

Computer Programs and Programming Languages

What is a **computer program**?

- Set of instructions that directs computer to perform tasks
 - **Programming language**—used to write instructions



Next

Computer Programs and Programming Languages

What are **low-level languages** and **high-level languages**?

Low-level language

Machine-dependent
runs only on one type of computer

Machine and assembly languages
are low-level

High-level language

Often **machine-independent**
can run on many different
types of computers

Next

Low-Level Languages

What is **machine language**?

- Only language computer
directly recognizes

000090	50E0	3082		000990
000094	1844			010B4
000096	1877			
000098	1855			
00009A	F273	3006	2C81	00C83
0000A0	4F50	3006		010D8
0000A4	F275	3006	2C7B	00C70
0000AA	4F70	3006		010D8
0000AE	5070	304A		0104C
0000B2	1C47			
0000B4	5050	304E		01050
0000B8	58E0	3082		010B4
0000BC	07FE			
0000BE	50E0	3086		000BE
0000C2	95F1	2C85		010B8
0000C6	4770	2002		00C87
0000CA	1855			00004
0000CC	5A50	35A6		
0000D0	47F0	2100		015A8
0000D4	95F2	2C85		00102
0000D8	4770	20E4		00C87
0000DC	1855			000E6
0000DE	5A50	35AA		
0000E2	47F0	2100		015AC
000102	1877			
000104	5870	304E		01050
000108	1C47			
00010A	4E50	3006		010D8
00010E	F075	3006		0003E
000114	4F50	3006	003E	010D8
000118	5050	3052		01054
00011C	58E0	3086		010B8
000120	07FE			
000122	50E0	308A		00122
000126	1855			010BC
000128	5A50	304E		
00012C	5850	3052		01050
000130	5050	305A		01054
000134	58E0	308A		0105C
000138	07FE			010BC

Next

Low-Level Languages

What is **assembly language**?

- Instructions made up of **symbolic instruction codes**, meaningful abbreviations and codes
- **Source program** contains code to be converted to machine language

```
* THIS MODULE CALCULATES THE REGULAR TIME PAY
*
CALCSTPY EQU *
ST 14,SAVERTPY
SR 4,4
SR 7,7
SR 5,5
PACK DOUBLE,OTHRIN
CVB 4,DOUBLE
PACK DOUBLE,RATEIN
CVB 7,DOUBLE
ST 7,RATE
MR 4,7
ST 5,OTPAY
L 14,SAVERTPY
BR 14

* THIS MODULE CALCULATES THE OVERTIME PAY
*
CALCOTPY EQU *
ST 14,SAVERTPY
TEST1 CLI CODEIN,C'D'
SR 5,5
A 5,-F'D'
ST 5,OTPAY
B AROUND
SR 4,4
SR 7,7
SR 5,5
PACK DOUBLE,OTHRIN
CVB 4,DOUBLE
PACK DOUBLE,RATEIN
CVB 7,RATE
MR 4,7
MR 4,-F'1.5'
ST 5,OTPAY
L 14,SAVERTPY
BR 14

* THIS MODULE CALCULATES THE GROSS PAY
*
CALCGPAY EQU *
ST 14,SAVERTPY
SR 5,5
A 5,OTPAY
A 5,OTPAY
ST 5,OTPAY
L 14,SAVERTPY
BR 14
```

Next

Procedural Languages

What is a **procedural language**?

Programmer assigns name to sequence of instructions that tells computer what to accomplish and how to do it

Uses series of English-like words to write instructions

Often called **third-generation language (3GL)**

Most widely used are BASIC, COBOL, and C

Next

Procedural Languages

What is a **compiler**?

