Module 1 - Introduction to Computers, the Internet, and the Web

Outline

Module 1 - Introduction to Computers, the Internet, and the Web

Outline

1.1 What is a Computer?

- Computer
  - Device capable of
    - Performing computations
    - Making logical decisions
  - Works billions of times faster than human beings
  - Fastest *supercomputers* today
    - Perform hundreds of billions of additions per second

Secret Agent 89

- Secret agent 89 is trying to find out how many days before a certain world leader is to be assassinated. His contact tells him that this information is located in a series of post office boxes. To ensure that no one else gets the information, it is spread through 10 different boxes. His contact gives him 10 keys along with the following instructions:

1. The information in each box is written in code.
2. Open box 1 first and execute the instructions located there.
3. Continue through the rest of the boxes in sequence unless instructed to do otherwise.
4. One of the boxes is wired to explode upon opening.

Secret Agent 89

Agent 89 takes the 10 keys and proceeds to the post office, code book in hand.

- Table 1 shows the contents of the 10 post office boxes after having been decoded. Assume that you are Agent 89; begin at box 1 and go through the sequence of operations to find the number of days before the assassination attempt.

Secret Agent 89

(1) Add the number stored in the box (9) to your secret agent code number	(2) Divide the previous result by the number stored in box (10)
(3) Subtract the number stored in box (8)	(4) if the previous result is not equal to 30, go to box (7). Otherwise continue to next box
(5) Subtract 13 from the previous result	(6) HALT. You now have the answer
(7) BOMB! Too bad	(8) 20
(9) 11	(10) 2

1.1 What is a Computer? (II)

- Programs
  - Sets of instructions that process data
  - Guide computer through orderly sets of actions specified by *computer programmers*
- Hardware
  - Physical devices of computer system
- Software
  - Programs that run on computers
- Computer system
  - Comprised of various *hardware* devices
    - Keyboard
    - Screen (monitor)
    - Disks
    - Memory
    - Processing Units

1.1 What is a Computer? (III)

- Every computer divided into six units
  1. *Input unit*
     - "Receiving" section of computer
     - Obtains data from *input devices*
       - Usually a keyboard, mouse, disk or scanner
     - Places data at disposal of other units
  2. *Output unit*
     - "Shipping" section of computer
     - Puts processed info on various *output devices*
       - Screens, paper printouts, speakers
     - Makes info available outside the computer

1.1 What is a Computer? (IV)

3. Memory unit

- Rapid access, low capacity "warehouse"
- Retains information entered through input unit
- Retains info that has already been processed until can be sent to output unit
- Often called *memory*, *primary memory*, or ***r****andom* ***a****ccess* ***m****emory* (RAM)

4. ***A****rithmetic and* ***L****ogic* ***U****nit*

- "Manufacturing" section of computer
- Performs calculations (addition, subtraction, multiplication and division)
- Contains decision mechanisms and can make comparisons

1.1 What is a Computer? (V)

5. ***C****entral* ***P****rocessing* ***U****nit* (CPU)
   - "Administrative" section of computer
   - Coordinates and supervises other sections
6. *Secondary storage unit*
   - Long-term, high-capacity "warehouse"
   - Stores programs or data not currently being used by other units on *secondary storage devices* (like discs)
   - Takes longer to access than primary memory

1.2 Evolution of Operating Systems

- Batch processing
  - One job (task) at a time
  - Operating systems developed
    - Programs to make computers more convenient to use
    - Switch jobs easier
- Multiprogramming
  - “Simultaneous” jobs
  - Timesharing operating systems

1.2 Personal, Distributed and Client/Server Computing

- Personal computing
  - Computers for personal use
- Distributed computing
  - Computing performed among several computers
- Client/server computing
  - Servers offer common store of programs and data
  - Clients access programs and data from server

1.3 Types of Programming Languages

- Computer programs
  - Called software
  - Programmers write instructions that comprise software in various programming languages
- Three general types of programming languages
  - Machine languages
  - Assembly languages
  - High-level languages

1.3 Types of Programming Languages (II)

- Machine languages
  - "Natural language" of a particular computer
  - Defined by hardware design of computer
  - Generally consists of strings of numbers
  - Are *machine dependent*
  - Cumbersome for humans
    - Example: Adding overtime pay to base pay and storing the result in gross pay

      ```
      +1300042774
      +1400593419
      +1200274027
      ```
  - Slow and tedious for most programmers

1.3 Types of Programming Languages (III)

- Assembly languages
  - Programmers began using English-like abbreviations to substitute for machine languages
  - Represents elementary operations of computer
  - *Translator programs* called *assemblers* convert assembly-language to machine-language
  - Example:

```
LOAD BASEPAY
ADD OVERPAY
STORE GROSSPAY
```


1.3 Types of Programming Languages (IV)

- High-level languages
  - Developed as computer usage increased, assembly language proved inadequate and time-consuming
  - Single statements can be written to accomplish substantial tasks
  - Translator programs called *compilers*
  - Allow programmers to write instructions almost like every-day English
  - Example:

```
grossPay = basePay + overTimePay
```


