C++ Basics

Announcements

Lab 1 this week!

Homework posted Wednesday (late)

Types of errors

Syntax error - code will not compile
 e.g. cout(“hi”);

Runtime error - code crashes after starting
 e.g. (0 input to runTimeError.cpp)

Logic error - code runs but doesn't return the correct answer
 (see: logicError.cpp)

Syntax

Syntax is a fancy word for the "grammar" of programming languages

The basic English syntax is:
(subject) (verb) (noun)
"I eat bananas" not "Bananas I eat"

The computer is VERY picky (and stubborn) about grammar, and will not understand you unless you are absolutely correct!

Comments

Comments are ignored pieces of code (computer will pretend they do not exist)

// denotes a single line that is commented
// (everything before hitting enter)

/* denotes the beginning of a comment and the end of a comment is denoted by */

Avoid errors

To remove your program of bugs,
you should try to test your program on
a wide range of inputs

Typically it is useful to start with a small piece of code that works and build up rather than trying to program everything and then debug for hours

Variables

Variables are objects in program

To use variables two things must be done:

- Declaration
- Initialization

See: uninitialized.cpp

Example if you forget to initialize:

```
I am 0 inches tall.
I am -1094369310 inches tall.
```

Variables

$$\texttt{int x, y, z;} \longleftarrow \text{Declaration}$$

$$\left.\begin{array}{l} \texttt{x = 2;} \\ \texttt{y = 3;} \\ \texttt{z = 4;} \end{array}\right\} \text{Initialization}$$

Same as:

```
int x=2, y=3, z=4;
```

Variables can be declared anywhere (preferably at start)

Assignment operator

= is the assignment operator

The object to the right of the equals sign is stored into the object in the left

```
int x, y;
y = 2;
x = y+2;
```

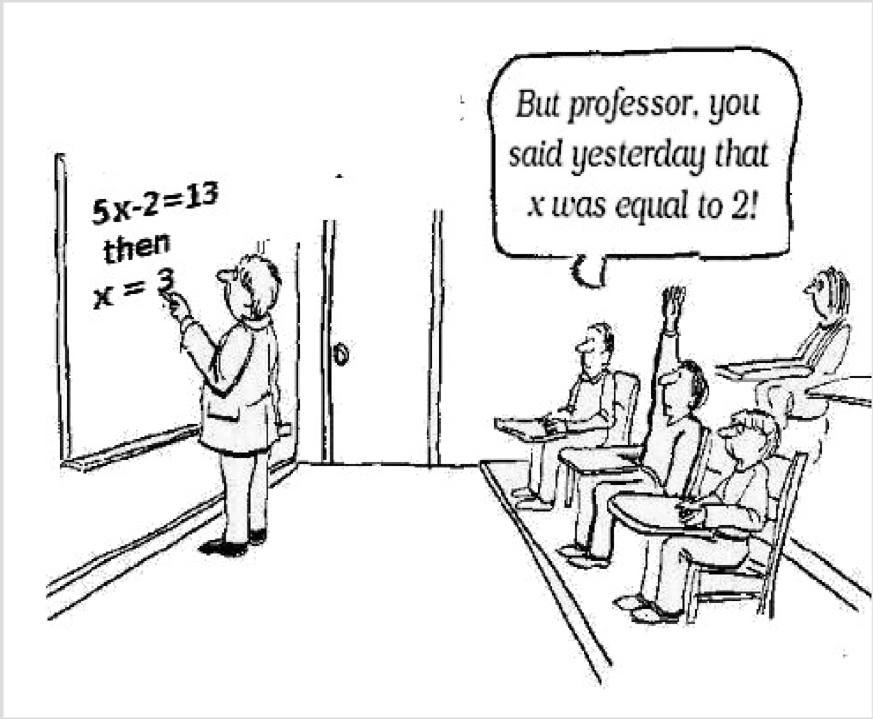
See: assignmentOp.cpp

Assignment operator

= is NOT a mathematic equals

```
x=3;
x=4;  // computer is happy!
```

This does not mean 3=4



Assignment operator

To the left of = needs to be a valid object that can store the type of data on the right

```
int x;
x=2.6; // unhappy, 2.6 is not an integer

x+2 = 6; // x+2 not an object

2 = x; // 2 is a constant, cannot store x
```

Assignment operator

What does this code do?

```
int x = 2, y = 3;
y=x;
x=y;
```

What was the intention of this code?

