# *Laboratory 1B*

CS-102

Warren Littlefield

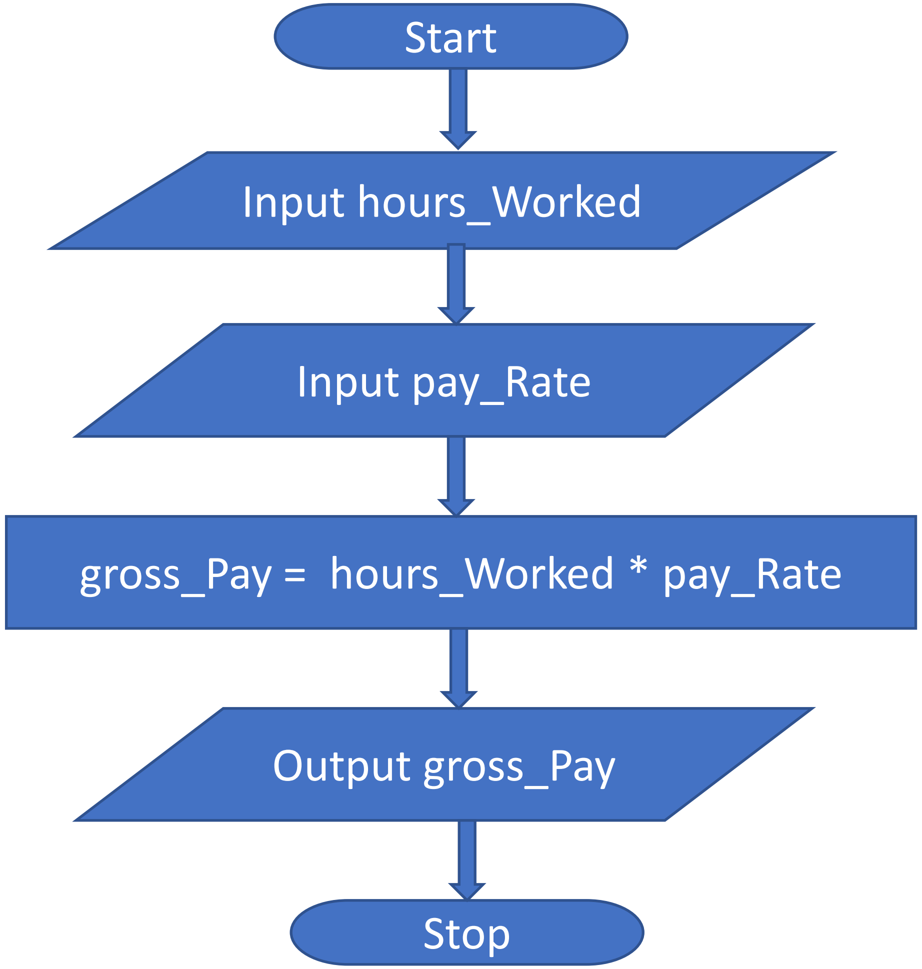
# In this Lab we will design a Program which will produce the Gross Pay earned by an employee, using Pseudocode, Flowchart, and Hierarchy chart

- The 1st step is to clearly define what the program is to do.
    - What is the purpose of the program?
    - What is the information to be input?
    - What is the processing that needs to take place?
    - What's the desired output?
- In our sample program:
    - Purpose: To calculate the user's gross pay
    - Input: Number of hours worked, hourly pay rate
    - Process: Multiply # hours worked by pay rate
    - Output: Display the message indicating user's pay

