Computer Science 31 Lecture 3  
  
#include <iostream>  
using namespace std;  
  
int main()  
{  
 // Gather hours worked and pay rate  
cout << “How many hours did you work? “;  
double hoursworked;  
cin >> hoursworked;  
  
cout << “what is your hourly rate of pay? “;  
double payrate;  
cin >> payrate;  
if (payrate < 9.00)  
 cout << “Ask for a raise!” << endl;  
  
 //compute and print amount earned and tax withheld  
  
cout.setf(ios::fixed);  
cout.precision(2);  
double amtEarned = hoursworked \* payrate;  
cout << “you earned $” << amtEarned << endl;  
*if (payrate >= 12.00) // TOO INEFFICIENT  
 cout << “$” << (0.10 \* amtEarned)  
 << “ will be withheld. “<< endl;  
else  
 cout << “$” << (0.05 \* amtEarned)  
 << “will be withheld.” << endl;*  
OR  
  
*if (payrate >= 12.00) // NOT QUITE RIGHT  
 double withholdingRate = 0.10; // not quite right!  
else  
 double withholdingRate = 0.05; // not quite right!  
cout << “$” << withholdingRate \* amtEarned  
 << “ will be withheld.” << endl;* // if you declare a variable in a branch of an If statement, you can only use that variable in that If statement  
  
OR   
  
double withholdingRate; // CORRECT  
*if (payrate >= 12.00)  
 double withholdingRate = 0.10;   
else  
 double withholdingRate = 0.05;   
 cout << “$” << withholdingRate \* amtEarned  
 << “ will be withheld.” << endl;* // if you declare a variable in a branch of an If statement, you can only use that variable in that If statement}  
  
hoursworked: 40  
payrate: 12.13  
amtEarned: 485.2  
  
declaration:  
double x; // create a new variable x, leaving it uninitialized  
double y = 3 \* 2 + 17 // create a new variable y, initializing it  
  
assignment statement:  
x = y – 12;  
x = 42;  
variable = expression // takes an already existing variable and stores values  
 // store the current value of expression in variable  
  
x = 17; // assign 17 to x  
x = y; // assign x to y  
 // set y to x  
 // x’s current value will be assigned to y  
if (x == 42) // is x equal to 42?  
  
  
  
m: 2  
n: 3  
  
int m = 2;  
int n = 3;  
…  
n = 4 \* m; // now n = 8  
  
m: 2  
n: 8  
  
…  
n = 2 \* n; // now n = 16  
  
m: 2  
n: 16  
  
  
another example  
a: 3  
b: ?  
  
int a = 3;  
int b = a + 5; // b is now 8  
…  
a: 3  
b: 8  
  
…  
a = 4; // b is still 8  
b = a + 40; // now b changes to 44  
a = 5; // b is still 44  
  
  
  
“magic numbers”  
  
int main()  
{  
const double PAY\_THRESHOLD = 12.00; // use capital letters to show that the values   
const double HIGH\_WITHHOLDING\_RATE = 0.10; are constant and simplify reading code  
const double LOW\_WITHHOLDING\_RATE = 0.05;  
…  
PAY\_THRESHOLD = 12.50 // error! Won’t compile! PAY\_THRESHOLD is set as constant  
double withholdingRate;   
*if (payrate >= PAY\_THRESHOLD)  
 double withholdingRate = HIGH\_WITHHOLDING\_RATE;   
else  
 double withholdingRate = LOW\_WITHHOLDING\_RATE;   
 cout << “$” << withholdingRate \* amtEarned  
 << “ will be withheld.” << endl;* }  
  
  
string citizenship;  
getline(cin,citizenship); // for example, US or Canada or Japan  
int age;  
cin >> age;  
  
  
if (citizenship == “US”)  
{   
 if (age >= 18)  
 cout << “You can vote in U.S. elections” << endl;  
 }  
else  
 cout << “You are not a U.S. citizen” << endl;  
  
  
if (citizenship == “US” || citizenship == “Canada”) someCondition || someCondition “or”  
 cout << “The country code is 1” << endl;  
  