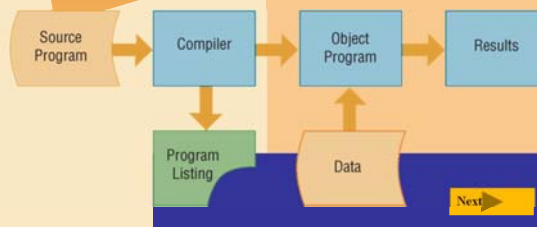
- Program that converts entire source program into machine language before executing it

```
* 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.

* PRINT GROSS PAY
MOVE GROSS-PAY TO GROSS-PAY-OUT.
WRITE REPORT-LINE-OUT FROM DETAIL-LINE
AFTER ADVANCING 2 LINES.
```

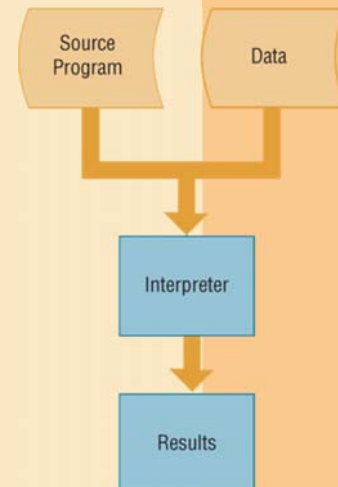


Next

Procedural Languages

What is an **interpreter**?

- Program that translates and executes one program code statement at a time
- Does not produce object program



Next

Procedural Languages

What is **BASIC**?

- Designed for use as simple, interactive problem-solving language
- Beginner's **All-purpose Symbolic Instruction Code**

```
REM COMPUTE REGULAR TIME PAY
Regular.Time.Pay = Regular.Time.Hours * Hourly.Pay.Rate

REM COMPUTE OVERTIME PAY
If Overtime.Hours > 0 THEN
    Overtime.Pay = Overtime.Hours * 1.5 * Hourly.Pay.Rate
ELSE
    Overtime.Pay = 0
END IF

REM COMPUTE GROSS PAY
Gross.Pay = Regular.Time.Pay + Overtime.Pay
REM PRINT GROSS PAY
PRINT USING "The gross pay is $##.###.##"; Gross.Pay
```

Next ➤

Procedural Languages

What is **COBOL**?

- Designed for business applications
- English-like statements make code easy to read, write, and maintain
- **CO**mmon **B**usiness-**O**riented **L**anguage

```
* 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.

* PRINT GROSS PAY
MOVE GROSS-PAY TO GROSS-PAY-OUT.
WRITE REPORT-LINE-OUT FROM DETAIL-LINE
AFTER ADVANCING 2 LINES.
```

Next ➤

Procedural Languages

What is **C**?

- Powerful language originally designed to write system software
- Requires professional programming skills

```
/* 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;

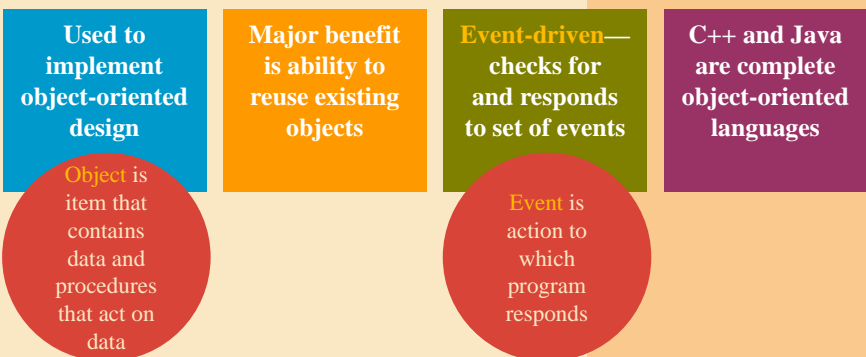
/* Compute Gross Pay */
gross = rt_pay + ot_pay;

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

Next ➤

Object-Oriented Programming Languages

What is an **object-oriented programming (OOP)** language?



Next ➤

Object-Oriented Programming Languages

What is **C++**?

- Includes all elements of C, plus additional features for working with object-oriented concepts
- Used to develop database and Web applications

