**Renee Thomas**

**CIS170C\_Lab01**

**1/12/14**

**Lab # CIS CIS170C-A1 (Part A)**

#include<iostream>

using namespace std;

int main(){

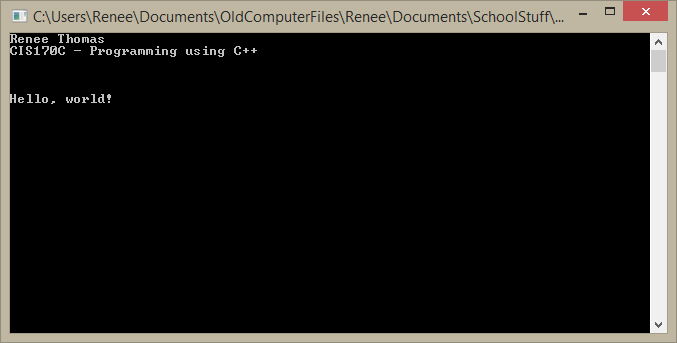
cout << "Renee Thomas" << endl;

cout << "CIS170C - Programming using C++\n";

cout << "\n\n\nHello, world!\n\n";

cin.ignore(2);

}



**Lab # CIS CIS170C-A1(Part B)**

// ---------------------------------------------------------------

// Programming Assignment: LAB1B

// Developer: Renee Thomas

// Date Written: 1/9/14

// Purpose: Ticket Calculation Program

// ---------------------------------------------------------------

// add libraries

#include <iostream>

using namespace std;

// add main function

void main()

{

//initialize variables as integers

int childTkts, adultTkts, totalTkts;

// add values to all the variables

childTkts = 3;

adultTkts = 2;

// Calculate total tickets by adding childTkts and adultTkts

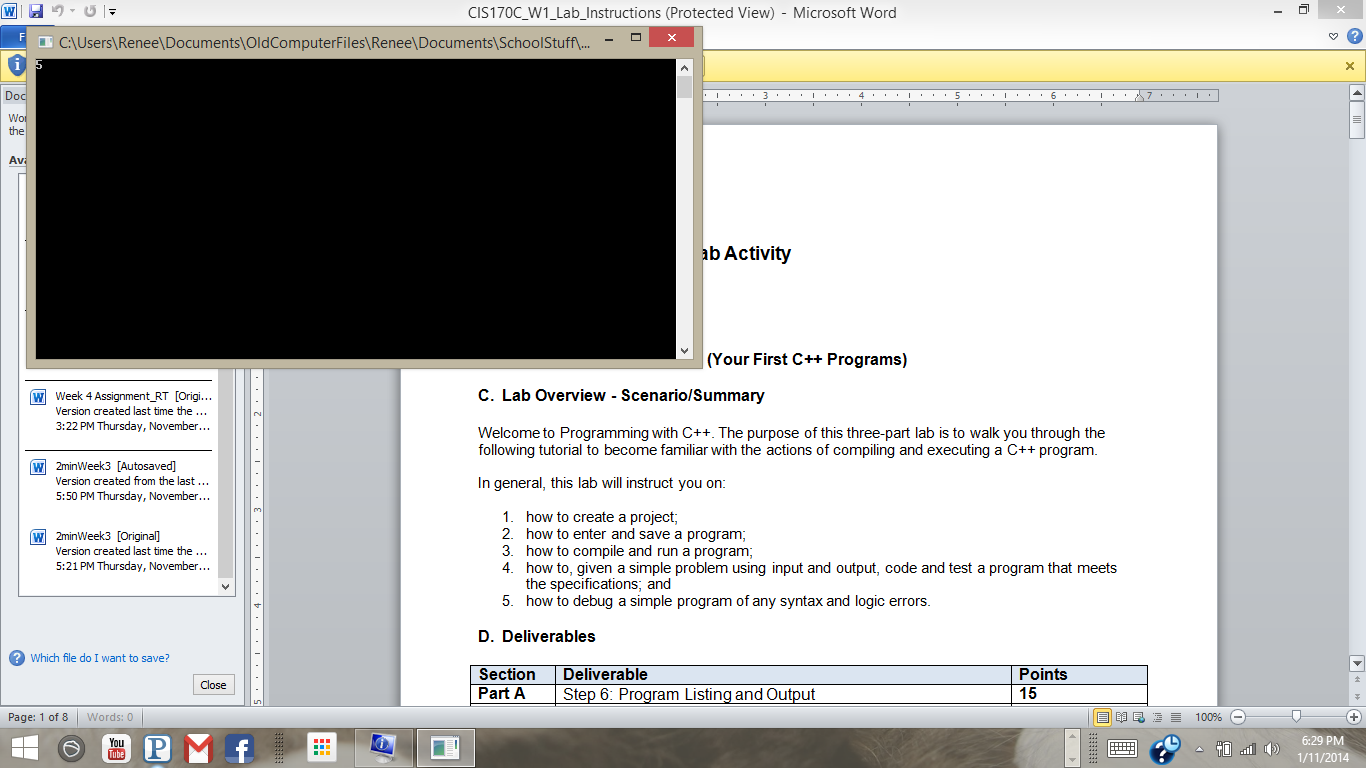
totalTkts = childTkts + adultTkts;