Increment operators

What does this code do?

```
int x = 2;
x=x+1;
```

Increment operators

What does this code do?

```
int x = 2;
x=x+1;
```

Same as:

```
x+=1;
```

or

```
x++;
```

Increment operators

Two types of increment operators:

```
x++;  // increments after command
    vs
++x;  // increments before command
```

Complex assignments

The following format is general for common operations:

variable (operator)= expression
variable = variable (operator) expression

Examples:

x+=2	$\Longleftrightarrow$	x = x + 2
x*=y+2		x = x * (y + 2)

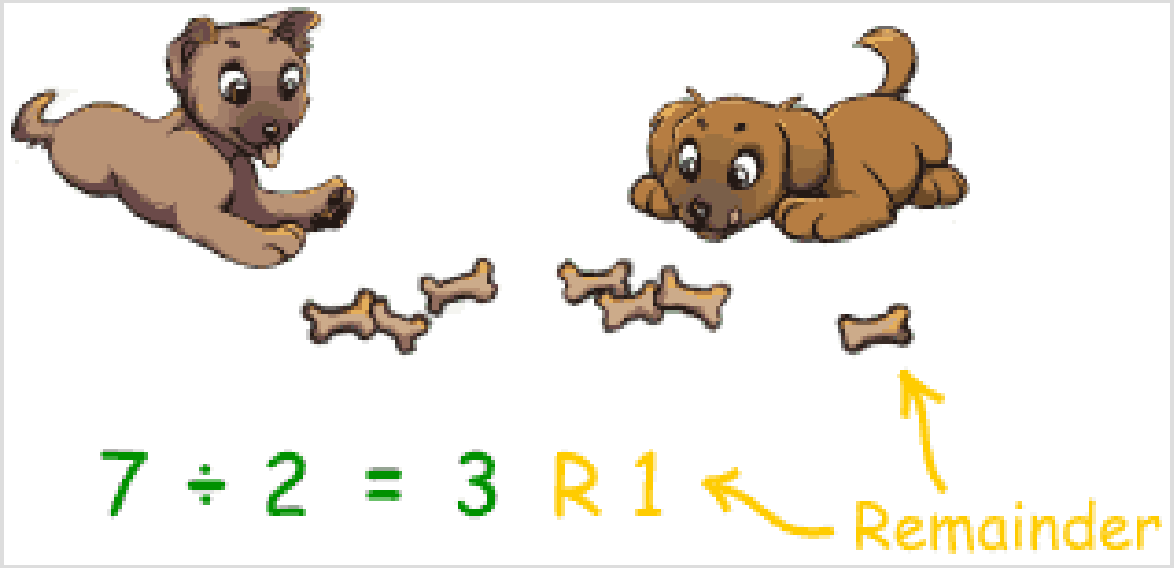
Order of operations

Order of precedence (higher operations first):
-, +, ++, -- and ! (unary operators)
*, / and % (binary operators)
+ and - (binary operators)

% is remainder operator, which you might not have used much but is awesome!

Order of operations

If you are dealing with whole numbers, % can tell you how many “items” do not divide equally



Order of operations

Binary operators need two arguments
Examples:
2+3, 5/2 and 6%2

Unary operators require only one argument:
Examples: (see binaryVsUnaryOps.cpp)
+x, x++, !x

(! is the logical inversion operator for bool)

Order of operations

When multiple operations have the same precedence level:

