

CS 161

Intro to CS I

Variables and Input

Odds and Ends



Oregon State University
College of Engineering

- Sign up for Assignment 1 Demo
- Peerceptiv Peer Reviews
 - Peer Reviews due Thursday, 11:59pm
 - Back Evaluations due Sunday, 11:59pm
- Assignment 2 posted
- Questions?

— KEC 1174
3 total



Reflections

- Why do we care about limits or sizeof()?
- Why did we have to typecast pow()?

pow(2,64)

exponent

floating-pt



More C++



- Programming Style: please read your class style guide
 - Program Header/Description
 - Placement of { }
 - Indentation: spaces vs. tabs

Be consistent

- String Literal in quotations, ""
 - Not single quotes!
 - INCORRECT: `std::cout << 'Hello World';`
 - Do not span more than one line!
 - INCORRECT: `std::cout << "Hello
World";`

More C++

*endl flushes buffer
prints newline*



Oregon State University
College of Engineering

\t \tab

- Escape Sequences

- Display special characters

- Use backslash, \, before special character to print

- Examples:

special meaning

```
std::cout << "Hello World\n";
```

newline Kinda like endl

*error
cout*

- Refer online for common escape sequences:

<http://en.cppreference.com/w/cpp/language/escape>

Diff than cplusplus.com



Data Type

What are you sending the function?

- What is data?
 - Information
 - Ex: `std::cout << "Hello World!" << std::endl;`
 - Literals
 - 23, 79.5, "Hello", etc.

- What is a data type?
 - Description of the kind of information
 - ~~Primitive Data~~
 - User Created – (we will cover later)

use a string

162



C++ Primitive Types

- char, double, float, int, long, short, bool
 - Fundamental
 - **short/int/long**: whole numbers, e.g. 45, -89, 0
 - **float/double**: real numbers, e.g. 2.612, -30.5, 2.3e5
 - **char**: characters, e.g. 'A', '&', 'x', '\\'
 - Signed by default, need to preface with unsigned keyword
 - **unsigned int**
 - **unsigned float**
 - **unsigned char**



Pieces of an Expression

- **Operators:** indicate operation

- Add +
- Subtract -
- Multiply *
- Divide /
- Remainder %

$10/5 \rightarrow 1$

$\text{Pow}(2, 64)$
operand

operator
operand

- **Operands:** values in the expression
- **Evaluation:** process of obtaining results from operations on operands



Precedence

- What is precedence?
 - Binding power of operator
- How do we override precedence?
 - Parenthesis!
- Examples:

1st – (*, /, %) vs. (+, -) *2nd*

$$12 * 4 + 6 * 10 \text{ vs. } ((12 * 4) + 6) * 10$$

1st
2nd
3rd



Arithmetic

- **Integer Arithmetic**

```
std::cout << 3/8; /*prints 0*/
```

```
std::cout << 34/5; /*prints 6*/
```

- **Floating Point Arithmetic**

```
std::cout << 34.0/5.0; /*prints 6.8*/
```

```
std::cout << 3.0/8; /*prints .375*/
```

```
std::cout << 3/8.0; /*prints .375*/
```



Type Casting

- **Casting**

```
std::cout << 34 / (int) 5.0; /*prints 6*/
```

```
std::cout << (int) (34 / 5.0); /*prints 6*/
```

```
std::cout << (float) 34 / 5; /*prints 6.8*/
```

- **What is wrong with these?**

```
std::cout << (int) 34 / 5.0; /*prints 6.8*/
```

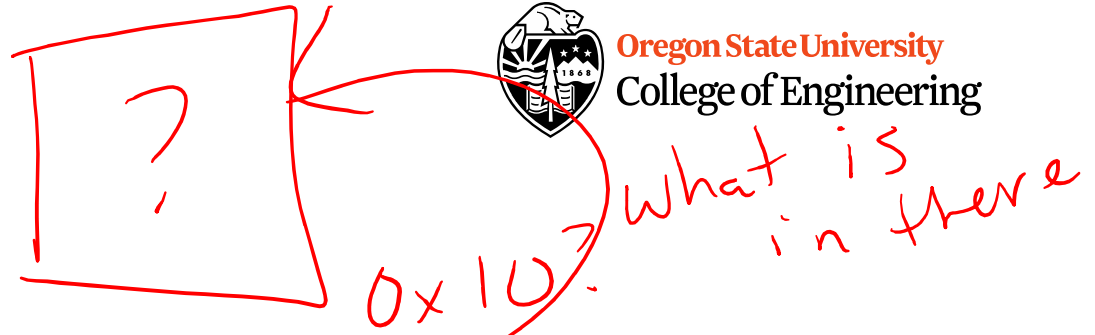
```
std::cout << (float) (34/5); /*prints 6.0*/
```

