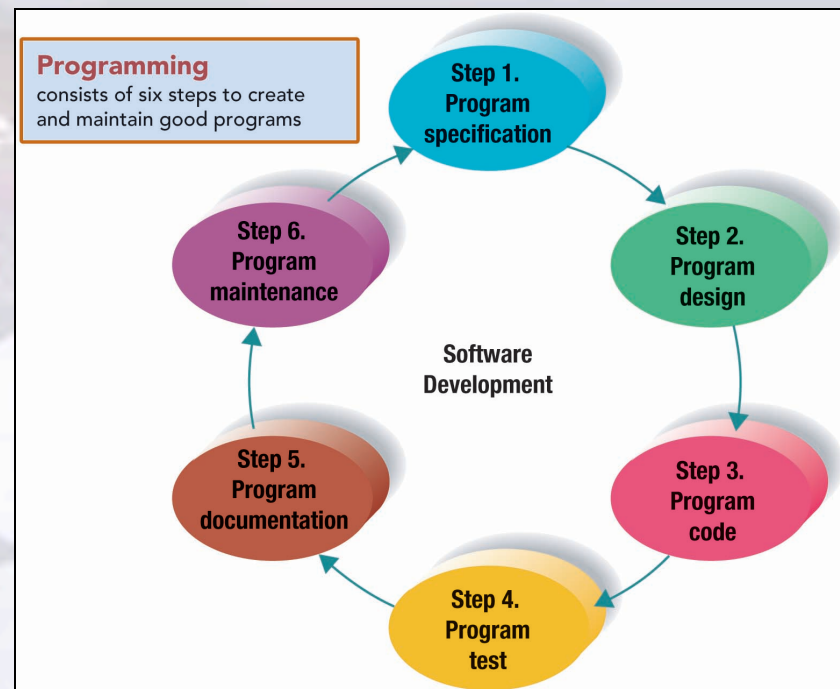


Chapter 14

Programming and Languages

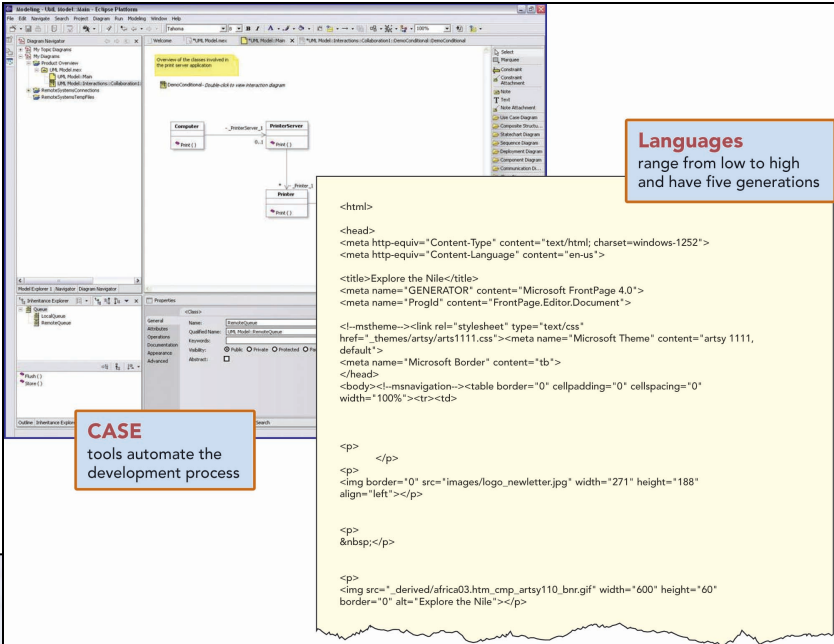
Competencies (Page 1 of 2)

- Describe the six steps of **programming**
- Discuss design tools including top-down design, **pseudocode**, flowcharts, and **logic structures**
- Describe program testing and the tools for finding and removing errors



Competencies (Page 2 of 2)

- Describe CASE tools and **object-oriented software development**
- Explain the five **generations of programming languages**



The screenshot displays the Microsoft Visual Studio IDE. On the left, a UML class diagram shows a class hierarchy with 'Computer' as a base class and 'WebServer' as a derived class. The 'WebServer' class has a 'Page' attribute. The right pane shows the generated HTML code for a web page titled 'Explore the Nile'. The code includes a head section with meta tags for content type and language, a title, and a link to a stylesheet. The body contains a navigation table and two image tags.