```
// portion of a C++ program that allows users to create a new zip code from a
// string or a number and expand zip codes, as appropriate, to a 10-digit number

ZipC::ZipC( const unsigned long zipnum )
{
    stringstream strInt;
    strInt << zipnum;
    code = strInt.str();
}

const string ZipC::getCode()
{
    return code;
}

void ZipC::setCode(const string newCode)
{
    code = newCode;
}

void ZipC::expand( const string suffix )
{
    if(code.length() == 5 && // small size?
       suffix.length() == 4) // length ok?
    {
        code += "-";
        code.append(suffix);
    }
}
```

Next

Object-Oriented Programming Languages

What is **Java**?

- Developed by Sun Microsystems
- Similar to C++ but uses **just-in-time (JIT) compiler** to convert source code into machine code

```
public void actionPerformed(ActionEvent e)
{
    foundKey = false;

    //Search for the key pressed
    for (int i = 0; i < keyArray.length && !foundKey; i++)
        if(keyArray[i] == keySource) //key match found
        {
            foundKey = true;
            switch(i)
            {
                case 0: case 1: case 2: case 3: case 4: //number buttons
                case 5: case 6: case 7: case 8: case 9: //0 - 9
                case 10: //decimal point button
                {
                    if(clearText)
                    {
                        lcdField.setText("");
                        clearText = false;
                    }
