Robert Puentes

Matthew Jaramillo  
Estevan Devargas

Documentation for Project 3

We were asked in Project 3 to design a model and solve a real-world issue called the Acequia Manager Simulation. It is a design to manage water flow through New Mexico, through its irrigation system in all the canals. The simulations run for a randomly generated number of hours and each region has a water level, a minimum required amount of water, and a flooded/drought status.

The goal is to balance the water system by designing the canals in a way that no region floods or is in a drought, that every region meets its water need, and is achieved within the allowed simulation time. It gives you a score based on how many regions are balanced and penalties for floods/droughts.   
  
We were given 3 .cpp files:  
- setup the water simulation

- create random region data

- manage simulation time and region state

- score the results

What we had to do was create the “solveProblems” function in the StudentSolution.cpp file which is where we had to implement the managing logic using the canals, region, water source.  
  
We wrote a loop that runs until a simulation ends, meaning until it gets solved or when time runs out.

Our process:

* Accessed the list of canals and regions
* For each hour:
  + Looped each canal
  + If the destination region needed water and the water source has water, open the canal
  + If the source region had more water than it needed, we allowed it to share water to other regions
  + Set the flow rate to 1
* Called the “manager.nexthour()” to move the forward and update the water levels

After running the simulation

* The initial state is saved to a file “randomvalues.dat”
* Code solves the problem using the logic
* Once finished, it evaluates the solution and prints:
  + If the region were solved in time
  + The number of penalties
  + A score