1.3 Types of Programming Languages (V)

- High-level languages (II)
  - Much more desirable from programmer's standpoint
  - Specific languages include
    - C, C++, Visual Basic and Java
    - Scripting languages: JavaScript, VBScript and Perl
    - Among most powerful and widely used languages today
  - *Interpreter programs* developed to execute high-level programs without compiling
    - Popular in program development environments
  - Once program developed, compiled version made

1.4 Other High-Level Languages

- Hundreds developed, only few widely used
  - *COBOL* (**CO**mmon **B**usiness **O**riented **L**anguage)
    - Used primarily for commercial applications that require precise and efficient manipulation of large amounts of data
    - Half of all business software still programmed in COBOL
  - *Pascal*
  - *BASIC*
    - Simple language to help novices become comfortable with programming

1.5 Structured Programming

- 1960's
  - People realized that software development far more complex than imagined
  - Resulted in evolution of *structured programming*
    - Disciplined approach to writing computer programs
    - Clearer and easier to debug and modify than unstructured programs
- Pascal
  - Designed for teaching structured programming in an academic environment
  - Became preferred programming language in most universities

1.5 Structured Programming

- Ada programming language
  - Developed under sponsorship of Dept. of Defense (DOD)
  - Wanted single language to fulfill all DOD programming needs
  - Based on Pascal but different in end
  - *Multitasking* capability
    - Many activities can occur in parallel
    - Not featured in C and C++
    - Similar to Java *multithreading* technique

1.6 History of the Internet

- *ARPAnet*
  - Implemented in late 1960's by ARPA (Advanced Research Projects Agency of DOD)
  - Networked computer systems of a dozen universities and institutions with 56KB communications lines
  - Grandparent of today's Internet
  - Intended to allow computers to be shared
  - Became clear that key benefit was allowing fast communication between researchers – *electronic-mail* (*email*)

1.6 History of the Internet (II)

- ARPA's goals
  - Allow multiple users to send and receive info at same time
  - Network operated *packet switching* technique
    - Digital data sent in small packages called *packets*
    - Packets contained data, address info, error-control info and sequencing info
    - Greatly reduced transmission costs of dedicated communications lines
  - Network designed to be operated without centralized control
    - If portion of network fails, remaining portions still able to route packets

1.6 History of the Internet (III)

- ***T**ransmission **C**ontrol **P**rotocol* (TCP)
  - Name of protocols for communicating over ARPAnet
  - Ensured that messages were properly routed and that they arrived intact
- Organizations implemented own networks
  - Used both for intra-organization and communication

1.6 History of the Internet (IV)

- Huge variety of networking hardware and software appeared
  - ARPA achieved inter-communication between all platforms with development of the *IP*
    - *Internetworking Protocol*
    - Current architecture of Internet
  - Combined set of protocols called *TCP/IP*
- The Internet
  - Limited to universities and research institutions
  - Military became big user
  - Next, government decided to access Internet for commercial purposes

1.6 History of the Internet (V)

- Internet traffic grew
  - Businesses spent heavily to improve Internet
    - Better service their clients
  - Fierce competition among communications carriers and hardware and software suppliers
  - Result
    - *Bandwidth* (info carrying capacity) of Internet increased tremendously
    - Costs plummeted

1.7 Personal Computing

- IBM
  - 1981, introduced *IBM Personal Computer*
  - Made personal computing legitimate in business, industry and government organizations
  - Computers were “stand-alone” units
    - Info only shared between computers through exchange of discs
  - Machines could be linked
    - Over telephone lines
    - Over ***L****ocal* ***A****rea* ***N****etwork*s (LANs)
  - Led to *distributed computing*

1.7 Personal Computing (II)

- Computers today
  - As powerful as million dollar machines from 20 years ago
  - *Workstations*
    - Most powerful desktops today
    - Provide users with enormous capabilities
  - Information easily shared over networks
    - Networks controlled by *servers*
      - Common programs and data used by *client* computers
  - Popular operating systems
    - UNIX, OS/2, MacOS, Windows, Windows NT, Linux

1.8 History of the World Wide Web

- WWW
  - Allows computer users to locate and view multimedia-based documents
  - Introduced in 1990 by *Tim Berners-Lee*
- Internet today
  - Mixes computing and communications technologies
  - Makes information constantly and instantly available to anyone with a connection

1.9 World Wide Web Consortium (W3C)

- W3C
  - Founded in 1994 by Tim Berners-Lee
    - Devoted to developing non-proprietary and interoperable technologies for the World Wide Web and making the Web universally accessible
  - Standardization
    - W3C *Recommendations*: technologies standardized by W3C
      - include Extensible HyperText Markup Language (XHTML), Cascading Style Sheets (CSS) and the Extensible Markup Language (XML)
    - Document must pass through *Working Draft, Candidate Recommendation* and *Proposed Recommendation* phases before considered for W3C Recommendation