 **WRONG: if (citizenship == “US” || == “Canada”)  
WRONG: if (citizenship == “US” || “Canada”)**  
int roll;  
…  
if (roll == 2 || roll == 3 || roll == 12)  
 cout << “You lose” << endl;  
  
**WRONG : if (roll == 2 || 3 || 12)**  
  
if (citizenship == “US” && age >= 18) someCondition && anotherCondition “and”  
 cout << “You can vote in U.S. elections” << endl;  
  
&& has higher precedence than ||  
for && , both must be TURE  
for ||, at least 1 must be TRUE  
  
if (citizenship == “US” || citizenship == “Canada” && age >= 21)  
…  
if ((citizenship == “US” || citizenship == “Canada”) && age >= 21)  
if (citizenship == “US” || (citizenship == “Canada” && age >= 21))  
  
  
if (some condition)  
 do something;  
else  
 int k = 20  
 do another thing  
 do yet another thing  
  
arithmetic expressions = \* / + -  
   
double a = (3+4) \* 2; // 14  
double b = 27/3 \* 3  
  
double c = 14.3  
cout << c/5.0; // writes 2.86  
cout << c/5 //writes 2.86  
  
int k = 14;  
cout << k/5.0; //writes 2.86  
cout << k/5 // 2 \*dividing 2 integers gives an answer where the fraction is dropped\*   
 int / int 🡺 int, drops fraction  
cout << k%5; // writes 4 (14 divided by 5 has a remainder of 4)  
  
double d = 14/5 //d is 2.0 (not 2.86, remember int / int)   
 14/5 🡺 2, 2 is used to initialize the double  
  
int f = 10  
int g = f\*f  
int h = 25/(g-100); // problem!! Division by 0  
  
int I;  
int j = 2 \* I;  
cout << j; // problem!! I is not initialized  
  
int k = 1000;  
int m = k \* k \* k;  
int n = k \* m; // problem!! Integer overflow  
  
double p = m;  
double n = k \* p;   
  
  
(something; something; something; ) a compound statement / a block  
  
if (some condition)  
 do something;  
else  
{  
 doOneThing;  
 doAnotherThing  
 doYetAnotherThing  
}  
***or***  
if (some condition)  
 do something;  
  
  
double x;  
cin >> x;  
if (x < 0)  
 cout << -x;  
else  
 cout << x;  
  
cout << “What is your name? “;  
string name;  
getline(cin, name);  
if (name == “”)  
 cout << “You didn’t type a name!”;  
else  
 cout << “Hello, “ << name;

What is your name? Sir Robin  
What is your quest? To seek the Holy Grail  
Hello, brave Sir Robin!  
You want To seek the Holy Grail  
  
===================================  
  
#include <iostream>  
#include <string>  
using namespace std;  
  
int main()  
{  
 cout << “What is your name? “;  
string personsName;  
getline(cin, personsName); // reads a line into personsName  
  
cout << “What is your quest? “;  
string quest;  
getline(cin, quest); // reads a line into quest  
  
cout << “Hello, brave “ << personsName << “!” << endl;  
cout << “You want “ << quest << endl;  
}  
  
personsName: Sir Robin  
quest: To seek the Holy Grail  
  
cin >> personsName; // reads just one word  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
double: about 10 to the -308 to 10 to the 308  
 about 15 significant digits  
 positive or negative  
  
int: about -2 billion up to +2 billion  
  
double x; // declares x, and leaves it uninitialized  
double y = 47; // declares y, and gives it an initial value  
  
  
  
  
  
  
  
  
  
***Free Drop in Tutoring  
Tau Beta Pi Tutoring Program  
Boelter 6266  
Mon – Fri (10am – 6pm)  
Tutoring schedule at tbp.seas.ucla.edu  
or like us at facebook @ “Tau Beta Pi Tutoring at UCLA”***