Objectives Overview

- Differentiate between machine and assembly languages
- Identify and discuss the purpose of procedural programming languages, and describe the features of C and COBOL
- Identity and discuss the characteristics of these object-oriented programming languages and program development tools
- Identify the uses of other programming languages and program development tools
- Describe various ways to develop Web pages

See Page 663 for Detailed Objectives
Objectives Overview

- Identify the uses of popular multimedia authoring programs
- List the six steps in the program development life cycle
- Differentiate between structured design and object-oriented design
- Explain the basic control structures and design tools used in designing solutions to programming problems

See Page 663 for Detailed Objectives

Computer Programs and Programming Languages

- A **computer program** is a series of instructions that directs a computer to perform tasks
  - Created by a **programmer** using a **programming language**
Low-Level Languages

• **Machine language** is the first generation of programming languages
  - Only language the computer directly recognizes

```
0000:DE 5A50 36AA 015AC
0001:DE 47F0 2100 081D2
0010:77 1B77 FF00 01050
0010:8F 1C47 1F00 01004
0011:04 4F50 3006 0109E
0011:1E 0075 3006 003E 01008
0011:1F 4050 30D6 01008
0011:28 5050 3022 01054
0011:2C 58ED 30B6 01080
0011:2F 67FE 00122
0011:2F 50EO 30BA 0109C
0011:2F 1855 01054
0011:2F 5650 304E 01054
0011:2F 5850 3052 01054
0011:30 5050 305A 0105C
0011:34 58ED 301A 0109C
0011:38 67FE
```

Low-Level Languages

• **Assembly language** is the second generation of programming languages
  - Programmer writes instructions using symbolic instruction codes
  - A **source program** contains the code to be converted to machine language
Procedural Languages

• In a **procedural language**, the programmer writes instructions that tell the computer what to accomplish and how to do it
  
  – Third-generation language (3GL)

A compiler translates an entire program before executing it

An interpreter converts and executes one code statement at a time

---

Figures 13-4 – 13-5

- COMPUTE REGULAR TIME PAY
  
  Multiply Regular Time Hours by Hourly Pay Rate
  
  Giving Regular Time Pay.

- COMPUTE OVERTIME PAY
  
  If Overtime Hours > 0
  
  Compute Overtime Pay = Overtime Hours * 1.5 * Hourly Pay Rate
  
  Else
  
  Move 0 to Overtime Pay.

- COMPUTE GROSS PAY
  
  Add Regular Time Pay to Overtime Pay
  
  Giving Gross Pay.
Procedural Languages

- The **C** programming language is used to write many of today’s programs.

```
/* Compute Regular Time Pay */
rt_pay = rt_hrs * pay_rate;

/* Compute Overtime Pay */
if (ot_hrs > 0)
  ot_pay = ot_hrs * 1.5 * pay_rate;
else
  ot_pay = 0;

/* Compute Gross Pay */
gross = rt_pay + ot_pay;

/* Print Gross Pay */
printf("The gross pay is %d\n", gross);
```

- **COBOL** (COmmon Business-Oriented Language) is designed for business applications, but easy to read because of the English-like statements.

```
* computes regular time pay
MUL RT_PAY BY REGULAR-HRS AND PAY-RATE.

* computes overtime pay
IF OVERTIME-HRS > 0
  COMPUTE OVERTIME-PAY = OVERTIME-HRS * 1.5 * PAY-RATE
ELSE
  MOVE 0 TO OVERTIME-PAY.

* computes gross pay
ADD REGULAR-PAY TO OVERTIME-PAY.

* prints gross pay
PRINT GROSS-PAY TBD GROSS-PAY OUT FROM DETAIL LINE.
```
Object-Oriented Programming Languages and Program Development Tools

• An **object-oriented programming (OOP) language** allows programmers the ability to reuse and modify existing objects
• Other advantages include:
  - Objects can be reused
  - Programmers create applications faster
  - Work well in a RAD environment
  - Most program development tools are IDEs

Java is an object-oriented programming language developed by Sun Microsystems
• The Just-in-time (JIT) compiler converts the bytecode into machine-dependent code
Object-Oriented Programming Languages and Program Development Tools

