Review

- What is composition?

- When the member object initialized?

- Which constructor is used to initialize the member object if we don't do anything special?

- How to initialize the member object with a particular constructor.

Arrays of objects

Arrays review

- An array is an indexed collection of data elements of the same type.

  int aaa[10];

  - ❖ 10 elements of the int type. Each array position is a single element
  - ❖ What is the range of the index?
  - ❖ What are the 10 elements?

Declaring arrays of objects

❑ Declaring arrays of objects is similar to declaring arrays of built-in types

   Fraction rationals[20]; // array of 20 Fraction objects

   Complex nums[50]; // an array of 50 Complex objects

   Hydrant fireplugs[10]; // an array of 10 fireplugs

❑ Each array position is a single object

   • 'Fraction rationals[20];' declares 20 Fraction objects, rationals[0], rationals[1], …, rationals[19].

Initializing the array of objects

- Similar to a number array declaration.

    - Do nothing to use the default constructor

        int x;

        Fraction num;

        Fraction num[4];

    - To initialize in a particular way, call an explicit constructor

        Int x(10);

        Fraction num(10, 20);

    - How to do array of objects? Need a way to specify different constructors to different elements.

Initializing the array of objects

- To initialize in a particular way, call an explicit constructor

    Int x(10);

    Fraction num(10, 20);

- How to do array of objects? Need a way to specify different constructors to different elements.

    – Use an initializer set to give a constructor to each element

    Fraction numlist[3] = {Fraction(2, 4), Fraction(5), Fraction()};

    – numlist[0] is initialized with constructor Fraction(2,4);
    – numlist[1] is initialized with constructor Fraction(5);
    – numlist[2] is initialized with constructor Fraction();

Using the array of objects

- Indexing works the same as with regular arrarys

  – Each object in the array is in the form of ***arrayName[index];***

- The dor-operator works the same as with single names.

  objectName.memberName

- The objectName is in the from of an array item:

  – ***arrayName[index].memberName***

- Example

  ```
  Fraction rationals[20];
  …
  rationals[2].show();
  rationals[6].Input();
  for (i=0; i<10; i++) rationals[i].setval(20);
  for(i=0; i<20; i++) rationals[i].pubval = 50;
  ```