Languages
range from low to high
and have five generations

CASE
tools automate the
development process

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<meta http-equiv="Content-Language" content="en-us">
<title>Explore the Nile</title>
<meta name="GENERATOR" content="Microsoft FrontPage 4.0">
<meta name="ProgId" content="FrontPage.Editor.Document">
</head>
<body>
<table border="0" cellpadding="0" cellspacing="0" width="100%">
<tr>
<td>
</td>
</tr>
</table>
<p>
</p>

</img>
<br>
</p>

</img>
```

Introduction

In this chapter programming is described in two parts...

- **The steps in the programming process**
- **Some of the programming languages available**

Programming is part of Phase 4, **Systems Development, in the **Systems Life Cycle**.**

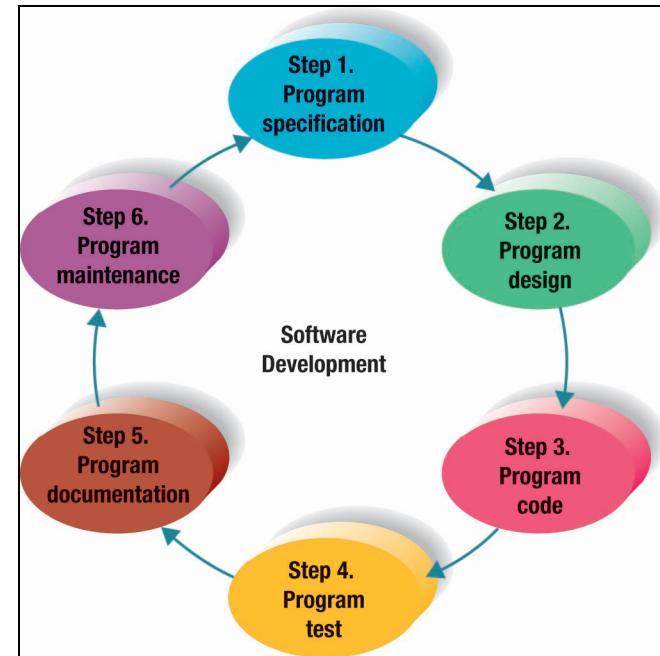
Competent end users need to understand the relationship between systems development and programming.

Programs and Programming

- **What is a **Program**?**
 - A program is a list of instructions
 - Code, Application
 - Programmer, Hacker
- **A problem-solving procedure**
 - A list of instructions
 - Prewritten
 - Custom-written
 - Application software
 - System software

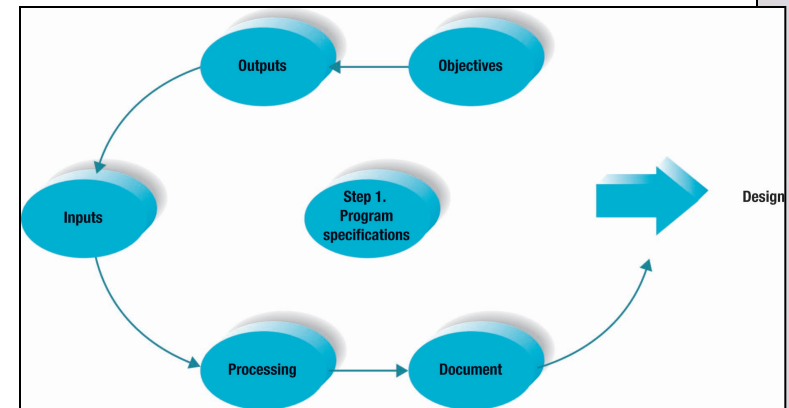
What is Programming?

- **Programming is software development**
- **Six step procedure**
 - **Program specification**
 - **Program design**