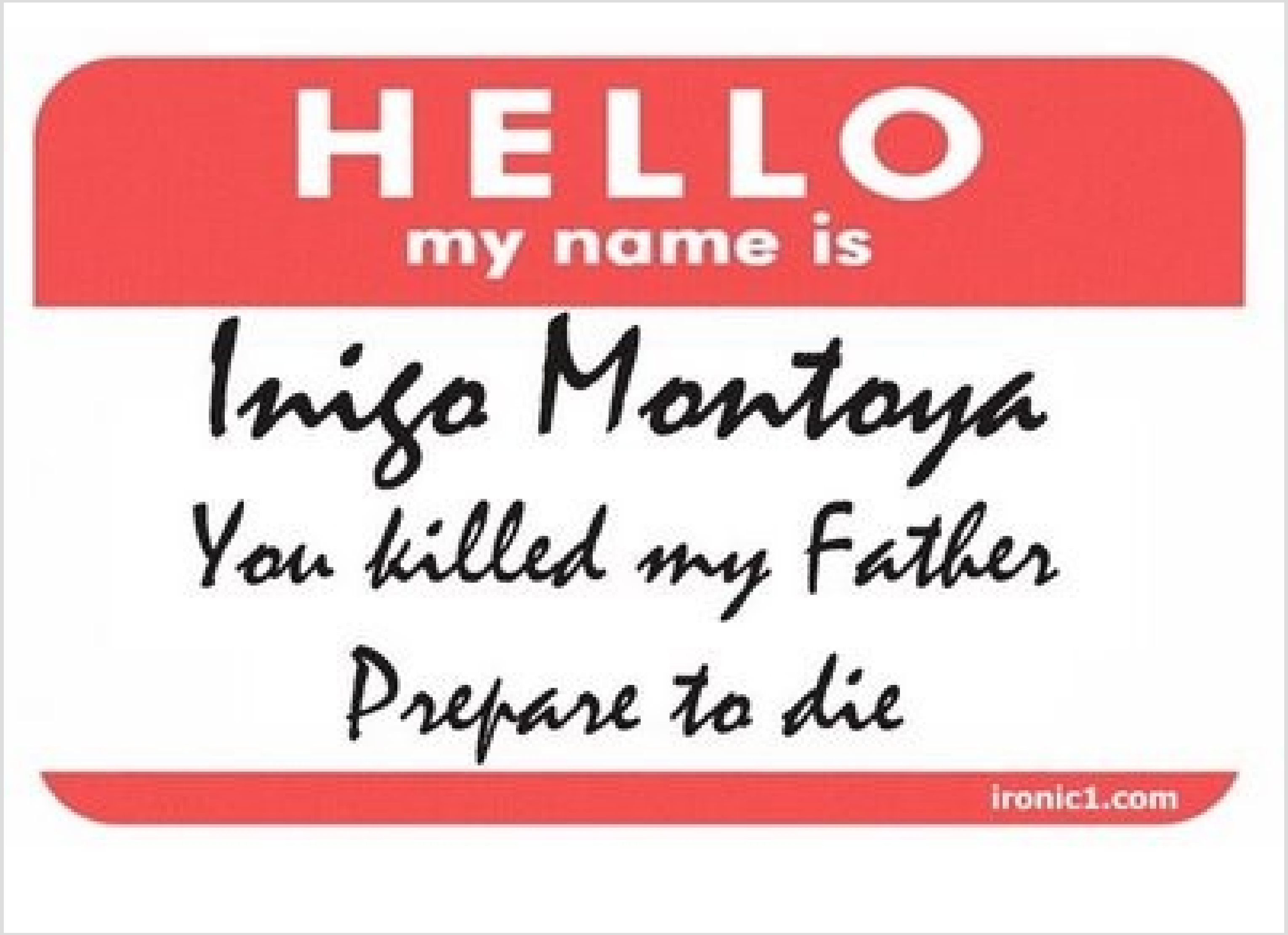
Binary operations go from left to right

7 + 3 + 4

Unary operations go right to left

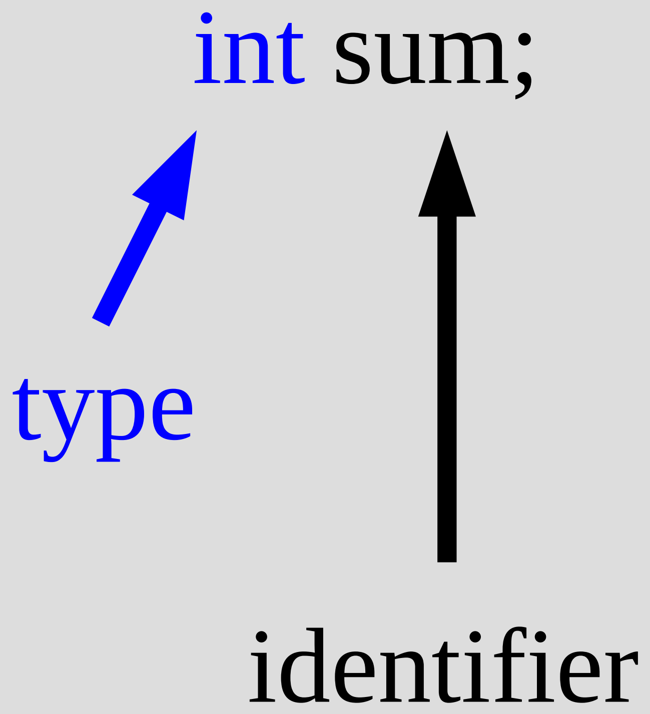
- -7 (double negative)