- The Microsoft .NET Framework allows almost any type of program to run on the Internet or an internal business network, as well as computers and mobile devices
- Features include:
  - CLR (Common Language Runtime)
  - Classes

Object-Oriented Programming Languages and Program Development Tools

- C++ is an extension of the C programming language
- C# is based on C++ and was developed by Microsoft
- F# combines the benefits of an object-oriented language with those of a functional language

```csharp
using System;

public class Example
{
    public static void Main()
    {
        // Example of C# code
    }
}
```
Object-Oriented Programming Languages and Program Development Tools

**Visual Studio** is Microsoft’s suite of program development tools

**Visual Basic** is based on the BASIC programming language

**Visual C++** is based on C++

**Visual C#** combines the programming elements of C++ with an easier, rapid-development environment

---

**Object-Oriented Programming Languages and Program Development Tools**

**Step 1**
The developer designs the user interface. Parts of the mobile device shown here. A form is the window in which the user enters data. A text box shows the text typed in, and the text typed in will display.

**Step 2**
The developer assigns properties to each object. Clicks include text boxes, option buttons, buttons, labels, and the form itself.

**Step 3**
The developer writes code to define the action of each event the user triggers.

**Step 4**
The developer tests the program. The C# code is displayed after the user clicks the **Calculate** button.
Object-Oriented Programming Languages and Program Development Tools

A visual programming language is a language that uses a visual or graphical interface for creating all source code.

Borland’s Delphi is a powerful program development tool that is ideal for building large-scale enterprise and Web applications in a RAD environment.
Object-Oriented Programming Languages and Program Development Tools

- **PowerBuilder** is a powerful program development RAD tool
- Best suited for Web-based, .NET, and large-scale enterprise object-oriented applications

Other Programming Languages and Development Tools

- A **4GL** (fourth-generation language) is a nonprocedural language that enables users and programmers to access data in a database
  - One popular 4GL is **SQL**
Other Programming Languages and Development Tools

• Classic programming languages include:

<table>
<thead>
<tr>
<th>Ada</th>
<th>ALGOL</th>
<th>APL</th>
<th>BASIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forth</td>
<td>FORTRAN</td>
<td>HyperTalk</td>
<td>LISP</td>
</tr>
<tr>
<td>Logo</td>
<td>Modula-2</td>
<td>Pascal</td>
<td>PILOT</td>
</tr>
<tr>
<td>PL/1</td>
<td>Prolog</td>
<td>RPG</td>
<td>Smalltalk</td>
</tr>
</tbody>
</table>

An application generator is a program that creates source code or machine code from a specification of the required functionality

– Often bundled as part of a DBMS
Other Programming Languages and Development Tools

- A **macro** is a series of statements that instructs an application how to complete a task
- You usually create the macro in one of two ways:
  - Record the macro with a macro recorder
  - Write the macro
Web Page Development

- **HTML** is a special formatting language that programmers use to format documents for display on the Web.

- **XHTML** is a markup language that allows Web sites to be displayed more easily on mobile devices.

Web Page Development

- **XML** allows Web developers to create customized tags and use predefined tags to display content appropriately on various devices.
  - **WML** is a subset of XML and is used to design pages for microbrowsers.

- Two applications of XML are **RSS 2.0** and **ATOM**.
Web Page Development

- Web browsers can execute short programs to add interactive elements to Web pages
- To send and receive information between your computer and a Web server, these programs use the CGI (common gateway interface)
Web Page Development

• Programmers write scripts, applets, servlets, or ActiveX controls using a variety of languages

- JavaScript
- Perl
- PHP
- Rexx
- Tcl
- VBScript
Web Page Development

Dynamic HTML (DHTML) allows Web developers to include more graphical interest and interactivity.

- Cascading style sheets (CSS) contain the formats for how a particular object should be displayed.

Ruby on Rails (RoR) provides technologies for developing object-oriented, database-driven Web sites.
Web Page Development

• Web 2.0 allows Web sites to provide a means for users to:
  - Share personal information
  - Allow users to modify Web site content
  - Have application software built into the site

Web Page Development

• Most Web 2.0 sites use APIs
  - An API enables programmers to interact with an environment such as a Web site or operating system
Web Page Development

- **Web page authoring software** can create sophisticated Web pages that include images, video, audio, animation, and other effects.