 - **Program code**
 - **Program test**
 - **Program documentation**
 - **Program maintenance**



Step 1: Program Specification

- Also called programs definition or **program analysis**
- **5 Step Process**
 - Program Objectives
 - Desired Output
 - Input Data
 - Processing Requirements
 - Document Program Specifications



Program Objectives

- **Objectives are the problems that you are trying to solve**
- **Programming requires a clear statement of the problem that you are looking to address**



Program Specification Steps Continued

- **Desired output**
- **Input Data**
- **Processing Requirements**
- **Program Specifications Document**

Client name: Allen Realty		Month and year: Jan '07		
Date	Worker	Regular Hours & Rate	Overtime Hours & Rate	Bill
1/2	M. Jones	5 @ \$10	1 @ \$15	\$65.00
	K. Williams	4 @ \$30	2 @ \$45	\$210.00

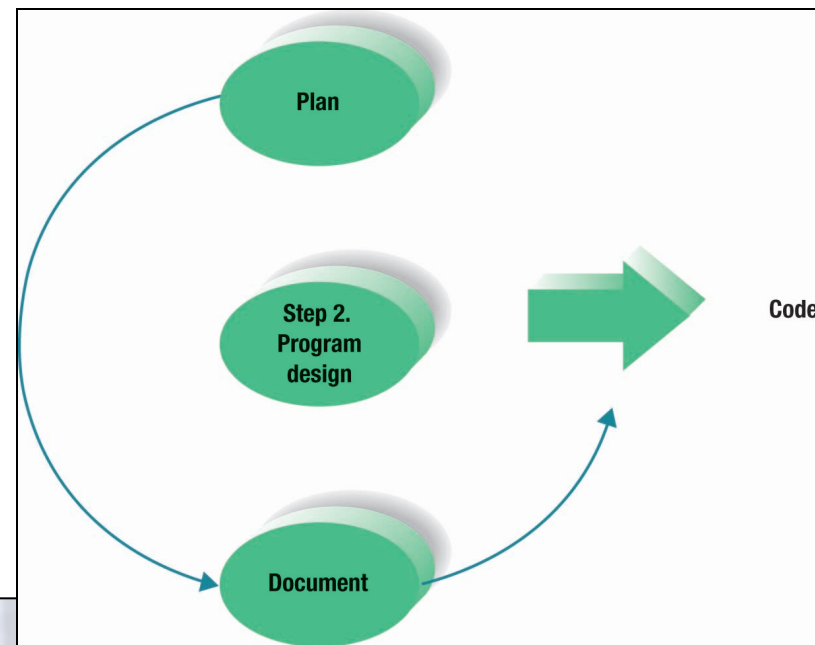
End user's sketch of desired output

Daily Log			
Worker:			
Date:			
Client	Job	Time in	Time out
A	TV commercial	800	915
B	Billboard ad	935	1200
C	Brochure	1315	1545
D	Magazine ad	1600	1745

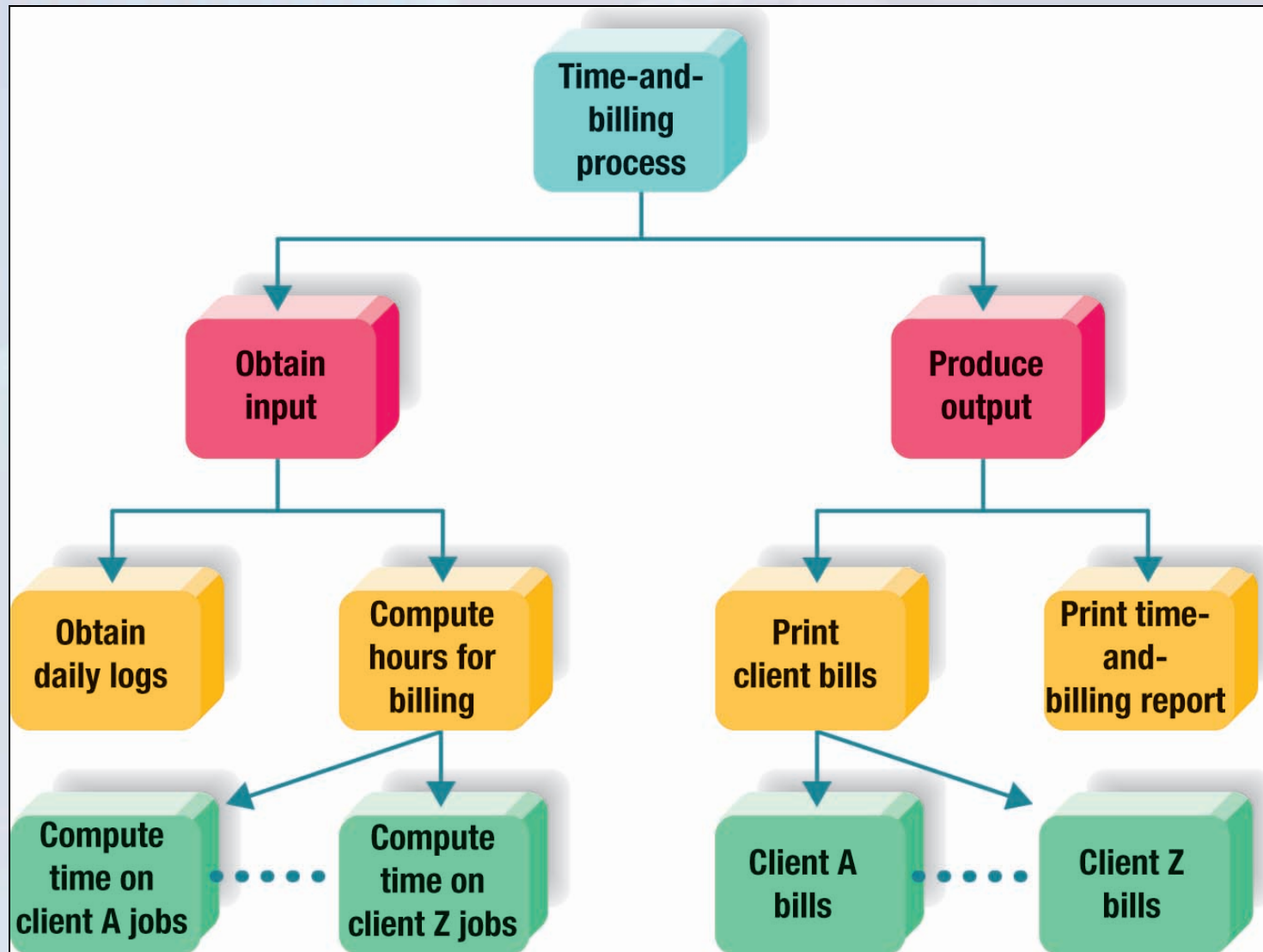
Example of statement of hours worked

Step 2: Program Design

- **Plan a solution**
- **Use structured programming techniques**
 - Top-down program design
 - Pseudocode
 - Flowcharts
 - Logic structures



Top-Down Program Design