Identifiers



Identifiers

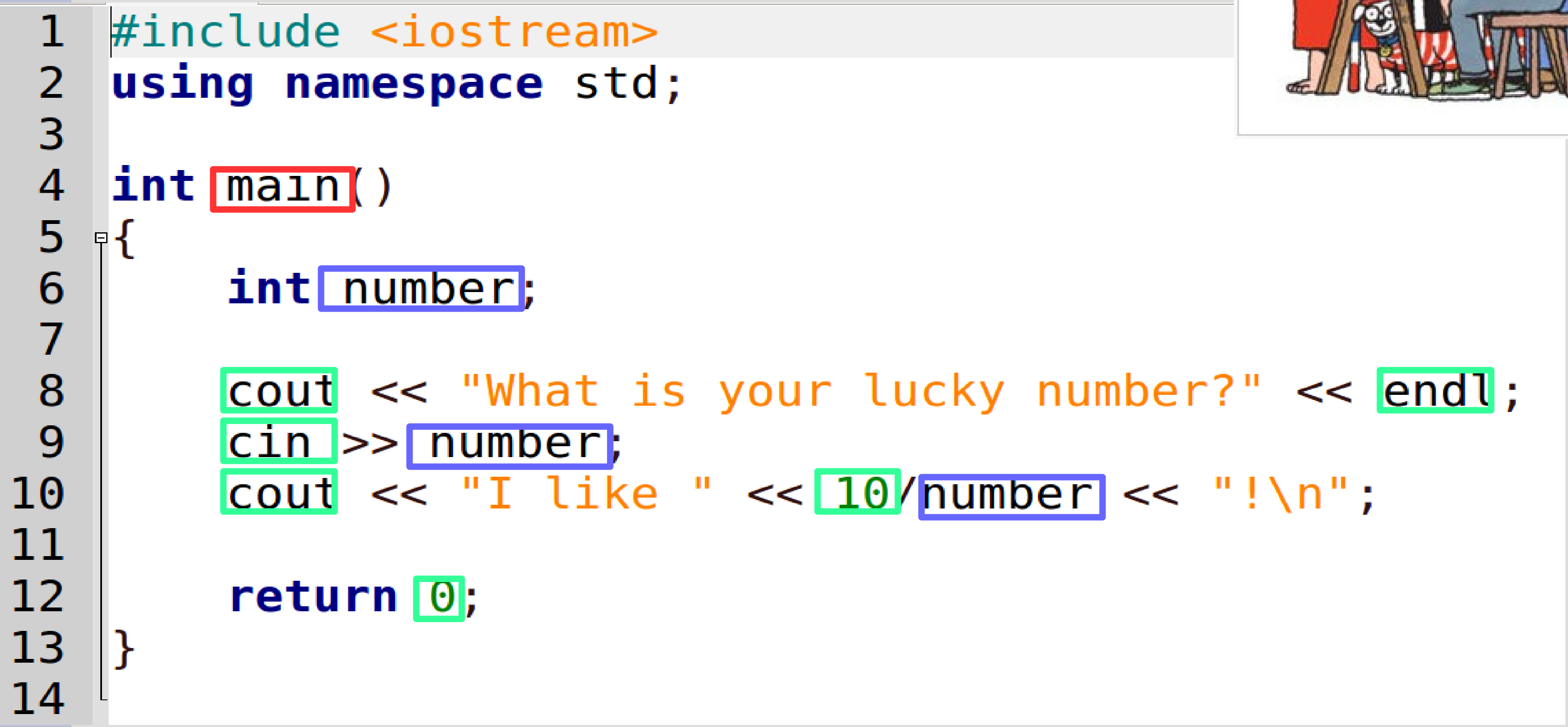
An identifier is the name of a variable (or object, class, method, etc.)