# You are given Pseudocode for Gross Pay

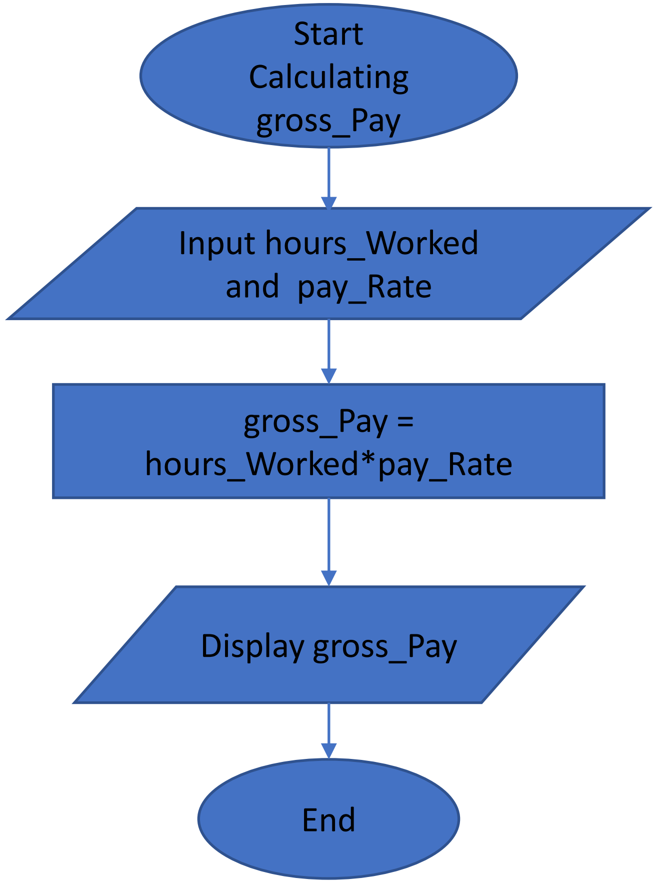
- Prompt the user to input the Hours worked
    - User enters the Hours
- Prompt the user input the hourly PayRate
    - User enters the PayRate
- Calculate the GrossPay
    - GrossPay = Hours * PayRate
- Display the GrossPay on the Screen

