

# **CS 161**

## **Intro to CS I**

Finish Loops/Begin C++ Strings

# Odds and Ends



Oregon State University  
College of Engineering

- Peer Reviews on design due Thursday.
- Exam I next Wednesday, 2/6/2019

```
2. ENGR
Re-attach Fullscreen Stay on top Duplicate
1 #include <iostream>
2 #include <string>
3
4 using namespace std;
5
6 int main() {
7     int i;
8     float f;
9     string s;
10
11     cout << "enter a integer: ";
12     cin >> i;
13     cout << "i: " << i << endl;
14
15     cout << "enter a float: ";
16     cin >> f;
17     cout << "f: " << f << endl;
18
19     cout << "enter your name: ";
20     //cin >> s;
21     getline(cin, s); //move past the newline in the buffer leftover from the float
22     getline(cin, s); //now read a line
23     cout << "s: " << s << endl;
24     cout << "length: " << s.length() << endl;
25     cout << "first letter in name: " << s.at(0) << endl;
26
27
28     return 0;
29 }
-- INSERT --
```

iversity  
neering

Solu

```
2. ENGR
Re-attach Fullscreen Stay on top Duplicate
1 #include <iostream>
2 #include <string>
3 #include <stdlib.h>
4
5 using namespace std;
6
7 int main() {
8     int i;
9     float f;
10    string s;
11    bool error;
12    do {
13        error=false;
14        cout << "enter an integer : " << endl;
15        getline(cin,s);
16
17        for(int i=0; i<s.size(); i++) {
18            if(!(s.at(i) >= '0' && s.at(i) <='9')) {
19                cout << "error!" << endl;
20                error=true;
21            }
22        }
23    }while(error);
24
25    i=atoi(s.c_str());
26    cout << i << endl;
27
28    cout << "enter a float: " << endl;
29    cin >> f;
30
31    return 0;
32 }
2 more lines; before #9 3 seconds ago 29,7 All
```

University  
Engineering

# Programming Errors

- Syntax errors
  - Misuse of C++ language
  - How are they caught?
- Logic errors
  - Doesn't perform task correctly (aka. bugs)
  - How are they caught?
- Runtime errors
  - Stops your program from running
  - How are they caught?

# Syntax Error Examples



- Missing main function
- Use of identifier not declared
- Misspelled Words
- Forget a Semicolon
- Forget Required Keyword
- Missing quote, curly brace, and parenthesis
- Use of single quotes instead of double

# Logic Error Examples



- Poorly written programs
  - Add instead of subtract (incorrect operation)
  - Using last two digits for date
  - Same error message for different errors
  - Program that never ends
  - Add one to the largest integer (could be syntax)

# Runtime Error Examples

- Segmentation fault or Core dump
  - Read a file that doesn't exist
  - Go outside of memory bounds
  - Run out of memory
  - Divide by variable that is zero



# Debugging Errors



- Syntax:
  - **READ compiler errors** (pay attention to line #)
  - Use **google** to search for error
- Logic/Runtime
  - Use **std::cout** to find where the code is breaking
    - **Print variable values**
    - **Print indicator messages**
  - **Trace** through the code
  - **Comment** out code

# Decomposition



- Divide Problem (task) Into Subtasks
  - Procedural Decomposition
  - Examples: cooking, cleaning, etc.
- Incremental Programming
  - Iterative Enhancement (Stepwise Refinement)
- Examples: Replicating Code

# Functions



- What is a function?
  - Block of code to perform action/subroutine
- When have we seen functions already?
  - Predefined
- What is the purpose?
  - Reduce
  - Reuse
  - Readability

# Predefined Functions

- `sqrt()`
- `pow()`
- `abs()`
- `rand()`
- `srand()`
- What is the difference b/w `srand()` and others?

# Procedural Decomposition

- Functions
  - int **main**() { }
  - User defined
    - void draw\_box() { }
- Function Call
  - draw\_box();

# Procedural Decomposition



```
#include <iostream>
using std::cout;
int main() {
    cout << "+-----+\n";
    cout << "|       |\n";
    cout << "+-----+\n";
    cout << "+-----+\n";
    cout << "|       |\n";
    cout << "+-----+\n";
    return 0;
}
```

```
#include <iostream>
using std::cout;
void draw_box(); //Declare function
int main() {
    draw_box(); //Use function
    draw_box();
    return 0;
}
void draw_box() { //Define function
    cout << "+-----+\n";
    cout << "|       |\n";
    cout << "+-----+\n";
}
```

# Functions Calling Other Functions



Oregon State University  
College of Engineering

```
#include <iostream>
void draw_box();
void draw_top_bottom();
void draw_sides();
int main() {
    draw_box();
    return 0;
}
void draw_box() {
    draw_top_bottom();
    draw_sides();
    draw_top_bottom();
}
void draw_top_bottom() {
    std::cout << "+-----+\n";
}
void draw_sides() {
    std::cout << "|           |\n";
}
```

# Generalization



- Does a function make a task more specific or more general?
  - Justification
  - Examples



# void Functions

- Doesn't return a value
- Still has arguments/parameters



# Programming Demo



Oregon State University  
College of Engineering