

The Battle of the Kingdoms

New Mexico
Supercomputing Challenge
Final Report
April 1, 2006

Team 12
Artesia High School

Jeffrey Heidenreich
Craig Mayberry
Ashley Durham
Jessica MacKinnon
Kathryn Moore

Mr. Randall Gaylor
Mr. Jim Orth

Dr. Tom Brown

Table of Contents:

Acknowledgments.....	3
Executive Summary.....	4
The Code.....	5
Putting It Simply.....	9
Advanced Observer.....	10
Advanced Turtle.....	11
Data and Its Application.....	12
Results.....	13
Conclusion.....	14
A Note From Team	15
Works Cited.....	16

Acknowledgments:

Mr. Gaylor:

We would like to thank Mr. Gaylor for always being there to help us when we were in need. He was always trying to keep us on task, even though none of us really wanted to. He would pester us until we actually did get something done. We would also like to acknowledge him for convincing the school administration to allow us to join into this program and to provide us with the funding necessary for the things we required. To put it simply, without Mr. Gaylor, nobody would be reading this report.

David Kratzer:

We would like to thank Mr. Kratzer for his aid in the last few days of the challenge. Without his help, we wouldn't have been able to complete our project. We would like to say thank you for allowing us the use of your computer and for spending the time running our program for us.

Tom Brown:

We would like to thank Mr. Brown for offering to be our mentor. If he hadn't have offered, we wouldn't have a mentor. Because of him, we also had some security knowing that if we needed help, we would have somebody just a phone call or e-mail away.

We would also like to thank everyone else who helped us during our project: Mr. Orth, all those at the Kickoff Conference, the school administration, the judges at ENMU-Portales, the Supercomputing Challenge Board of Directors, Mr. Stall, and last, but not least, those who will be reading and judging this report.

If there is anyone that we failed to mention, we apologize and say 'Thank You!'

Executive Summary:

Many people often argue and disagree on the common problem of deforestation. It has been a hot topic of many naturalists around the world, saying how the human race is going to die out because we are going to kill off all of our oxygen producing plant life. We, the Artesia High School Supercomputing Challenge Team, wondered if this argument was really founded in fact, or if it was just an over exaggeration founded by naturalists who just wanted to save the trees.

How much of the forests are being destroyed? Are the consequences really as bad as people make them out to be? How long is it going to take for all humans to die off, if that really is what's going to happen? What are the consequences on things other than humans? These are some high priority questions that our program is designed to answer.

In thinking about the potential ways to solve this problem, our team thought that maybe the way things kill and die is like a predator-prey model. That thought led us to StarLogo, a problem often used to model predator-prey. The humans were to be the predators, and the trees the prey.

After going through a few simple models and constantly enhancing them, adding more variables, more actions, and more values, we came to the program that we now use.

Our program is a model that shows trees, people, and animals. They go about life doing the things they do in real life. The actual modeling of the running doesn't really show too well. The actual information comes from the graph running on the side. This graph shows the number of humans, animals, and trees living in our program. It shows as things are put in and/or taken out.

Due to the fact that every time the program is run, things are set in random motion, results are never exactly the same, but due to the program's basic functions, they end up very close.

Our program models and logs the effects of deforestation on the animal and human populations.

The Code:

Observer:

```
to startup
  ca
  intro-trees
  intro-dioxy
  intro-animals
  intro-destruction
  intro-people
end

to intro-trees
  crt 101
  ask-turtles
  [seth random 360
  jump random 72
  if count-turtles-here > 1
  [seth random 360
  fd 1]
  stamp green]
  ask-patches
  [if color = black
  [sprout
  [setc green setshape tree3
  setbreed tree
  setage random 157
  setlifeforce (age / 2)
  setrepr random 1000
  setcarbc random 34]]]
  ask-turtles
  [if who < 101
  [die]]
  cp
end

to intro-animals
  create-anim 3000
  ask-anim
  [setc purple
  setxy (random screen-height) (random screen-width)
  set repr random 20
  if count-tree-here = 0
  [die]]
end
```

```
to intro-destruction
  create-dest 24
  ask-dest
  [if count-tree-here < 5 [ht]]
end
```

```
to intro-people
  create-ppl pplnum
  ask-ppl
  [setxy (random screen-height) (random screen-width)
  setc 14
  setlifeforce random 4
  setage random 78
  set repr random 25]
end
```

```
to intro-dioxy
  create-oxy 2500
  ask-oxy
  [setxy (random screen-height) (random screen-width)
  ifelse count-tree-here > 0 [ht]
  [setbreed co2 ht]]
end
```