- Case sensitive
- Must use only letters, numbers or _
- Cannot start with a number
- (Some reserved identifiers, like main)

Identifiers

Already did this in week 1!
See: RuntimeError.cpp



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

int main()
{
     int number;

     cout << "What is your lucky number?" << endl;
     cin >> number;
     cout << "I like " << 10/number << "!\n";

     return 0;
}

```

Identifiers

Which identifiers are valid?

1) james parker
2) BoByBoY
3) x3
4) 3x
5) x_________
6) _________x
7) Home.Class
8) Five%
9) x-1

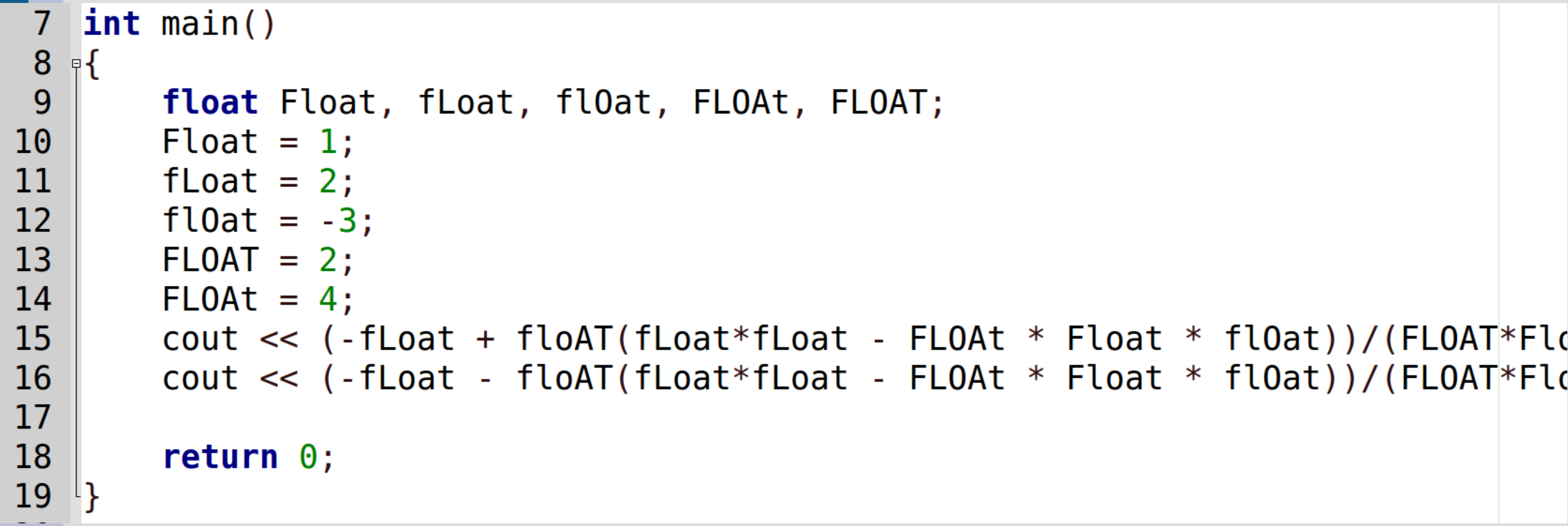
Identifiers

Which identifiers are valid?

1) ~~james parker~~
2) BoByBoY
3) x3
4) ~~3x~~
5) x__________
6) __________x
7) ~~Home.Class~~
8) ~~Five%~~
9) ~~x-1~~

Identifiers

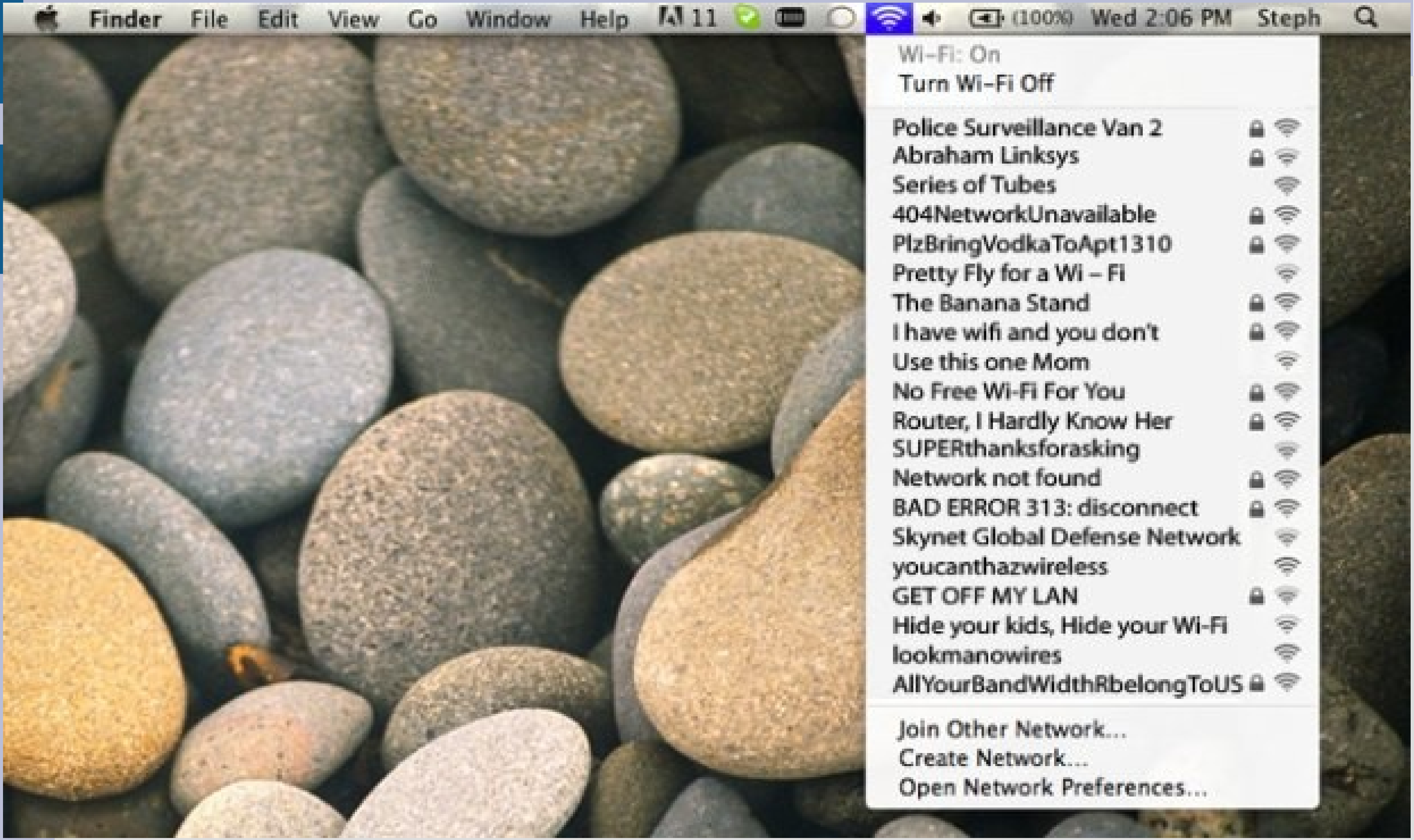
(See: float.cpp)



```
int main()
{
    float Float, fLoat, fl0at, FLOAt, FLOAT;
    Float = 1;
    fLoat = 2;
    fl0at = -3;
    FLOAT = 2;
    FLOAt = 4;
    cout << (-fLoat + floAT(fLoat*fLoat - FLOAt * Float * fl0at))/(FLOAT*Fl
    cout << (-fLoat - floAT(fLoat*fLoat - FLOAt * Float * fl0at))/(FLOAT*Fl

    return 0;
}
```

Identifiers



Wi–Fi: On
Turn Wi–Fi Off

Police Surveillance Van 2
Abraham Linksys
Series of Tubes
404NetworkUnavailable
PlzBringVodkaToApt1310
Pretty Fly for a Wi – Fi
The Banana Stand
I have wifi and you don't
Use this one Mom
No Free Wi-Fi For You
Router, I Hardly Know Her
SUPERthanksforasking
Network not found
BAD ERROR 313: disconnect
Skynet Global Defense Network
youcanthazwireless
GET OFF MY LAN
Hide your kids, Hide your Wi-Fi
lookmanowires
AllYourBandWidthRbelongToUS

Join Other Network...
Create Network...
Open Network Preferences...