Lecture 4
Overload the = operator to Array copy.
1. Why pass by value can be inefficient
2. Pass by const reference
3. Why returning value can be inadequate and inefficient
4. Returning a reference

Size and capacity (memory management), static const data member

Copy semantics (shallow, deep)
Memory leak, memory fault (dangling reference)

Default constructor, copy constructor, destructor
Why C++-given constructors and destructor are dangerous

Exercise: change getInput into several versions: (member function overloading)
(overloading: one name, many meanings), default argument

Void getInput(int),
void getInput(int, RandomNumberGenerator * rng=0)
{
    if (rng)
        for (int I=0; I<n; ++I)
            a[I] = rng->nextInt();
}

value, pointer, reference

RandomNumber, static data member, factory method (GoF), singleton (GoF)

Read [Lippman, sec 4.15, sec. 5.11]