Turtle:

```
breeds [ppl CO2 oxy tree anim dest]
turtles-own [lifeforce age repr carbc]
```

```
to go
  if breed = dest
    [destroy]
  if breed = co2
    [move
    convert]
  if breed = ppl
    [move
    breathe
    old
    reproduce]
  if breed = oxy
    [move
    convert]
  if breed = tree
    [life]
  if breed = anim
```

```

[scamper]
end

to breathe
ifelse count-oxy-here > 0
[set lifeforce 4]
[set lifeforce (lifeforce - 1)]
end

to old
if lifeforce < 0 [die]
setage age + 1
if age > 78 [die]
end

to reproduce
setrepr (repr + 1)
if age > 16 and repr > 100
[if count-ppl > 1 and count-ppl < 1000
[hatch
[setage 0
setlifeforce 4
setxy (random screen-height) (random screen-width)]
setrepr 0]]
end

to move
fd 1
rt random 360
lt random 360
end

to convert
if count-ppl-here > 0
[setbreed co2 setc white]
if count-tree-here > 0
[setbreed oxy setc blue]
end

to life
tree-die
grow
end

to tree-die

```

```

    if count-dest-here > 0
      [die]
    end

    to grow
      setage (age + 1)
      setlifeforce (lifeforce + .5)
      if age > 75
        [seed]
      end

      to seed
        setrepr (repr + 1)
        if repr > 1000
          and count-tree < 2500
            [hatch
              [setage .1
                setlifeforce .05
                setxy (random screen-height) (random screen-width)]
              setrepr 0]
          end
        end

        to destroy
          setxy (random screen-height) (random screen-width)
          if count-ppl < 1 [die]
          end

        to scamper
          ifelse count-tree-here < 1
            [seth random 360
              fd 1
              if count-tree-here < 1
                [die]]
            [if repr > 75 and count-anim < 3000
              [hatch [setxy (random screen-height) (random screen-width)]
                set repr 1]]
          end
        end
      end
    end
  
```


Putting It Simply:

This section explains the program simply and makes the instructions flow together.

The observer commands simply introduce everything into the world. The correct numbers of oxygen, trees, carbon dioxide, people, animals, and destruction turtles are put in the world, and their variables are randomly set.

The actual running of the program is a little more difficult. These are the turtle commands. The simple running of the program is as follows:

1. Destruction turtles move to a random space. If there is a tree, the tree dies.
2. Oxygen and carbon dioxide molecules move one space. If there is a person, it becomes carbon dioxide. If there is a tree, it becomes oxygen.
3. People wander around aimlessly, breathing air, reproducing, and growing old.
4. Trees sit around doing nothing except growing old and occasionally reproducing.
5. Animals live happily in their tree homes until the tree they are in dies. They then get one chance to move to a new tree. If they don't find a tree in one turn, they die.

Advanced Observer:

StarLogo commands are actually quite simple. A programmer only needs to set up their own series of functions and put them together into one large program.

Our observer code introduces everything into the environment.

It begins by introducing the trees: approximately 2500, possibly a few extra or under. 101 turtles are created and they randomly move to a place on the field and stamp the patch green. All non-green patches then sprout a tree. These trees then set their variables to random values.

Next, the oxygen and carbon dioxide molecules are introduced. For each tree, there is one molecule, so 2500. They are all introduced as oxygen, and in fact, most of them stay that way. All of them move to a random place on the model and settle down. If it is on a tree, it remains oxygen. If it is on a blank space, it becomes carbon dioxide. All these turtles then become invisible.

Now the animals enter the world. 3000 of them are created and randomly spread across the map. If they are on a tree, they stay. If they are on one of the spaces without a tree, they die. Their variables are then randomly set.

