10/14/2019

int main () {

const double PAYRATE\_THRESHOLD = 14.00;

const double HIGH\_WITHHOLDING\_RATE = 0.10;

const double LOW\_WITHHOLDING\_RATE = 0.05;

cout << “How many hours did you work? “;

double hoursWorked;

cin >> hoursWorked;

cout << “What is your hourly rate of pay? ‘;

double payrate;

cin >> payrate;

double amtEarned = hoursWorked \* payrate;

cout.setf(ios::fixed);

cout.precision(2);

cout << “You earned $” << amtEarned << endl;

double withholdingRate;

if (payrate >= PAYRATE\_THRESHOLD)

withholdingRate = HIGH\_WITHHOLDING\_RATE;

else

withholdingRate = LOW\_WITHHOLDING\_RATE;

cout << “$” << (withholdingRate \* amtEarned) << “ will be withheld.” << endl;

}

Use variables for specific type of number -> convenience when it changes

Ex.

int m = 2;

int n = 3;

…

n = 4 \* m; //n = 8

n = 2 \* n; // n = 16

int a = 3;

int b = a + 5; //b = 8

…

a = 4; //b = 8

b = a + 40; //b = 44

a = 5; // b = 44

double x = 4;

int i = 3.8;

string citizenship;

int 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;

&& and

|| or (either one or both: true -> true, only false when both are false)

if (citizenship == “ US” && age >= 18)…

if ((a/b + c/d) > 10) { //err when divided by zero

}

if (b != 0 && d != 0 && (a/b + c/d) > 10) {

{

Backward reasoning

Ex.

if (citizenship == “US” || citizenship == “Canada”)

if (citizenship == “US” || “Canada”)

if (age == 18 || 19 || 20) -> wrong -> always be true

if (age <= age <= 20) -> always be true

if ((citizenship == “US” || citizenship == “Canada”) && age >= 18)

&& higher precedence than ||

Good to group together even if the precedence works

if (citizenship != “US” || citizenship != “Canada”) -> always be true – do not use != for both in ||

if (citizenship != “US” && citizenship != “Canada”)

De Morgan’s Laws:

not (A and B) -> (not A) or (not B)

not (A or B) -> (not A) and (not B)

if (citizenship != “US” || age < 18) {

cout << “Wou cannot vote in U.S. elections” << endl;

}

Be careful of switch >= or <=

Remove = when doing opposite

if (income < 30000)

cout << “Low”;

else {

if (income < 100000)

cout << “Middle”;

else {

if (income < 500000)

cout << “High”;

else

cout << “Very High”;

}

}

Result of assignment operator -> always of the new value of the left side -> false

able to complier