Pseudocode

Summarizes the logic for a program.

Informal list of instructions.

Like a program, but written in English words.

Compute time for Client A

Set total regular hours and total overtime hours to zero.

Get time in and time out for a job.

If worked past 1700 hours, then compute overtime hours.

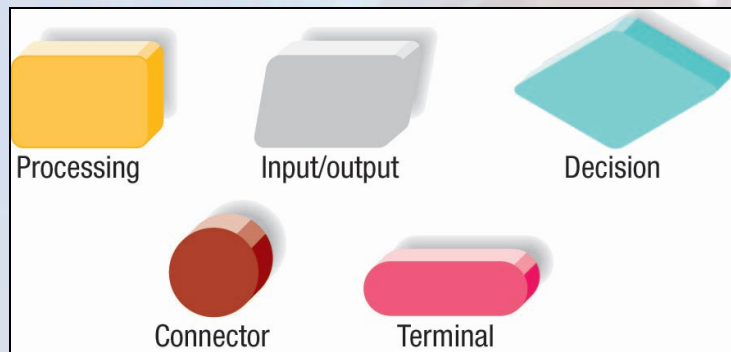
Compute regular hours.

Add regular hours to total regular hours.

Add overtime hours to total overtime hours.

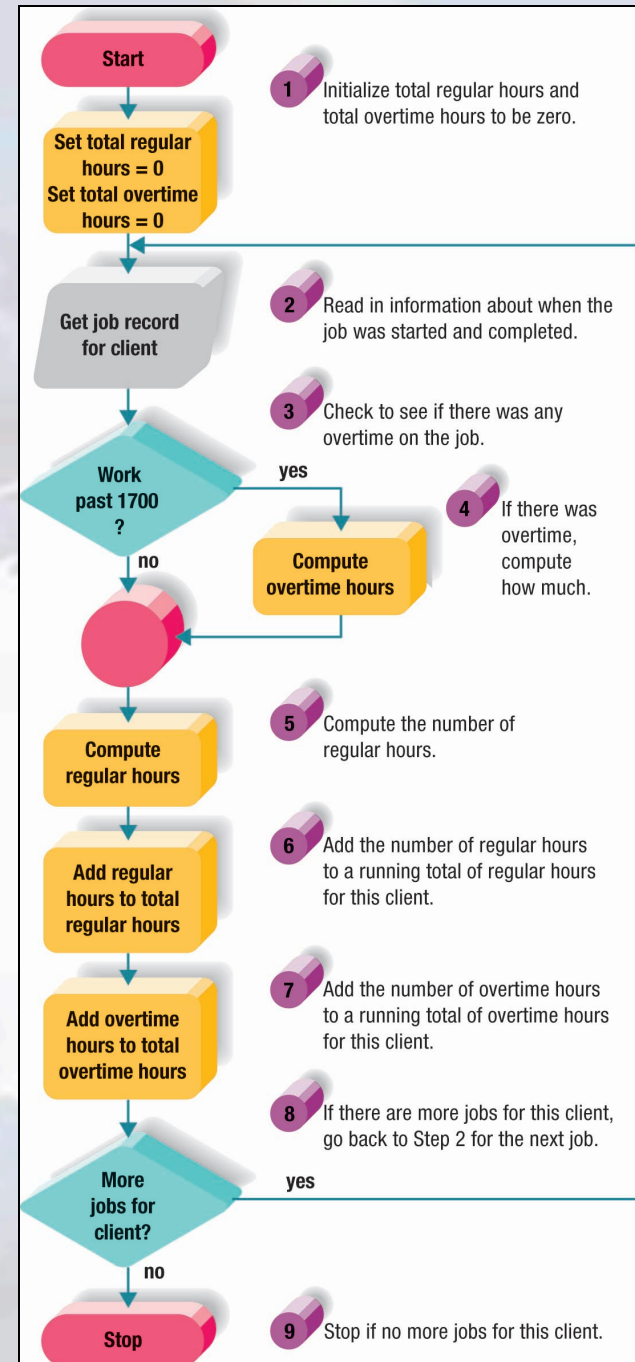
If there are more jobs for that client, go back and compute for that job as well.

Flowchart



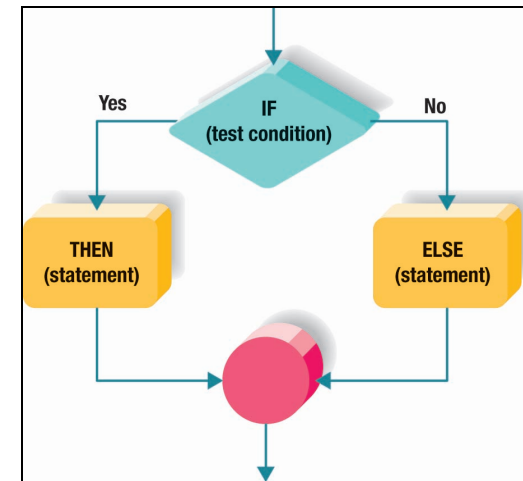
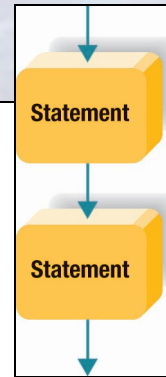
Flowchart symbols

Flowchart example

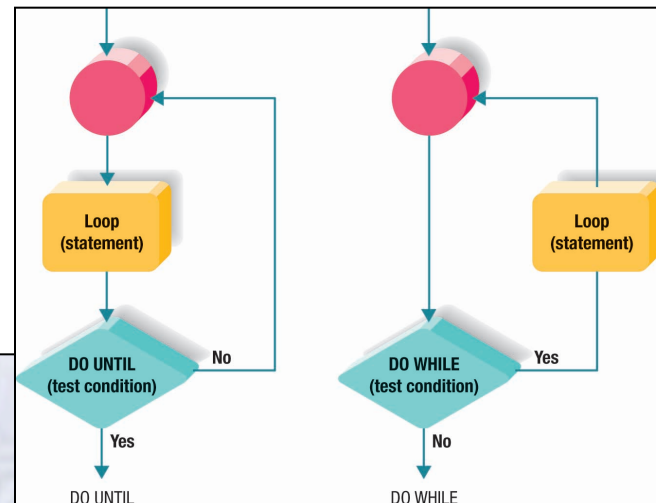


Logic Structures

- **Sequence structure**
- **Selection structure**
- **Loop structure**

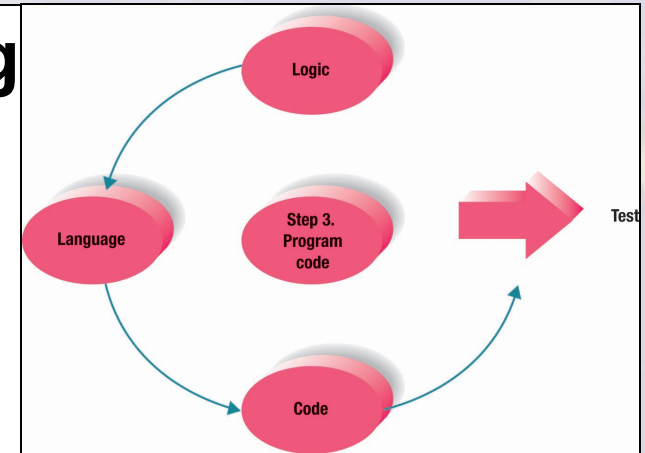


Technique	Description
Top-down design	Major processing steps, called program modules, are identified
Pseudocode	A narrative expression of the logic of the program is written
Program flowcharts	Graphic representation of the steps needed to solve the programming problem is drawn
Logic structures	Three arrangements are used in program flowcharts to write structured programs



Step 3: Program Code

- **Writing the program or coding**
- **The good program**