# Part 1: Now draw the Flowchart for the Gross Pay Calculation



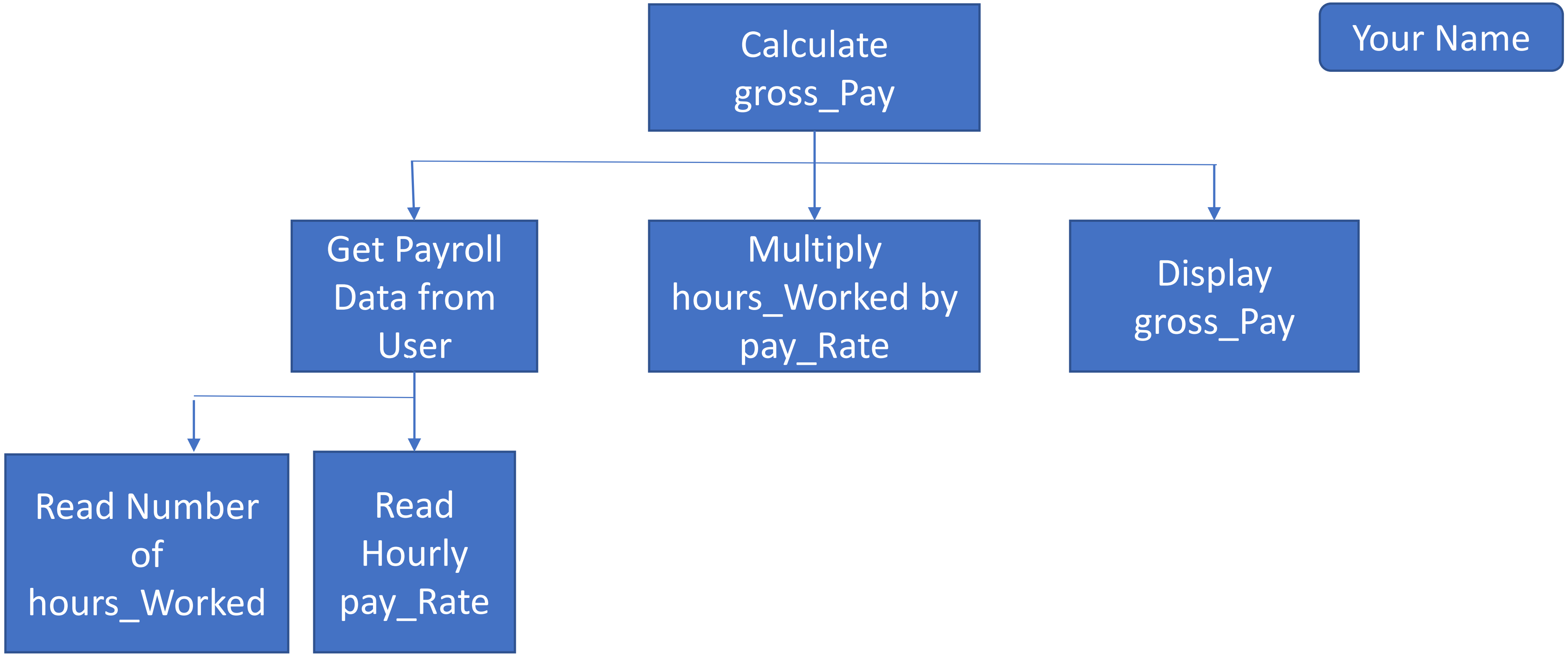
Your Name

# Laboratory 1B: Creating a Flow Chart
# The link is: https://youtu.be/r5bxbxuExhk


Start Calculating gross_Pay
Input hours_Worked and pay_Rate
gross_Pay = hours_Worked*pay_Rate
Display gross_Pay
End

- Go on-line and watch the 7 minute video on how to create the flowchart shown at left.
- Using what you learned from the video, proceed to create the flow chart shown on the previous page.
    - Note the flow chart that you are to draw is similar to the one shown at left, but it is not identical.
- Be sure to note that capitalization is important, as well as the exact shapes of each of these flow-chart elements.
- When you have completed it, call your instructor who will credit you for your work.
- **Use naming conventions shown at left instead of the ones shown in the video.**
- **Save the file as a pdf instead of as an image as shown in the video.**

