Review

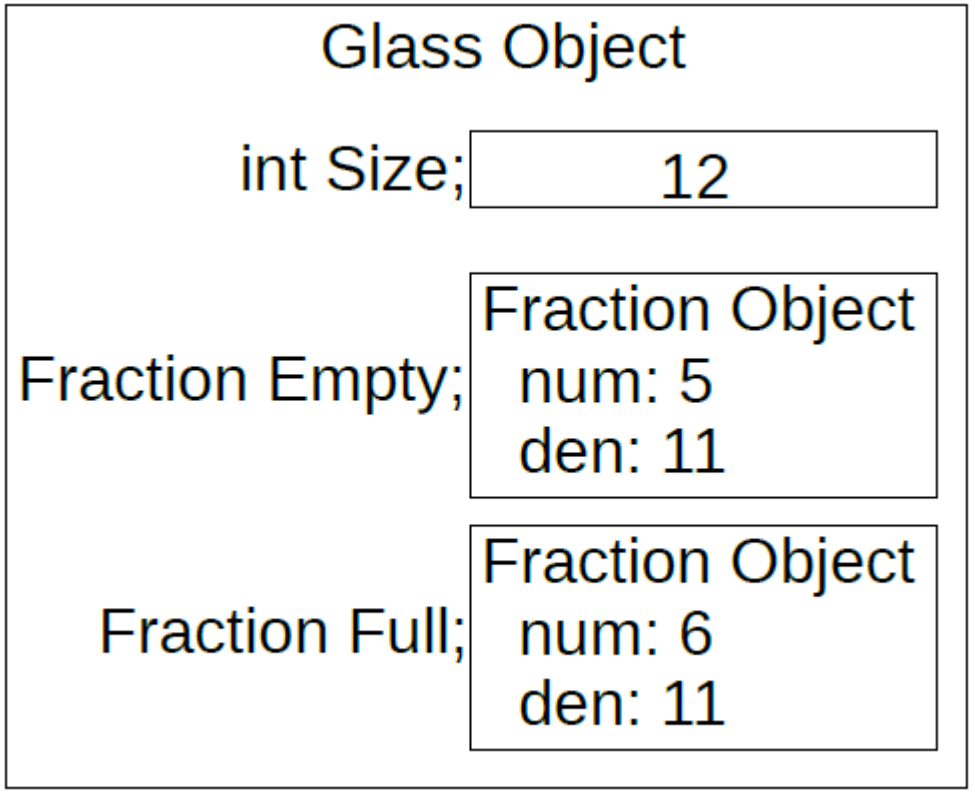
- What is operator overloading
- How to overload operator + in your class?
- Name three limitations of operator overloading
- List two potential methods to overload the + operator

Composition

Object as member data

- Objects are a combination of member data, member functions and an interface.
- Objects can also be member data (objects within objects). For example

```
Class Glass {
  int size;
  Fraction Empty;
  Fraction Full;
}
```



Composition

- The relationship of “an object within an object” is called composition.
  - Can be implemented by declaring an object or an object pointer/reference within the member data of a class.
  - Often described as the “has-a” relationship
    - Glass has-a Fraction
    - Car “has a” Engine (object Engine is member data within Car class)
    - Deck object has 52 Card objects
- Composition allows code to be more modularized
  - We can create smaller classes and combine them to realize larger functionality.
  - See PokerHand Example

Member Data Object Constructor

- When an object is created, its constructor runs, it must also invoke the constructor for any embedded objects

```
Class small_class {
  public:
     small_class(int);
  private:
     int data;
  }
 small_class::small_class(int d) { data = d;}

Class large_class {
  public:
    large_class();
  private:
   small_class sc; /* cannot call constructor here */
}
```

- If nothing else is done, the default construct for the member function will be called.
  - Which constructor is called earlier? See ph2.cpp
  - What if we want to use a (non-default) constructor for the member data?

Member Data Object Constructor

- How the object within an object is initialized?

```
Class small_class {
  public:
     small_class(int);
  private:
     int data;
  }
 small_class::small_class(int d) { data = d;}

Class large_class {
  public:
    large_class();
  private:
   small_class sc; /* cannot call constructor here */
}
```

- What if we want to use a (non-default) constructor for the member data?
- Use initialization list: large_class::large_class(): small_class(1000) { }, see ph3.cpp, ph4.cpp
  - This has limitations. May need to just call the constructor inside the constructor for the large class.

Extending the dot operator

- If and object that is member data of another object has public members (data or functions), we can access it using the dot operator.
- See sample1.cpp