 - **Reliable**
 - **Catch input errors**
 - **Understandable to other programmers**
 - **Structured programs** best method
 - **Use the most appropriate computer language**
- **Coding**



Coding

- **Formatting or presentation language**

Language	Description
HTML	Stands for HyperText Markup Language and is the most common formatting language to present Web pages
DHTML	Stands for Dynamic HTML and improves HTML by including animations, interaction, and dynamic updating
XHTML	Stands for eXtended HTML; combines HTML and XML to add structure and flexibility to HTML
XML	Stands for eXtensible Markup Language; assists sharing of data and interactivity
WML	Stands for Wireless Markup Language; provides a standard for describing data in wireless applications

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<meta http-equiv="Content-Language" content="en-us">

<title>Explore the Nile</title>
<meta name="GENERATOR" content="Microsoft FrontPage 4.0">
<meta name="ProgId" content="FrontPage.Editor.Document">

<!--mstheme--><link rel="stylesheet" type="text/css"
href="..themes/artsy/arts1111.css"><meta name="Microsoft Theme" content="artsy 1111,
default">
<meta name="Microsoft Border" content="tb">
</head>
<body><!--msnavigation--><table border="0" cellpadding="0" cellspacing="0"
width="100%"><tr><td>

<p>
</p>
<p>
</p>

<p>
&nbsp;</p>

<p>
</p>
```

