

## Problem-Solving Techniques

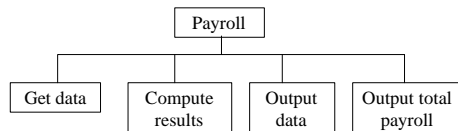
- Ask questions.
- Look for things that are familiar.
- Solve by analogy.
- Identify the inputs and outputs.
- Divide and conquer.
- Use building blocks (new or existing).
- Merge solutions.

## Problem-Solving Case Study

- Problem: A small company needs an interactive program to figure its weekly payroll. The input data and each employee's wages should be saved in a secondary storage file, and the total wages for the week should be displayed on the screen so that the payroll clerk can transfer the appropriate amount into the payroll account.
  - Data consists of id, hourly pay rate, and hours worked
  - Regular wage for 40 hours, time and a half after that
  - Store results in a file called `payFile`
  - Use id 0 to indicate end of data since it is not a valid id

## Case Study (continued)

- Four obvious steps
  - Get the data
  - Compute the results
  - Output data
  - Output total payroll



## Main Algorithm

Open file `payFile`, Set total payroll to zero

Get Data

As long as the employee number is not zero

    Compute Result

    Add employee's wages to the total payroll

    Output data

    Get data

Output total payroll

## Get Data

Prompt user for id number  
 Read id number  
 Prompt user for pay rate  
 Read pay rate  
 Prompt user for number hours worked  
 Read number hours worked

## Compute Results

If hours worked is greater than 40, then

$\text{Wages} = 40 * \text{pay rate} + (\text{hours} - 40) * 1.5 * \text{pay rate}$

Otherwise

$\text{Wages} = \text{hours} * \text{pay rate}$

## Output Data

Write id number into payFile  
 Write pay rate into payFile  
 Write hours worked into payFile  
 Write wages into payFile

## Output Total Payroll

- Write total payroll to the screen

## Payroll Program

```

//*****
// Payroll program
// This program computes each employee's wages and the
// total company payroll
//*****

#include <iostream>
#include <fstream>

using namespace std;

void CalcPay (float, float, float);

const float MAX_HOURS = 40.0; // Maximum normal hours
const float OVERTIME = 1.5;  // Overtime pay factor

```

## Payroll Program (continued)

```

int main
{
    float payRate;      // Employee's pay rat
    float hours;         // Hours worked
    float wages;         // Wages earned
    float total;         // Total company payroll
    int empNum;          // Employee ID number
    ofstream payFile;    // Company payroll file

    payFile.open("payfile.dat"); // Open the output file
    total = 0.0;           // Initialize total

```

## Payroll Program (continued)

```

cout << "Enter employee number: "; // Prompt
cin >> empNum;                     // Read employee id no.
while (empNum != 0)
{
    cout << "Enter pay rate: "; // Prompt
    cin >> payRate;             // Read hourly pay rate
    cout << "Enter Hours worked: "; // Prompt
    cin >> hours;               // Read hours worked
    CalcPay(payRate, hours, wages); // Compute wages
    total = total + wages;         // Add wages to total
    payFile << empNum << payRate // Put results in file
    << hours << wages;
    cout << "Enter employee number: "; //Prompt
    cin >> empNum;                 // Read ID number
}
cout << "Total payroll is " // Print total payroll
<< total << endl;         // on screen
return 0;                  // Indicate successful
                             // completion

```

## Payroll Program (continued)

```

//*****
void CalcPay(/* in */ float payRate, // Employee's pay rate
            /* in */ float hours,   // Hours worked
            /* out */ float& wages) // Wages earned

// CalcPay computes wages from the employee's pay rate
// and the hours worked, taking overtime into account

{
    if (hours > MAX_HOURS; // Is there overtime?
        wages = (MAX_HOURS * payRate) + // Yes
        (hours - MAX_HOURS) * payRate * OVERTIME;
    else
        wages = hours * payRate; // No
}

```

## Always Remember and Never Forget

---

- Computers are dumb; they must be told what to do.
- A computer is a tool to help you solve problems.
- You need to apply the same strategies to solving problems using a computer as you do by hand.