab time
Programs and Programming Languages

- What is a program?

Programs are written in high-level languages
  - Instructions look like everyday English (sort of)
  - Each instruction can perform many machine language instructions

Compilers
  - Translate programs into machine language which is
    - zeroes and ones
    - machine dependent

Programming

- Be very specific about what you want the computer to do
- The computer follows directions precisely
- You can't just make stuff up and expect the computer to understand
- On the other hand, sometimes you don't know exactly what you want to do ... try something ... anything ... you can't hurt the computer!!!!! 😊
### C++ Programming Language

- C++ is
  - based on the C programming language
  - a high-level programming language
  - one of today's most popular programming languages
  - used extensively in industry

---

### C++ HelloWorld Program

**What is the output?**

```cpp
//****************************************************************************
// File name: hello.cpp
// Author: Sharon Smith
// Date: 08/29/2016
// Purpose: This program displays a welcome message to the user after the user enters their name
//****************************************************************************
#include <iostream>
#include <string>
using namespace std;

int main()
{
    string name;
    cout << "Type your name, then press enter " << endl;
    cin >> name;
    cout << "Hello " << name << "! " << endl;
    return EXIT_SUCCESS;
}
```
Language Elements

- **Key Words**
  - Have special meaning in C++
  - `using namespace int`

- **Programmer-Defined Identifiers**
  - Names made up by the programmer
  - Example: `employer, name`

- **Operators**
  - Perform operations
  - `* =`

- **Punctuation**
  - Used to mark the beginning and end of the program `{ }`
  - Used to separate C++ statements `;`

Syntax (Grammar)

- Rules that must be followed when constructing a program

- Controls the use of key words, programmer-defined identifiers, operators, and punctuation
Variables and Variable Definitions

- **variable** - named storage location in the computer’s memory which holds a piece of information
- **variable definition** - statement used to define one or more variables

Does the Hello World program have any variables?

Input, Processing, Output

- Input, processing, and output are three main activities performed by a program
- Assume we have three variables hoursWorked, payRate, and pay
- Input
  ```
  cin >> hoursWorked;
  cin >> payRate;
  ```
- Processing
  ```
  pay = hoursWorked * payRate;
  ```
- Output
  ```
  cout << pay;
  ```