7.4 Arrays of Objects

When working with arrays of objects, you will need to use new twice. Once to create the actual array, and then for each object you want to store in the array.

Fraction fractions = new Fraction [1000];



fractions[0].num = 3; X Error, no fractions[0] object created yet fractions[0] = new Fraction(1,2); fractions[0].num = 3; fractions[1] = fractions[0].times(fractions[0]; Fraction f = new Fraction(3,5); fractions[2] = f;

Example: Suppose we need to store the names, studentNumbers and averages of students.

There are two approaches. (1) If not using Object Oriented Programming (OOP) you would use something called parallel arrays.

```
String[] names = new String[30];
long[] stuNum = new long[30];
float[] ave = new float[30];
```

index 0 in each of the arrays is referring to the same student. index 1 in each array is referring to the second student and so on.

To print info about the first student we could do something like this:

```
println(names[0] + ": Student # " + stuNum[0] + " Average = " + ave[0]);
```

(2) With objects:

```
class Student
{
   String name;
   long stuNum;
   float ave;
```

IIOUC UV

}

Student [] students = new Student[30];

To print info about the first student:

println(students[0].name + ": Student # " + students[0].stuNum + " Average = "
+ students[0].ave);

With objects you store everything in one array. It is more clear that you are talking about 1 entity. With parallel arrays you have multiple related arrays but it is not as clear when you are talking about 1 entity.

Read section 7.4 in the textbook. Do exercise 7.4 #1-3 (I'll post solutions on Friday). Remember the assignment is due Monday. Send me an email when you decide who you want to work with for the Project and what program you would like to.