# Part 2: Now draw theHierarchy Chart For Gross Pay Calculation



Calculate gross_Pay

Your Name

Get Payroll Data from User

Multiply hours_Worked by pay_Rate

Display gross_Pay

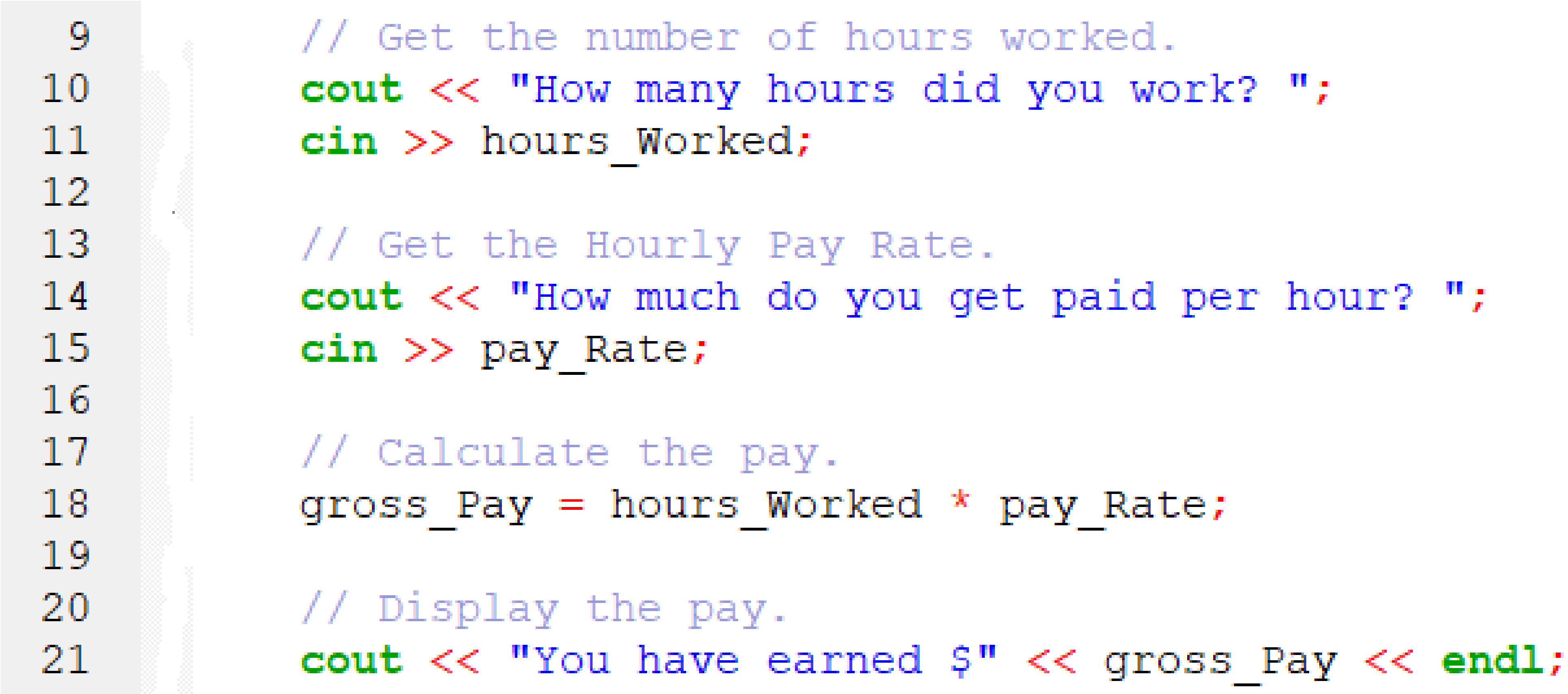
Read Number of hours_Worked

Read Hourly pay_Rate

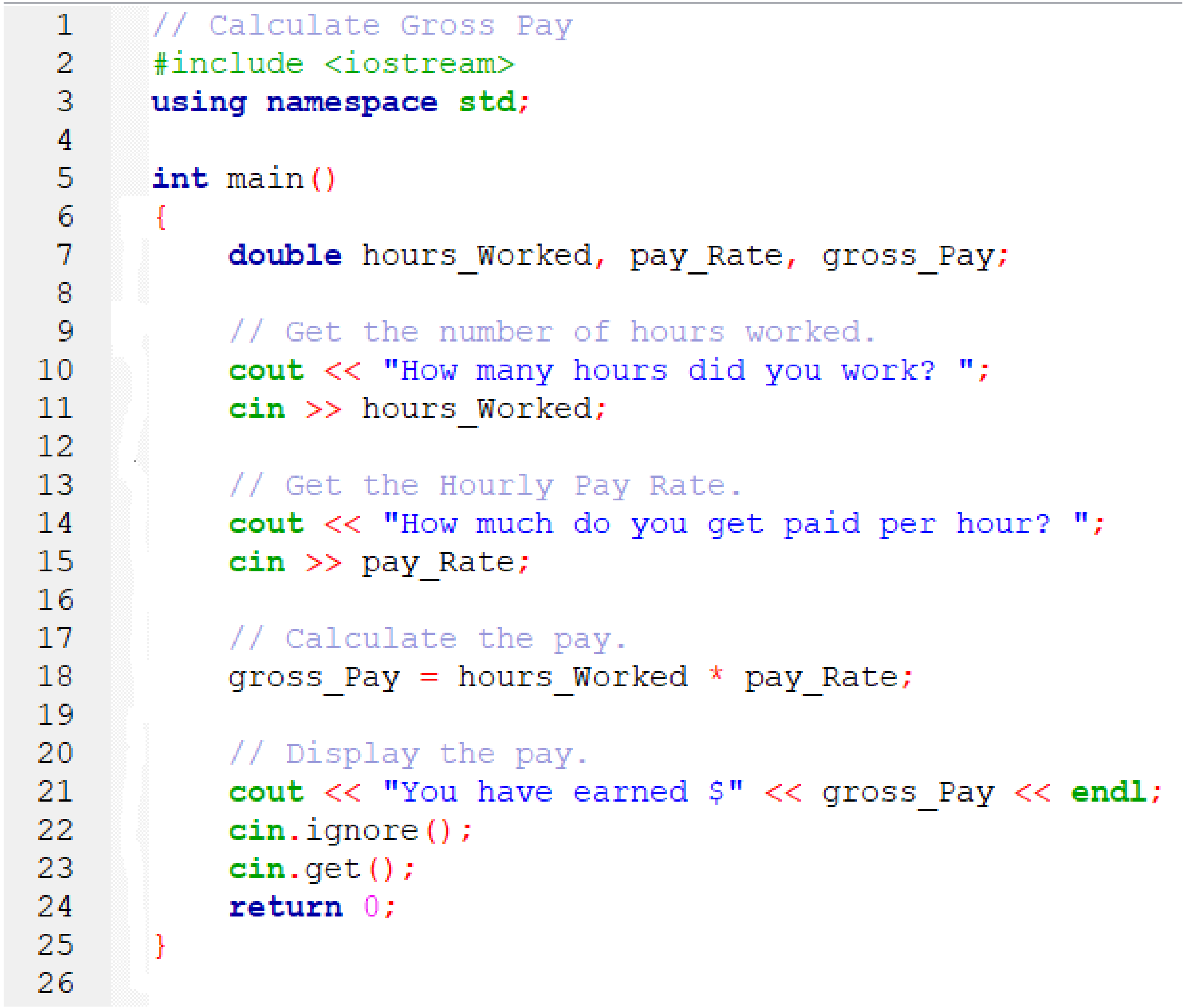
When you are done, call the Instructor over so you may get credit for your work.

