```
// -------------- C++ simulate.h

/**
   * Simulation controls a simulation of fish as
   * represented in an Environment.
   *
   * One step of a simulation can be performed via Step(..),
   * an arbitrary number of steps via Run(..).
   *
   */

class  Environment;

class  Simulation
{
   public:
      Simulation();
        // postcondition: simulation is ready to run

      void Step(Environment & env);
        // postcondition: one step of simulation in env has been made

      void Run(Environment & env, int steps);
        // postcondition: simulation on env run for # steps passed as steps

};


// -------------- C++ simulate.cpp

#include  "simulate.h"
#include  "apvector.h"
#include  "environ.h"

Simulation::Simulation()
// postcondition: simulation is ready to run
{
}

void Simulation::Step(Environment & env)
// postcondition: one step of simulation in env has been made
{
    apvector<Fish> fishList;
    int k;

    fishList = env.AllFish();
    for (k = 0; k < fishList.length(); k++)
    {
       fishList[k].Move(env);
    }
}

void Simulation::Run(Environment & env, int steps)
// postcondition: simulation on env run for # steps passed as steps
{
    int k;

    for (k = 0; k < steps; k++)
    {
       Step(env);
    }
}
```