It is now time for the destruction turtles to join the game. 24 of these turtles are introduced onto the field. They find a random place to settle, then disappear.

Finally, people enter the model. A slider allows the user to choose how many humans they would like to be in the forest at one time. That many humans are created and randomly placed about the map. Their variables are randomly set, and they're ready to begin their journey through life.

Advanced Turtle:

The turtle commands actually run the program.

The first little bit simply shows what the things we added ourselves are. There are our breeds, which we have already introduced, and there are our variables, such as age, life force, and repr (reproduction).

First, the destruction turtles move to a random spot on the field. If there is less than one person on the board, all the destruction turtles die.

The carbon dioxide molecules move and convert. To move, they simply move forward one space, then rotate to a new direction. If they land on a tree, they become an oxygen atom.

People move, breathe, get old, and reproduce. Moving consists of going forward one space, then rotating randomly right or left. They breathe every time they land on an oxygen atom. If they take five turns without breathing, they die. Every time they land on an oxygen atom, their life force is replenished back to 5. Every movement, they age one year, and at 78 years old, they die. To reproduce, they wander around. Every turn, they gain one repr. When they reach the top number, they hatch a clone, which then sets its age and repr to the correct level.

Oxygen also moves and converts... It does the same things as carbon dioxide except it converts to carbon dioxide when they land on a spot with a human.

Trees live a normal tree's life. They sit and do nothing. However, if a destruction turtle lands on its square, the tree dies. They also grow... Every turn, they age one year and gain .5 life force. If they are older than 75 years, they seed. This means that they begin to gain repr points. When they reach their level, they hatch a clone. These clones set their age and repr correctly and move to a new square.

Finally we come to the animals. Animals stay in their little home of trees. If a tree gets chopped down, they turn a random direction and move one. If there's another tree there, they settle down. If there isn't, they die. They also reproduce, but at a slightly faster rate than others.

Data and Its Application:

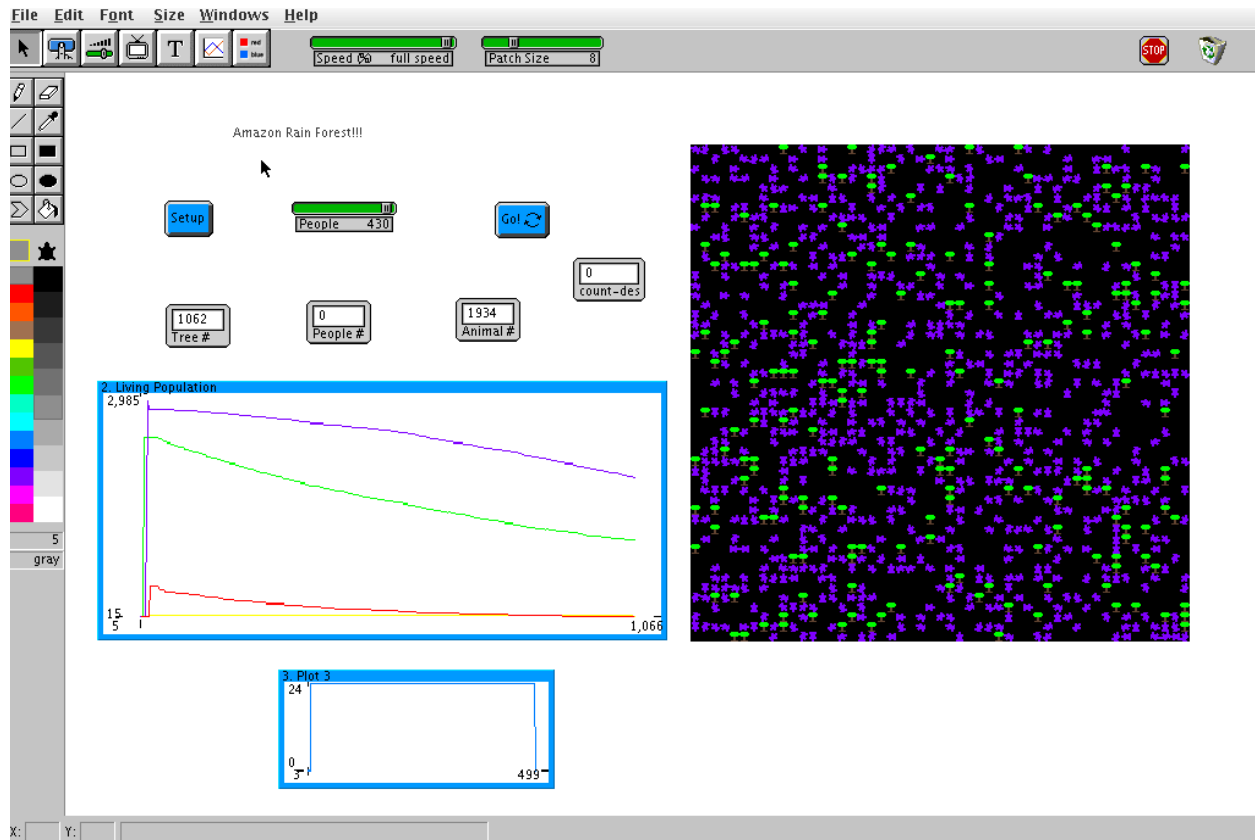
The data that we found is based off of the Amazon Rainforest.