# Part 3: Now code your program design using C++

- Lines 9-15 show the Inputting part of the program
- Lines 17-18 show the processing part of the program
- Lines 20-21 show the Outputting part of the program



```
 9      // Get the number of hours worked.
10      cout << "How many hours did you work? ";
11      cin >> hours_Worked;
12
13      // Get the Hourly Pay Rate.
14      cout << "How much do you get paid per hour? ";
15      cin >> pay_Rate;
16
17      // Calculate the pay.
18      gross_Pay = hours_Worked * pay_Rate;
19
20      // Display the pay.
21      cout << "You have earned $" << gross_Pay << endl;
```



```
1   // Calculate Gross Pay
2   #include <iostream>
3   using namespace std;
4
5   int main()
6   {
7       double hours_Worked, pay_Rate, gross_Pay;
8
9       // Get the number of hours worked.
10      cout << "How many hours did you work? ";
11      cin >> hours_Worked;
12
13      // Get the Hourly Pay Rate.
14      cout << "How much do you get paid per hour? ";
15      cin >> pay_Rate;
16
17      // Calculate the pay.
18      gross_Pay = hours_Worked * pay_Rate;
19
20      // Display the pay.
21      cout << "You have earned $" << gross_Pay << endl;
22      cin.ignore();
23      cin.get();
24      return 0;
25  }
26
```

Lines 1-6 show the infrastructure that starts the program.

Lines 22-23 show the infrastructure that ends the program.

This is what is called the source code.

1. Always Add Your Name to the Source Code, as a Comment, at the beginning of the program.

2. Save the program with the name: *YourName*-Lab01B.cpp (where *YourName* is Your own Name).

3. Just preceding the line: **return 0;** note the two lines:

```
cin.ignore();
cin.get();
```

(If you are using the CodeBlocks IDE, this will hold the program on the screen when you run the .exe file from the command line).

# From a High-Level Program to an Executable File

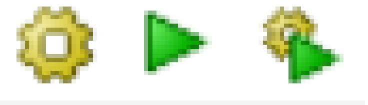
a) Create file containing the program with a text editor.

b) Run preprocessor to convert source file directives to source code program statements.

c) Run compiler to convert source program into machine instructions.