- **Programming language**

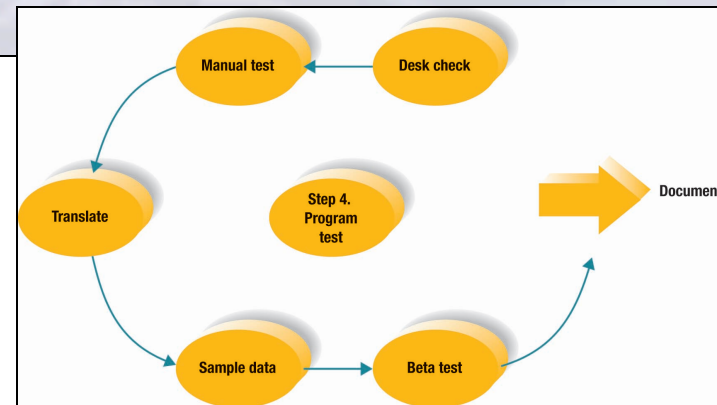
Language	Description
C	Widely used programming language, often associated with the UNIX operating system
C++	Extends C to use objects or program modules that can be reused and interchanged between programs
C#	Extends C++ to include XML functionality and support for a new Microsoft initiative called .NET
Java	Primarily used for Internet applications, similar to C++, runs with a variety of operating systems
JavaScript	Embedded into Web pages to provide dynamic and interactive content
Visual Basic	Uses a very graphical interface making it easy to learn and to rapidly develop Windows and other applications

Common Programming Languages

- C
- C++
- Java
- Python, Perl
- Visual Basic
 - Basic
- Fortran – Older, for numerical
- Cobol – Older, for transaction processing systems (TPS)
- Assembler (= machine language)

Step 4: Program Test

- **Debugging**
 - Testing
 - Eliminating errors
- **Syntax errors**
- **Logic errors**
- **Testing process**



Task	Description
1	Desk check for syntax and logic errors
2	Manually test with sample data
3	Translate program to identify syntax errors
4	Run program with sample data
5	Beta test with potential users

```

#include <fstream.h>

void main (void)
{
    ifstream input_file;

    float total_regular, total_overtime, regular, overtime;
    int hour_in, minute_in, hour_out, minute_out;
    input_file.open("time.txt", ios::in);

    total_regular = 0;
    total_overtime = 0;

    while (input_file != NULL)
    {
        input_file >> hour_in >> minute_in >> hour_out >> minute_out
        if (hour_out >= 17)
            overtime = (hour_out-17) +(minute_out/(float)60);
        else
    }
}
  