6

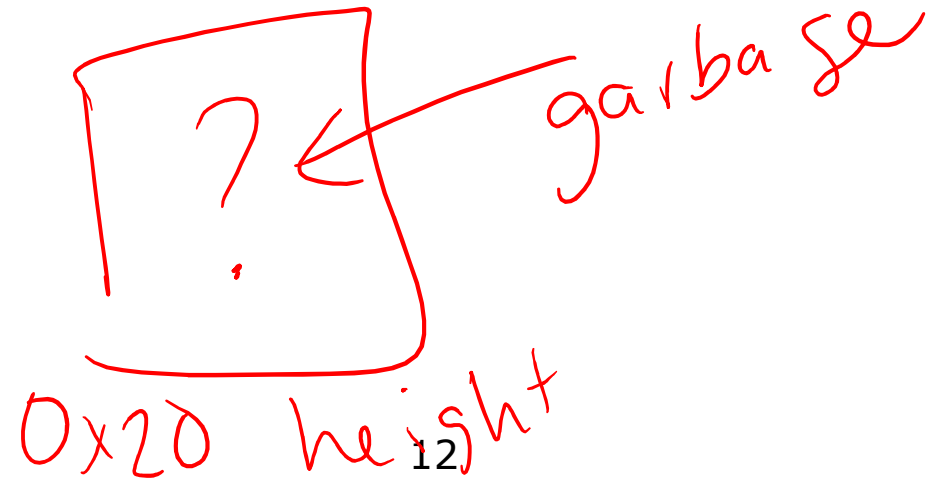
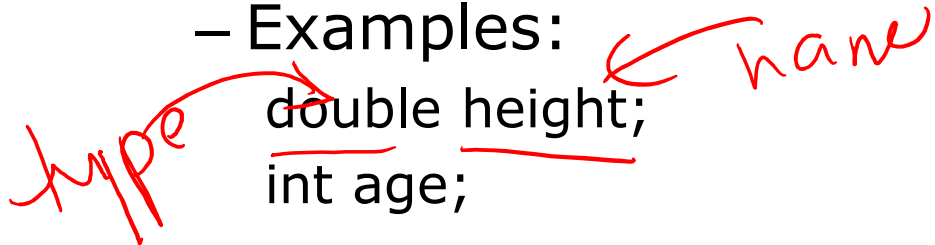
Variables



- What is a variable?
 - Memory location with name and type to store value
- What is a declaration?
 - Statement requesting variable w/ name and type



– Examples:
double height;
int age;





Variables/Identifiers

- Identifier: name given to item in program
 - Ex. Variables and Functions
 - Start with letter
 - Letters include: upper-case, lower-case, underscore (_)
 - Followed by sequence of letters and digits
- **Good examples:** hiThere, two_plus_two, _hello
- **Bad examples:** 5dogs, hi-there, hello there
- Can't Use Keywords:
<http://en.cppreference.com/w/cpp/keyword>

Variables

lvalue = rvalue
store *fetch*



- How do we get a value in the variable?
 - Assignment Statement

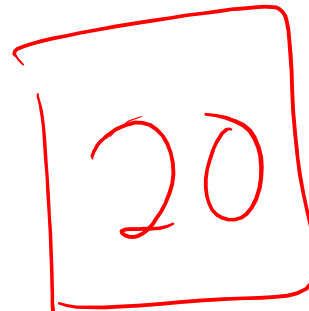
~~int~~ age;
age = 20;

Or

int age = 20;

– = IS NOT equal to!!!!

- “gets” or “is assigned”



0x20 age

*assignment
op.*



Printing Variables/Reading Into Variables

- C++: **cout**

- Example:

```
std::cout << "The integer value is: " << value;
```

- What about the newline?

variable to print

cout is opposite "

- C++: **cin**

- Example:

```
std::cin >> value;
```

easy to remember that cin is pushing info (>>) into the variable