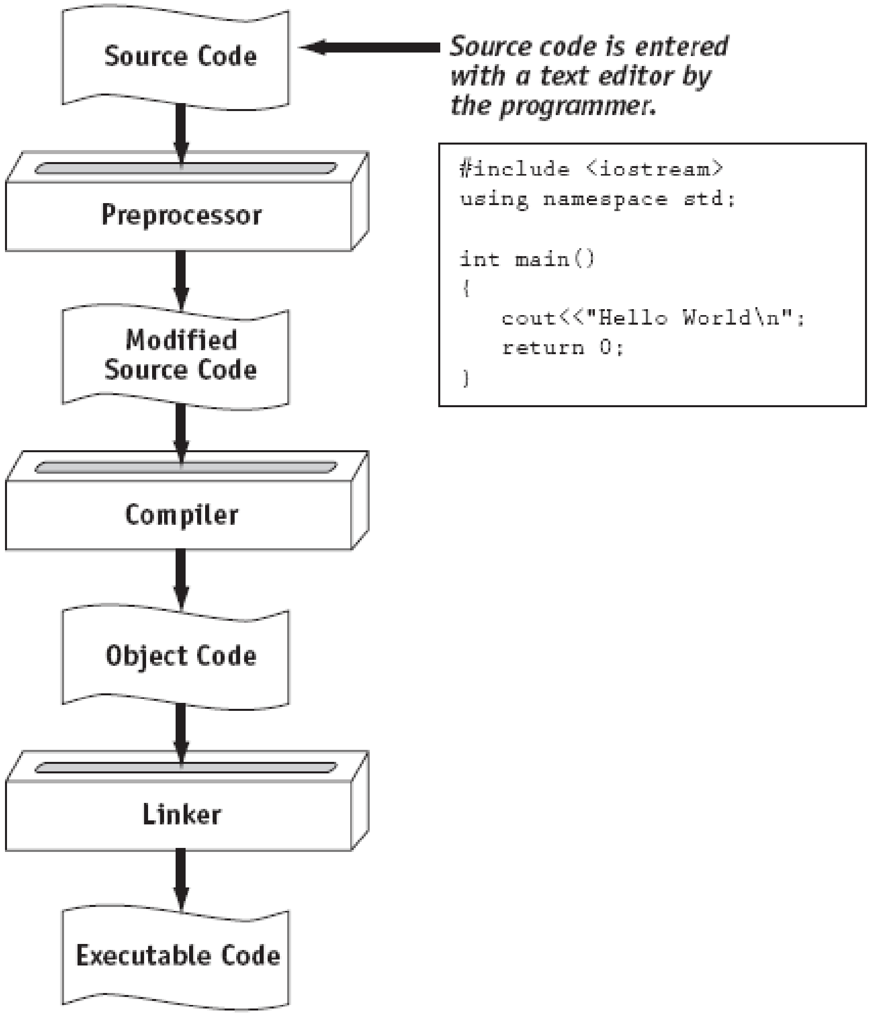
d) Run linker to connect hardware-specific code to machine instructions, producing an executable file.

- Steps b–d are often performed by a single command or button click.



- Errors detected at any step will prevent execution of following steps.