// output the total number of tickets to the console

cout << totalTkts << endl;

//add cin.ignore(2); to stop the console widow from closing

cin.ignore(2);



**Lab # CIS CIS170C-A1(Part C)**

// ---------------------------------------------------------------

// Programming Assignment: LAB1C

// Developer: Renee Thomas

// Date Written: 1/12/14

// Purpose: Payroll Program

// ---------------------------------------------------------------

// add libraries

#include <iostream>

#include <iomanip>

using namespace std;

// add main function

void main()

{

// assign contants

const double TAX = 0.18;

const double RETIREMENT = 0.10;

const double SSECURITY = 0.06;

const double EMPLOYEECUT = 0.07;

// assign variables

double weeklySales, grossPay, taxes, socialSecurity, retirement, totalDeductions, takeHomePay;

// get weekly sales amount from user

cout<< "Input Weekly Sales Amount: $";

// put user input of weekly sales amount into varable "weeklySales"

cin>> weeklySales;

// calculations

grossPay = weeklySales \* EMPLOYEECUT;

taxes = grossPay \* TAX;

socialSecurity = grossPay \* SSECURITY;

retirement = grossPay \* RETIREMENT;

totalDeductions = taxes + socialSecurity + retirement;

takeHomePay = grossPay - totalDeductions;

// outputs

cout<<fixed<<showpoint<<setprecision(2); // use to make two floating points

cout<<"\nTotal Sales: " << setw(12) << "$" << weeklySales << endl;

cout<<"Gross pay (7%): " << setw(10) << "$" << grossPay << endl;

cout<<"Federal tax paid: " << setw(9) << "$" << taxes << endl;

cout<<"Social security paid: " << setw(5) << "$" << socialSecurity << endl;

cout<<"Retirement contribution: " << setw(2) << "$" << retirement << endl;

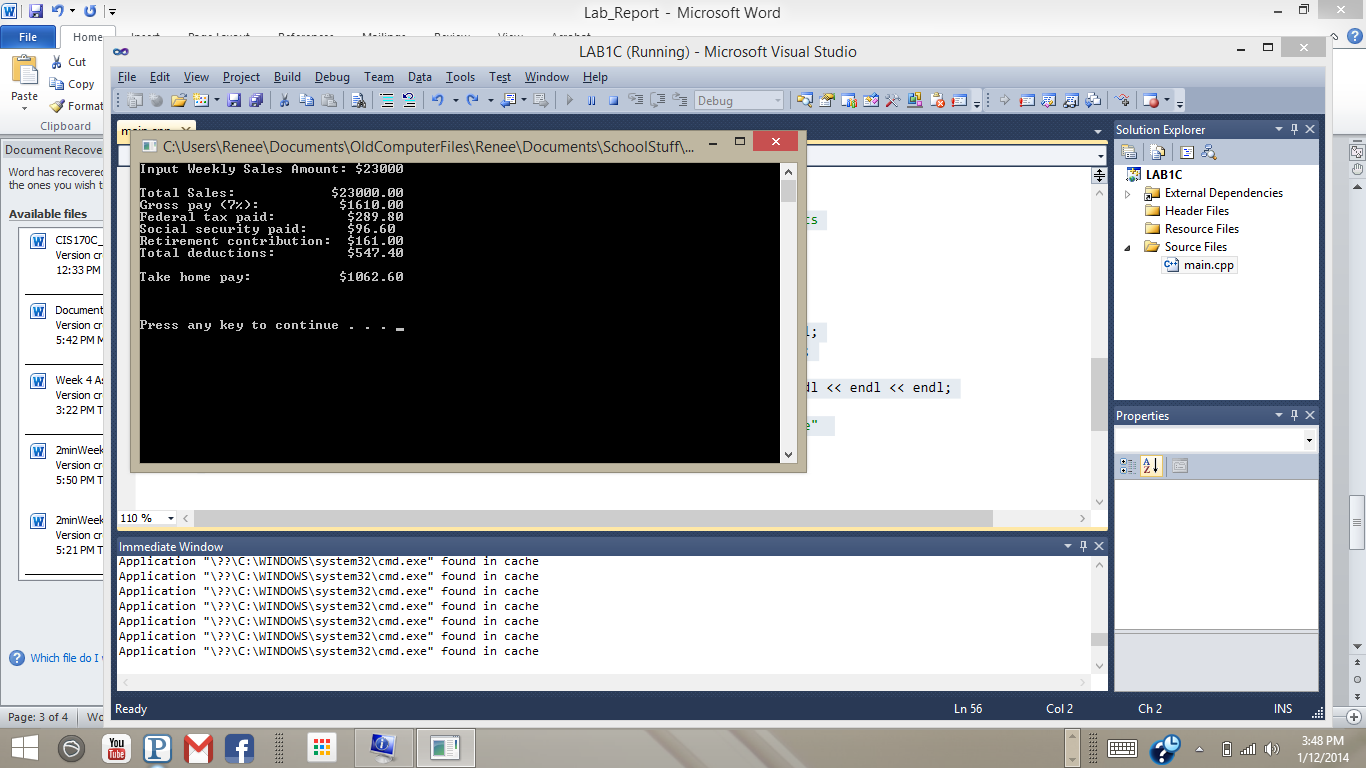
cout<<"Total deductions: " << setw(9) << "$" << totalDeductions << endl;