```

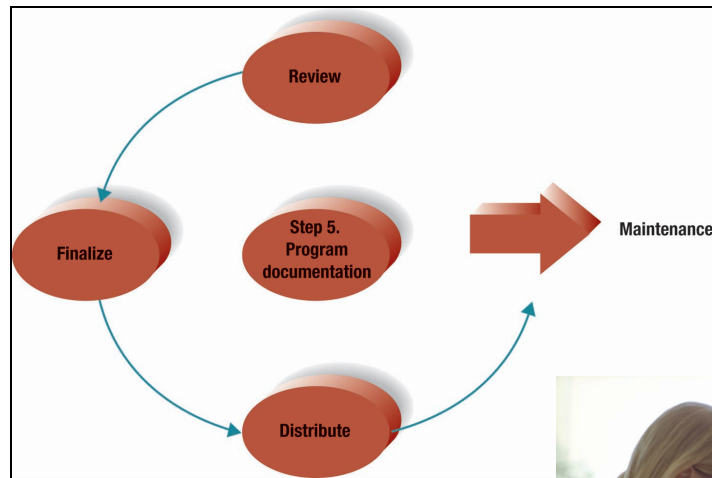
A:\Compute Time.cpp(18) : error C2143: syntax error : missing ';' before 'if'
 Error executing cl.exe.
 Compute Time.exe - 1 error(s), 0 warning(s)

Types of Testing

- **Desk checking – sit and look at it**
- **Manual testing – try specific data values**
- **Translation – try to “compile” i**
 - This catches syntax errors
- **Run the program**
 - Testing on sample data
- **Beta testing**
 - Give preliminary version to a few brave users

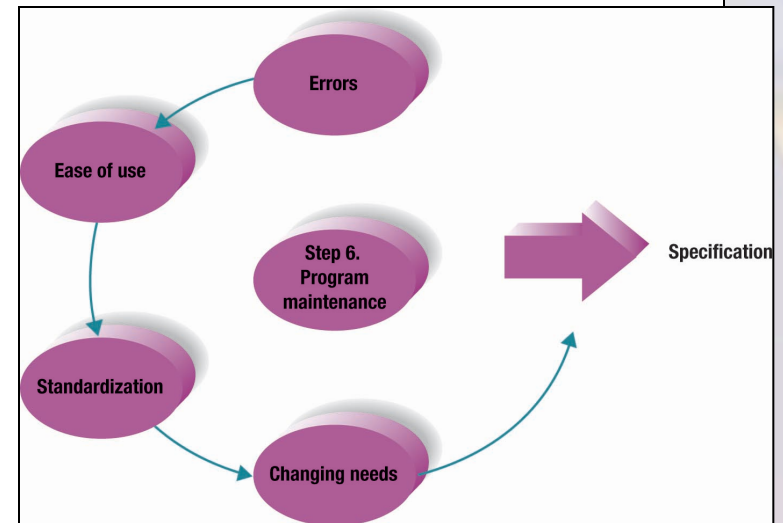
Step 5: Program Documentation

- **Written descriptions and procedures about a program**
- **Important for people who will use the program**
 - Users
 - **Operators**
 - Programmers



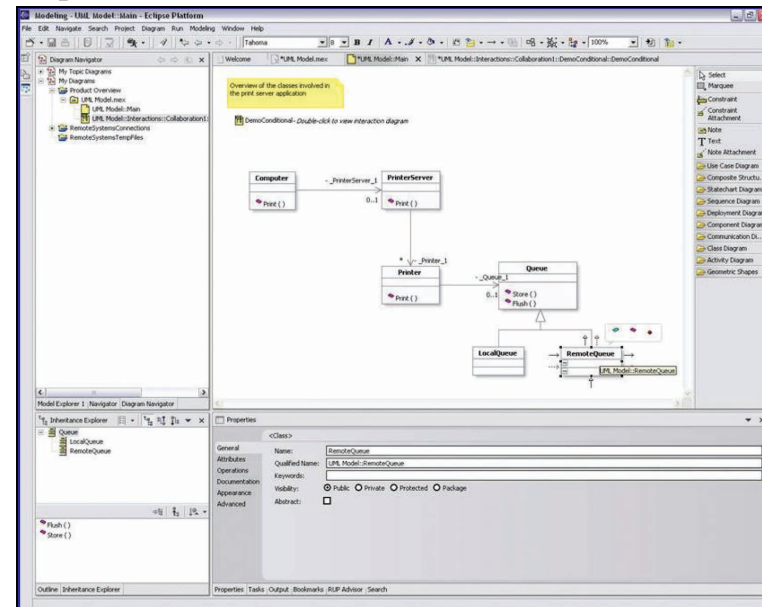
Step 6: Program Maintenance

- **75% of total lifetime cost**
- **Error-free operations**
- **Effective program**
- **Two categories**
 - **Operations**
 - **Changing needs**



Case Tools

- **Computer-aided software engineering (CASE)**
 - Automate development process
 - Designing
 - Coding
 - Testing activities