The Amazon Rainforest is 7 million square kilometers large (“Amazon Rainforest”). We found that approximately 26129 square kilometers are being destroyed every year. Only 10 per cent (2610 square kilometers) are being replenished every year. Therefore, we are losing about 24000 square kilometers every year.

In our program, there are 2500 originating tree turtles. These each represent 1000 square kilometers... Therefore, 2,500,000 square acres of forest are being represented in our model. The 24 destruction turtles represent the approximate 24000 square kilometers destroyed.

We realize that the Amazon Rainforest is only a small fraction of the world’s oxygen producing plants, but if all variables are increased, the results should remain about the same.

Results:



(Thanks to Mr. David Kratzer for this Image)

These are the results of our program. On the large line graph, the purple line represents animals, the green line represents trees, and the red line represents people.

As you can see, the people meet their doom in just over 1060 years. The trees and animals dropped significantly from their beginning, but never bottomed out. With the way this is programmed, both will gradually and slowly grow to their maximum level and remain co-existing peacefully for the rest of eternity.

Conclusion:

Our model showed a very significant effect on humankind. We all died. This is a very large problem, indeed. What can be done to fix this? We could always stop or slow the cutting down of trees. We could also plant more trees.

Of course, there is a potential flaw in our results. With our experience of programming, we found StarLogo incapable of lessening the number of destruction turtles as the number of people decreased. As the human population sinks, so will the need for supplies from wood. Therefore, fewer trees would be chopped down.

The reason that is only a *potential* flaw is because another StarLogo incapability could have compensated for it. We never discovered a way to make the destruction turtles appear only on trees. So, as the number of trees decreases, the destruction turtles become less likely to land on the trees.

In all conclusion, we discovered that a project this broad is very difficult to do with a program such as StarLogo. Results are never quite exact. But with what results we did end with, we still note that humankind is in potential danger. We would like to advise people to at least watch as they chop down trees, and if too many become chopped down, to slow production and replenish our air-givers.

A Note From Team 12:

We would like to add something that isn't really about our program. We would like to say how much we learned from this experience. We learned some advanced programming skills, which was nice. We also learned some valuable lessons.

We learned how valuable teamwork was. We found out how if a team doesn't participate as a team, there can be some consequences. We faced that dilemma and soon solved it. We saw how working as a team can accomplish more than working individually.

We would also like to note that we enjoyed this year's competition very much and most of us look forward to joining it again next year.

Works Cited:

- “Amazon Rainforest.” Wikipedia. Microsoft Internet Explorer. 1 Mar. 2006. 3 Mar. 2006
<http://en.wikipedia.org/wiki/Amazon_Rainforest>.
- “Deforestation in the Amazon 2004.” Mongabay.com. Microsoft Internet Explorer. 2005.
7 Mar. 2005 <<http://www.mongabay.com/brazil.html>>.
- “Facts and Figures.” Rainforest Live. Microsoft Internet Explorer. 2005. 10 Mar. 2006
<<http://www.rainforestlive.org.uk/index.cfm?articleid=214>>.