

Administrivia

- ❖ Exercise 8 released today, due Wednesday
 - First C++ exercise!
 - Be sure to use the new linter: cpplint.py
 - Parallels to ex0 – compare user input checking between C/C++
- ❖ Homework 2 due next Thursday (2/6)
 - File system crawler, indexer, and search engine
 - Note: Make sure a copy of libhw1.a is in the hw1/ directory
 - Either your own (run make) or ours (copy from hw1/solution_binaries)
 - Demo: use Ctrl-D to exit searchshell gracefully, test on directory of small self-made files

3

3

Hello World in C++

helloworld.cc

```
#include <iostream> // for cout, endl
#include <cstdlib> // for EXIT_SUCCESS

int main(int argc, char** argv) {
    std::cout << "Hello, World!" << std::endl;
    return EXIT_SUCCESS;
}
```

- ❖ “<<” is an **operator** defined by the C++ language
 - Defined in C as well: usually it bit-shifts integers (in C/C++)
 - C++ allows classes and functions to overload operators!
 - Here, the `ostream` class overloads “<<”
 - i.e. it defines different **member functions** (methods) that are invoked when an `ostream` is the left-hand side of the << operator (LHS)

12

12



Let's Refine It a Bit

helloworld2.cc

```
#include <iostream> // for cout, endl
#include <cstdlib> // for EXIT_SUCCESS
#include <string> // for string

using namespace std;

int main(int argc, char** argv) {
    string hello("Hello, World!");
    cout << hello << endl;
    return EXIT_SUCCESS;
}
```

- ❖ The `using` keyword introduces a namespace (or part of) into the current region
 - `using namespace std;` imports all names from `std::` *(linter will complain, but we will ignore)*
 - `using std::cout;` imports *only* `std::cout` (used as `cout`)

18

18

Poll Everywhere

pollev.com/cse33320su

- ❖ How many *different* versions of << are called?
 - For now, ignore manipulator functions
 - Bonus: what is printed?

A. 1

B. 2

C. 3

D. 4

E. We're lost...

msg.cc

```
#include <iostream>
#include <cstdlib>
#include <string>
#include <iomanip>

using namespace std;

int main(int argc, char** argv) {
    int n = 172;
    string str("m");
    str += "y";
    cout << str << hex << setw(2) // Don't count these
        << 15U << n << "e!" << endl;
    return EXIT_SUCCESS;
}
```

30

30