  - Dreamweaver
  - Expression Web
  - Flash
  - SharePoint Designer

Multimedia Program Development

- **Multimedia authoring software** allows programmers to combine text, graphics, animation, audio, and video in an interactive presentation.

  - ToolBook
  - Director
Program Development

- **Program development** consists of a series of steps programmers use to build computer programs.
Step 1 – Analyze Requirements

- To initiate program development, programmer:
  - Reviews the requirements
  - Meets with the systems analyst and users
  - Identifies input, processing, and output
    - IPO chart

<table>
<thead>
<tr>
<th>Input</th>
<th>Processing</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regular Time</td>
<td>Read regular time hours worked, overtime hours</td>
<td>Gross Pay</td>
</tr>
<tr>
<td>Hours Worked</td>
<td>worked, hourly pay.</td>
<td></td>
</tr>
<tr>
<td>Overtime Hours</td>
<td>Calculate regular time pay.</td>
<td></td>
</tr>
<tr>
<td>Worked</td>
<td>If employee worked overtime, calculate overtime</td>
<td></td>
</tr>
<tr>
<td>Hourly Pay Rate</td>
<td>pay.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Calculate gross pay.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Print gross pay.</td>
<td></td>
</tr>
</tbody>
</table>

Step 2 – Design Solution

- Design a solution algorithm
- In **structured design**, the programmer typically begins with a general design and moves toward a more detailed design
- Programmers use a **hierarchy chart** to show program modules graphically
Step 2 – Design Solution

- With object-oriented (OO) design, the programmer packages the data and the program into a single object
  - Encapsulation
Step 2 – Design Solution

• The sequence control structure shows one or more actions following each other in order.

Sequence Control Structure

- Action 1
- Action 2
- Action 3

Step 2 – Design Solution

• The selection control structure tells the program which action to take, based on a certain condition.
  - If-then-else
  - Case

If-Then-Else Control Structure

- Condition
  - True
    - Action 1
  - False
    - Action 2
Step 2 – Design Solution

- The repetition control structure enables a program to perform one or more actions repeatedly as long as a certain condition is met.
Step 2 – Design Solution

- A program **flowchart** graphically shows the logic in a solution algorithm

![Flowchart Image]

Step 2 – Design Solution

- **Flowcharting software** makes it easy to modify and update flowcharts
  - SmartDraw
  - Visio

![Flowchart Image]
Step 2 – Design Solution

- Pseudocode uses a condensed form of English to convey program logic

```
MAIN MODULE:
CALL Initialization
CALL Process
CALL Wrap-Up
END

PROCESS MODULE:
DO WHILE Not EOF
   CALL Read a Record
   CALL Calculate
   CALL Accumulate Totals
   CALL Print Detailed Line
ENDDO
RETURN

CALCULATE OVERTIME PAY MODULE:
IF Hours Worked > 40 THEN
   Overtime Pay = Overtime Hours * 1.5 * Pay Rate
ELSE
   Overtime Pay = 0
ENDIF
RETURN
```

Step 2 – Design Solution

- UML (Unified Modeling Language) has been adopted as a standard notation for object modeling and development
Step 3 – Validate Design

- Check for **logic errors** using **test data**

Step 4 – Implement Design

- **Implementation** of the design includes using a program development tool that assists the programmer by:
  - Generating or providing some or all code
  - Writing the code that translates the design into a computer program
  - Creating the user interface

- Extreme programming is a strategy where programmers immediately begin coding and testing solutions as soon as requirements are defined.
Step 5 – Test Solution

The goal of program testing is to ensure the program runs correctly and is error free.

- Errors include syntax errors and logic errors.
- Debugging the program involves removing the bugs.
- A beta is a program that has most or all of its features and functionality implemented.

Step 6 – Document Solution

- In documenting the solution, the programmer performs two activities:
  - Review the program code
  - Review all the documentation
Summary

- Various programming languages used to create computer programs
- A variety of Web development and multimedia development tools
- Steps in the program development life cycle and tools used to make this process efficient