

Name: _____

Today's Date: _____

Today's Goals

- Explain why we need *iterators* for our data structures.
- List the functions that need to be implemented to have a functioning iterator.
- Practice writing an iterator.

Today's Question(s)

Why should a class's *interface* not leak information about the class's *encoding*?

Lingering Questions

Cow.hpp and Barn.hpp

Last time, we had a (relatively) simple Barn class:

```
class Barn {  
  
public:  
    Barn();  
    Barn(const Barn& otherBarn);  
    ~Barn();  
  
    void visit();  
    void addCow(const string& cowName);  
    bool hasCow(const string& cowName);  
};
```

along with the definition of a Cow class:

```
class Cow {  
public:  
  
    Cow() = default;  
    ~Cow() = default;  
    Cow(const std::string& cowName);  
  
private:  
    std::string name_;  
};
```

Looping Through a Barn

How can we print all of the Cows in the Barn?

```
int main()
{
    Barn b;
    b.addCow("bessie");
    b.addCow("mabel");

    for (
        cout <<          << endl;
    ) {

        return 0;
    }
```

Interface, Encoding, and Implementation

Iterators

- ▶ An Iterator is....
- ▶ We need to change the Barn class by...

We also need to be sure to...

Class Exercise: Implementation