Object-Oriented Software Development

- Focuses less on procedures, more on relationship between **objects**
- **OOP: Object-oriented programming**
- **Object contains both the data and the processing operations**

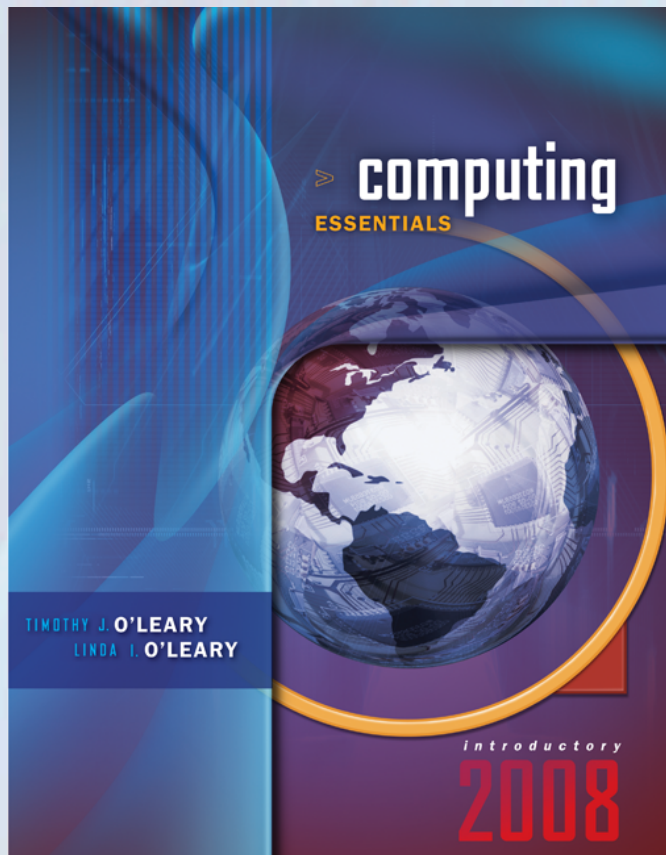
Generations of Programming Languages (Page 1 of 2)

- Occurring in “generations” or “**levels**”
 - Levels-**Machine languages** to **natural languages**
- There are five generations :
 - **Lower level** closer to machine language
 - **Higher level** closer to human-like language

Generations of Programming Languages (Page 2 of 2)

- 1st -- Machine languages
- 2nd -- Assembly languages
- 3rd -- High level procedural languages (3GL)
- 4th -- Problem-Oriented languages (4GL)
- 5th -- Natural Languages & Visual programming languages (5GL)

Generation	Sample Statement
First: Machine	111100100111001111010010000100000111000000101011
Second: Assembly	ADD 210(8, 13),02B(4, 7)
Third: Procedural	if (score > = 90) grade = 'A';
Fourth: Problem	SELECT client FROM dailyLog WHERE serviceEnd > 17
Fifth: Natural and Visual	If patient is dizzy, then check temperature and blood pressure.



Chapter 15

Your Future and Information Technology

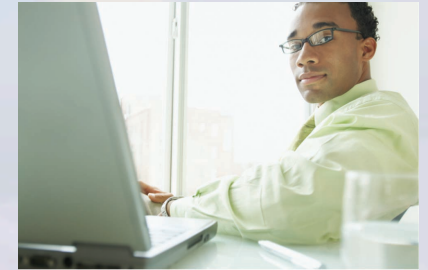
CH15: Technology and People

- **Cynicism**
- **Naiveté**
- **Frustration**



- **Proactivity**

Careers In IT



- **Computer programmers create, test, and troubleshoot programs**
- **Responsibility may also include updating and repairing existing programs**
- **Employers seek individuals with a bachelors degree in computer science or information systems but there are positions available for those with a two-year degree**
- **Desired traits include patience, logical thinking, and attention to detail**
- **Computer Programmers can expect to earn \$48K - \$81K annually**

A Look to the Future

MI-Tech

- **Synapse Solutions has created a system called MI-tech**
- **Understands word order and meaning**
- **Computer translates "wish list" into machine language**



Discussion Questions (Page 1 of 2)

- **Identify and discuss each of the six steps of programming.**
- **Describe CASE tools and OOP. How does CASE assist programmers?**
- **What is meant by “generation” in reference to programming languages? What is the difference between low-level and high-level languages?**

Discussion Questions (Page 2 of 2)

- **What is the difference between a compiler and an interpreter?**
- **What are logic structures? Describe the differences between the three logic types.**