                    lcdField.setText(lcdField.getText() + keyArray[i].getLabel());
                    break;
                }
            }
        }
}
```

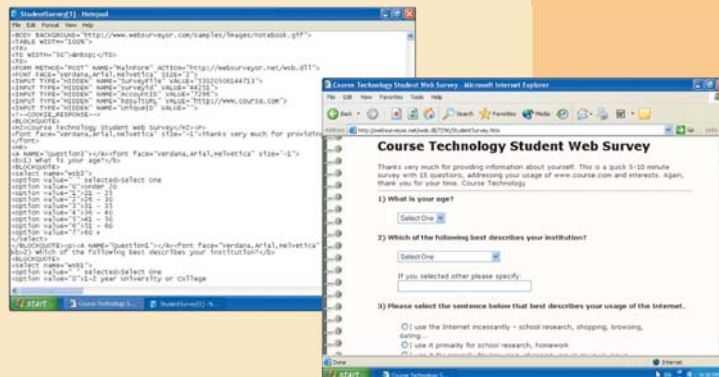


Next

Web Page Development

What is **HTML (Hypertext Markup Language)**?

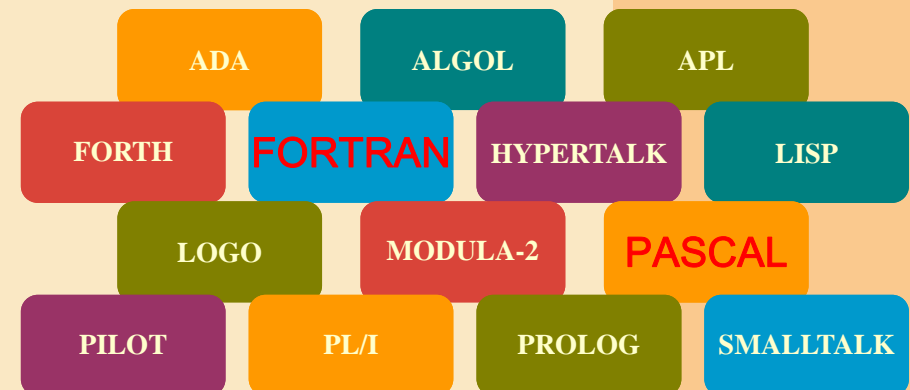
- Used to create Web pages



Next

Other Programming Languages

What are other available programming languages?



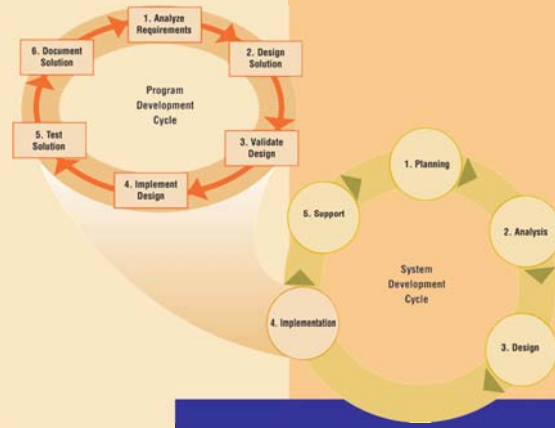
Next

The Program Development Cycle

What is the **program development cycle**?

➤ Steps programmers use to build computer programs

- **Programming team**—Group of programmers working on program

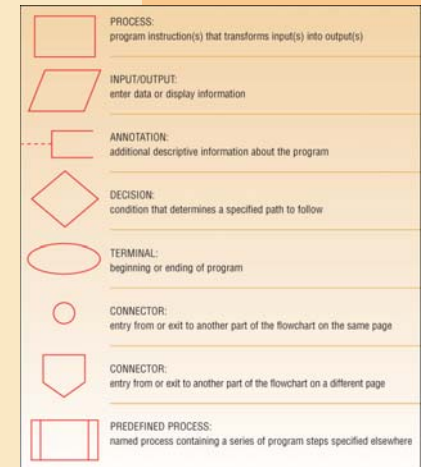


Next ➤

Design Solution

What is a **program flowchart**?

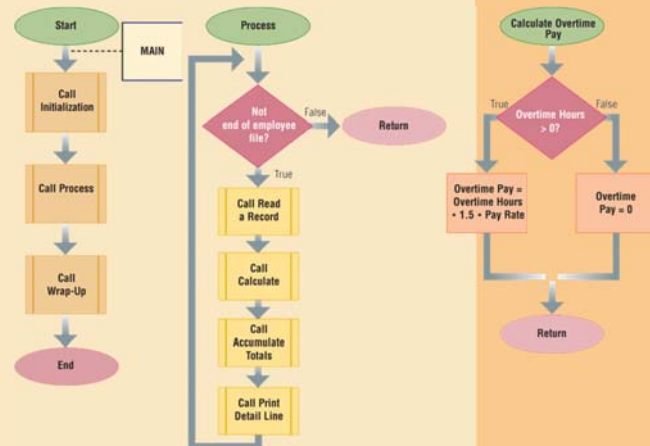
➤ Graphically shows logic in solution algorithm



Next ➤

Design Solution

What is an example of a flowchart?

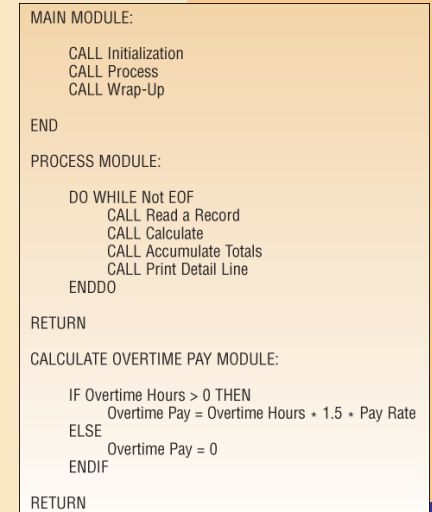


Next ➤

Design Solution

What is **pseudocode**?

➤ Uses condensed form of English to convey program logic



Next ➤

1.14 Typical C++ Development Environment

Phase 1: Creating a Program in an Editor
source code in editor program

.cpp, .cxx, .cc, .C

Phase 2 and 3: Preprocessing and Compiling a C++ Program
preprocessor (directives)
converting to machine language

Phase 4: Linking
functions or data else where
linking them to an

Phase 5: Loading
placed in memory

Phase 6: Executing a Program