# From a High-Level Program to an Executable File



*Source code is entered with a text editor by the programmer.*

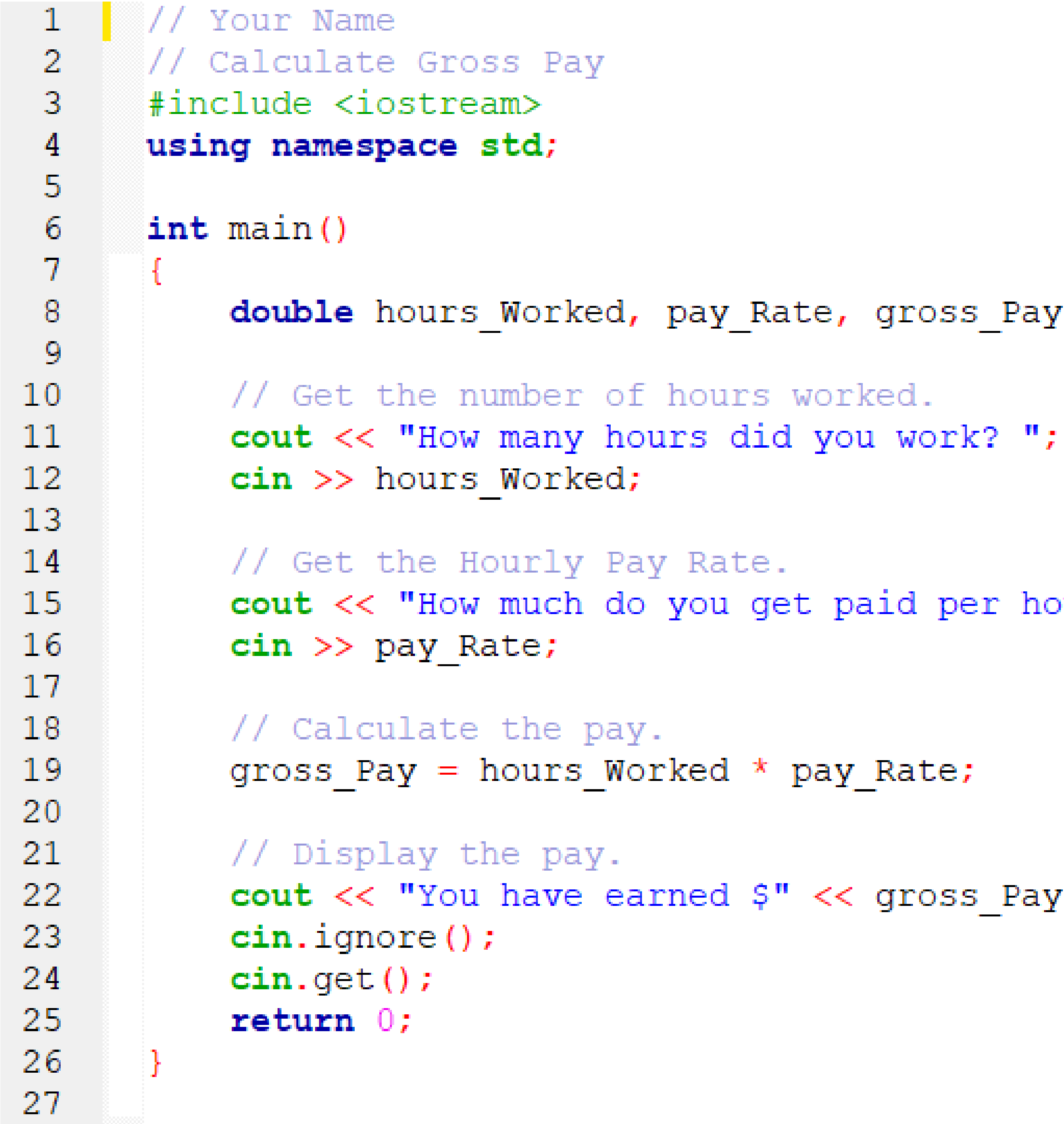
```
#include <iostream>
using namespace std;

int main()
{
   cout<<"Hello World\n";
   return 0;
}
```

# Part 4
# Now compile and run your program.

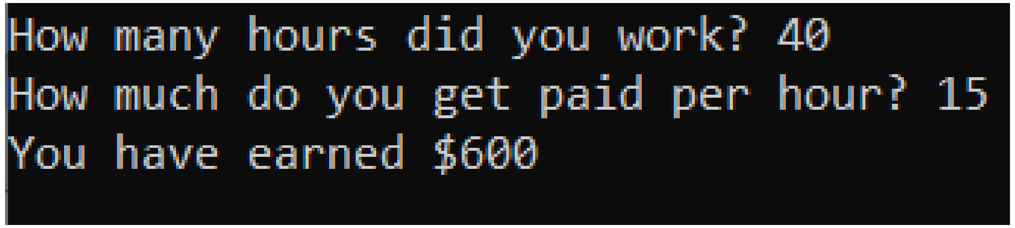
Call the instructor over so that you may get full credit for your work, and submit your .cpp and .exe files to Canvas.

Executing the Program,

This is what you should get!



```
// Your Name
// Calculate Gross Pay
#include <iostream>
using namespace std;

int main()
{
    double hours_Worked, pay_Rate, gross_Pay;

    // Get the number of hours worked.
    cout << "How many hours did you work? ";
    cin >> hours_Worked;

    // Get the Hourly Pay Rate.
    cout << "How much do you get paid per hour? ";
    cin >> pay_Rate;

    // Calculate the pay.
    gross_Pay = hours_Worked * pay_Rate;

    // Display the pay.
    cout << "You have earned $" << gross_Pay << endl;
    cin.ignore();
    cin.get();
    return 0;
}

```



```
How many hours did you work? 40
How much do you get paid per hour? 15
You have earned $600
```

When you have finished, and you are confident you have them right, submit your Part 1, Part 2, Part 3 and Part 4 of Lab01B into Canvas.