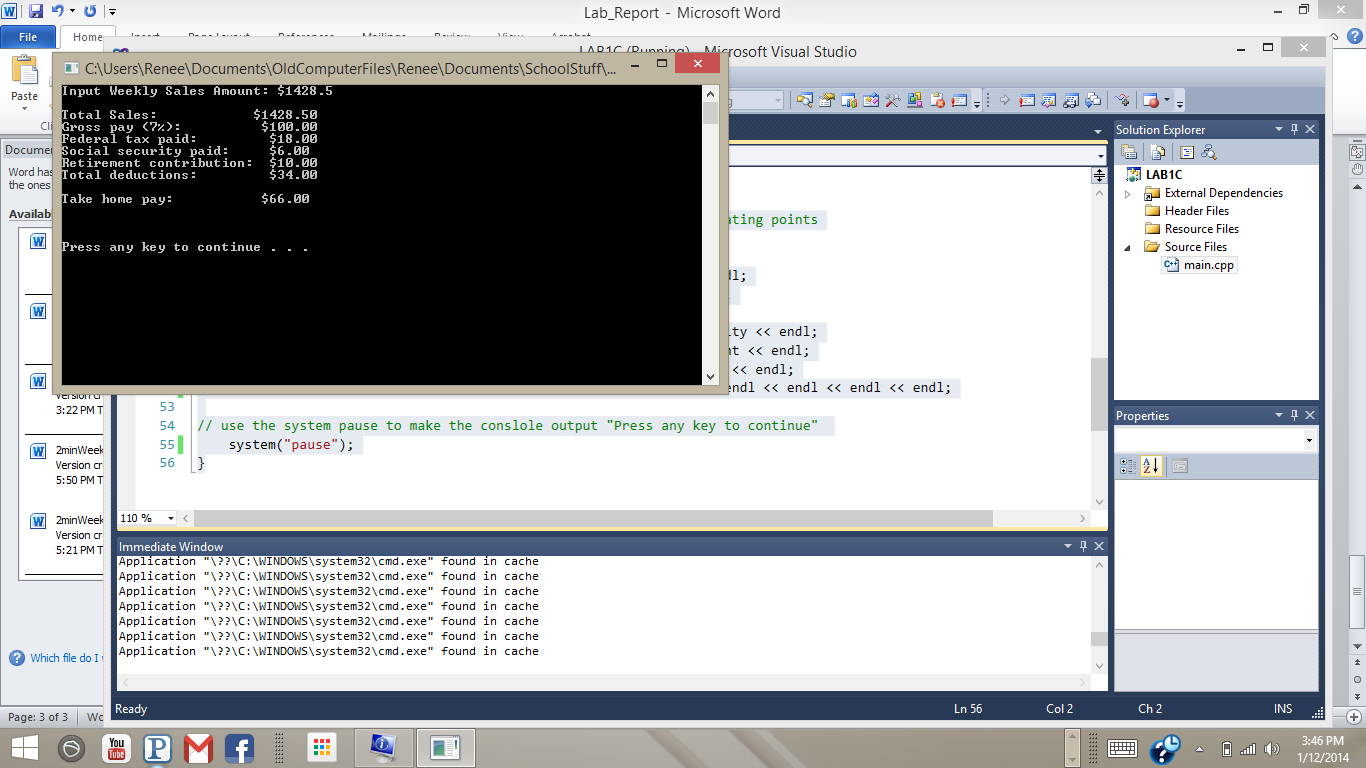
cout<<"\nTake home pay: " << setw(11) << "$" << takeHomePay << endl << endl << endl << endl;

// use the system pause to make the conslole output "Press any key to continue"

system("pause");

}





There has to be a better way to align the decimals… What if we were inputting a whole bunch of salaries from a database using an array with a loop? We wouldn’t be able to “babysit” the decimals one at a time like I did for the outcome. Also, as you can see, it works to line up the decimals for the 23000 input, but they don’t all align when I input 1428.5. I would love to see how to fix this. I saw something on the net about aligning everything to the right or left, but couldn’t understand how they were doing it. Thanks for your help!