1.9 World Wide Web Consortium (W3C) (II)

- W3C Structure
  - 3 Hosts
    - Massachusetts Institute of Technology (MIT)
    - INRIA (Institut National de Recherche en Informatique et Automatique)
    - Keio University of Japan
  - 400 Members (including Deitel & Associates)
- W3C homepage at www.w3.org
- W3C Goals
  - User Interface Domain
  - Technology and Society Domain
  - Architecture Domain and Web Accessibility Initiatives

1.10 Hardware Trends

- Improving technologies
  - Internet community thrives on improvements of
    - Hardware
    - Software
    - Communications
  - Cost of products and services
    - Consistently dropping over the decades
  - Computer capacity and speed
    - Doubles every two years (on average)
  - Microprocessor chip
    - Laid groundwork in late 1970s and 1980s for productivity improvements of the 1990s

1.11 The Key Software Trend: Object Technology

- Costs
  - Hardware costs dropping
  - Software development costs rising
    - More sophisticated and powerful programs being developed
- *Objects*
  - Reusable software *components* that model items in the real world
  - Makes software developers more productive
  - Object-oriented programs often easier to understand, correct and modify than older types of programs

1.11 The Key Software Trend: Object Technology (II)

- *Structured Programming*
  - Led to first improvements in software technology
- Larger improvements
  - Only appeared with object-oriented programming in 1980s and 1990s
- Object technology
  - Dates to 1960s
  - C++, developed in 1980s, based on two languages
    - C – Developed to implement the UNIX OS in early 1970's
    - Simula 67 – Simulation programming language from 1967
  - C++ absorbed capabilities of C and added Simula's capabilities of creating and manipulating objects

1.11 The Key Software Trend: Object Technology (III)

- Object technology
  - Packaging scheme that helps create meaningful software units
    - Large and highly focused on particular applications areas
  - Before appeared, programming languages were focused on actions (verbs) rather then on objects (nouns)
    - Programmers would program primarily with verbs
    - Made program awkward
    - We live in a world filled with complex objects and simple actions

1.11 The Key Software Trend: Object Technology (IV)

- Object technology (continued)
  - Object-oriented programming
    - Programmers work in manner similar to how they see the world
    - More natural process
    - Significant productivity enhancements
  - Procedural programming
    - Not particularly reusable
    - Forces programmers to constantly "re-invent the wheel"
      - Wastes time and resources
  - Objects
    - Software modules
    - Kept in libraries
    - Reusable – save time and resources

1.12 JavaScript: Object-Based Scripting for the Web

- JavaScript
  - Attractive package for advancing level of programming language education
  - Object-based language
  - Supports proper software engineering techniques
  - Free for download in today's most popular Web browsers
    - Attractive to colleges
    - Bug fixes and new versions easily obtained
  - Powerful scripting language
    - Portable
    - Programs execute interpretively on client machines

1.13 Browser Portability

- Browser portability
  - Great challenge
    - Great diversity of client browsers in use
    - Many different platforms also in use
- Difficult to
  - Know capabilities and features of all browsers and platforms in use
  - Find correct mix between absolute portability, complexity and usability of features

1.14 C and C++

- History of C and C++
  - Evolved from B language (developed by Dennis Ritchie)
  - C Implemented in 1972 as contemporary of Pascal
    - C++ developed by Bjarne Stroustrup in 1980s
    - C++ Initially used in Unix, today used in virtually all new operating systems
  - Deitel books in early 90's encouraged use of C over Pascal
    - Many believed C to be too difficult
- Advantages of C++
  - Extends C programming into object orientation
  - Older C code may be integrated into C++

1.15 Java

- History of Java
  - Project Green (1991): developed C based language (later called Java) for intelligent consumer electronic devices
    - Advanced by World Wide Web explosion in 1993: potential to create Web pages with *dynamic content*
    - Java introduced in May of 1995
- Advantages of Java
  - Allows Web pages with dynamic and interactive content
  - Allows Large-scale enterprise applications
  - Enhances Web Servers
  - Provides applications for consumer devices
  - Now one of most widely implemented languages in world

1.16 Internet and World Wide Web How to Program

- Rise of electronic-commerce (e-commerce)
  - Reconstruction of modern business
- *Internet and World Wide Web How to Program*
  - Teaches programming languages, programming language principles and Internet and Web-based application technologies
  - Intended audiences
    - Introductory courses in C++, Java and Visual Basic
    - Upper-level elective programming courses
    - Corporate training programs for professional programmers

1.16 Internet and World Wide Web How to Program (II)

- Internet and World Wide Web How to Program includes treatments of 6 other popular programming languages
  - Microsoft Active Server Pages (ASP)
  - Perl and Common Gateway Interface (CGI)
  - Python and PHP
  - Java Servlets and JavaServer Pages

1.17 Dynamic HTML

- DHTML
  - Two versions
    - Microsoft
    - Netscape
  - Consists of number of technologies freely available for download
  - Used for developing high-performance, Web-based applications
    - Much of application's work performed directly on client rather then on server or Internet