Predicting Wild Fires

New Mexico
Supercomputing Challenge
Final Report
April 1st 2007

Team 1
The Academy for Technology and the Classics

Team Members
Chris Hughes
Ruben Hamming-Green
Teachers
Terrill Nickerson
Patricia Joseph

Table of Contents

Summary

People pay a lot of attention to hurricanes and tornadoes as the most devastating natural disasters that Americans face. Forest fires are not usually as devastating as hurricanes or tornadoes, but they can do a lot of damage. Predicting and modeling them is very important in helping reduce the damage they do in New Mexico and other dry parts of the country. Not only will better forest fire modeling programs help to reduce the damage and spread of forest fires, but it can help authorities decide where they should do controlled burns and forest clearing.

The main goal of our project was to create a graphical computer program that will be simple to use and can be applied to many different forest fire scenarios. We wanted the program to be as accurate as possible to the actual behavior of forest fires. Our program is easy to use and also predicts the spread of forest fires as accurately as possible taking into consideration many different factors like wind, elevation, distance between trees, etc.. Our program allows anyone using the program to easily change variables such as wind speed, wind direction, and topography. The user can also clear certain areas of trees.

Our program could potentially be useful to fire fighters fighting a fire who need to know where the fire is going to go and where they should clear. Also, it would be useful to policy makers who needed to know how controlled burns and forest clearing would affect future forest fires. We wanted to be sure that our program would not just be useful in stopping a fire, but also in preventing a fire. With improvements like more research and some more tweaking to make it more accurate, we believe our program could be used by policy makers as well as fire fighters in real life.

The Problem

Forrest fires are a growing concern in Santa Fe county and in many other areas of the world where the weather is hot and dry. They take a toll on natural resources and cause damage to lots of valuable property. Managing forest fires and keeping them under control is important to the safety of many American's. Since forest fires do occur naturally and are important to the ecosystems of the nation's forests we must find a way to make sure they do happen, but are kept under control. Government officials have to make decisions about where to do thinning projects or controlled burns. Government officials would be able to make more informed decisions if they had a simple way to analyze the results of their actions.

Forest fires have caused a lot of damage, especially in New Mexico. Reducing the amount of damage they do is an important goal for New Mexico government officials. They need an accurate, easily understandable way to see what effects their decisions will have. A forest fire modeling program could be used in a decision making environment to show the consequences of any decisions related to forest fires.

Fire fighters work very hard to keep blazing forest fires under control, but they are often under prepared for what the forest fire does. They need an accurate and fast way to predict what the forest fire will do and prepare accordingly. A forest fire modeling program could give fire fighters an easy way to get an accurate prediction of the path the fire might take.

The Solution

We have created a program that will model the behavior of a forest fire in response to weather conditions, topography, the density of the trees, and other factors. It shows a visual of the progression of the fire and other helpful data. The program would be helpful to anyone who needed to know what a forest fire would do in various conditions. Government officials, fire fighters and other authorities could be able to use the program to predict the behavior of a forest fire to help them make decisions.

The program shows a simple but clear image of the spread of a forest fire and the conditions can be altered to see how the fire reacts. Decision makers and fire fighters alike could use our program or a similar forest fire mapping program to see how their actions as well as the weather and other factors will effect an actively burning or hypothetical fire.

Results and Observations

Our program showed us a lot of useful things about forest fires. We noticed a lot of trends and behaviors of forest fires we might not otherwise have known about. Our program, although it doesn't show exactly what would happen in a certain situation, gives a very good idea of the behavior of forest fires. Our program shows the effects of different wind speeds and directions on a forest fire as well as what effects thinning and clearing would have on a forest fire in a certain area.

One thing we noticed that we were a little surprised about is that wind has a huge effect on the behavior of forest fires. When the wind is blowing in a certain direction, even lightly, the forest fire will spread much more rapidly with the wind than against the wind. We knew that wind had a large effect on forest fires, but not as large as our program shows. With a strong wind the forest fire will not spread at all against the wind.

Another thing we noticed in our program is that forest fire has quite a bit of trouble spreading over long distances between trees. In real life situations where there is a lot of brush in between trees fire probably has a much easier time spreading between trees that are far away. Since our model doesn't incorporate undergrowth we see that fire doesn't spread very easily between trees that are distant from each other without the help of undergrowth.

Our program helped us understand forest fires a lot better, so I'm sure it could be very helpful to fire fighters and policy makers that needed to know more about forest fires.

References

1. 'Wildfire' – wikipedia.org

2. 'New Mexico Fire Information' – nmfireinfo.com

3. 'National Interagency Fire Center' – nifc.gov

4. 'New Mexico Incident Management Team' – nmimt.com

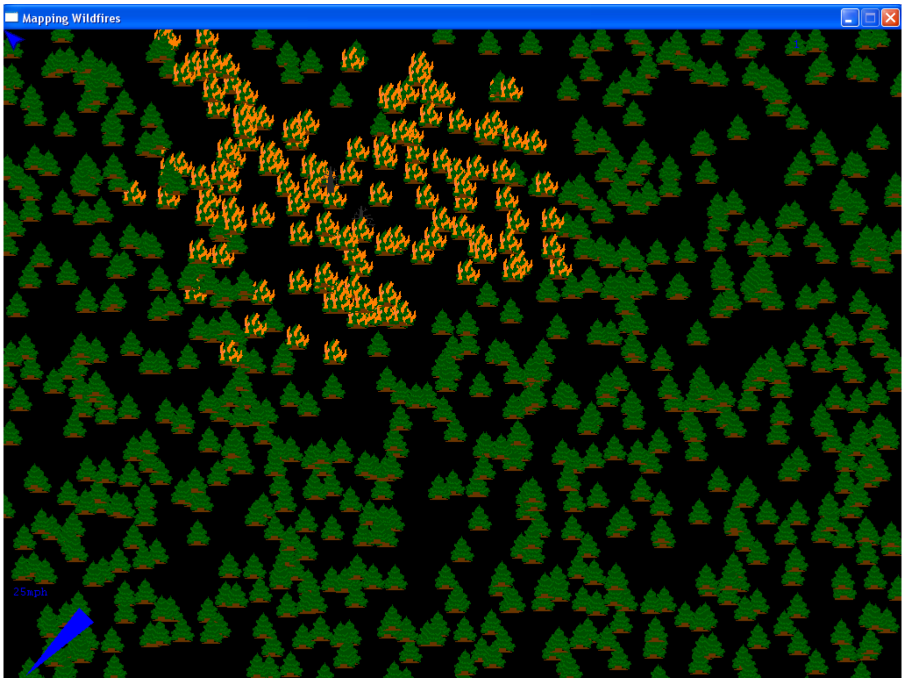
5. 'Bureau of Land Management' – blm.gov

6. 'FEMA' - fema.gov

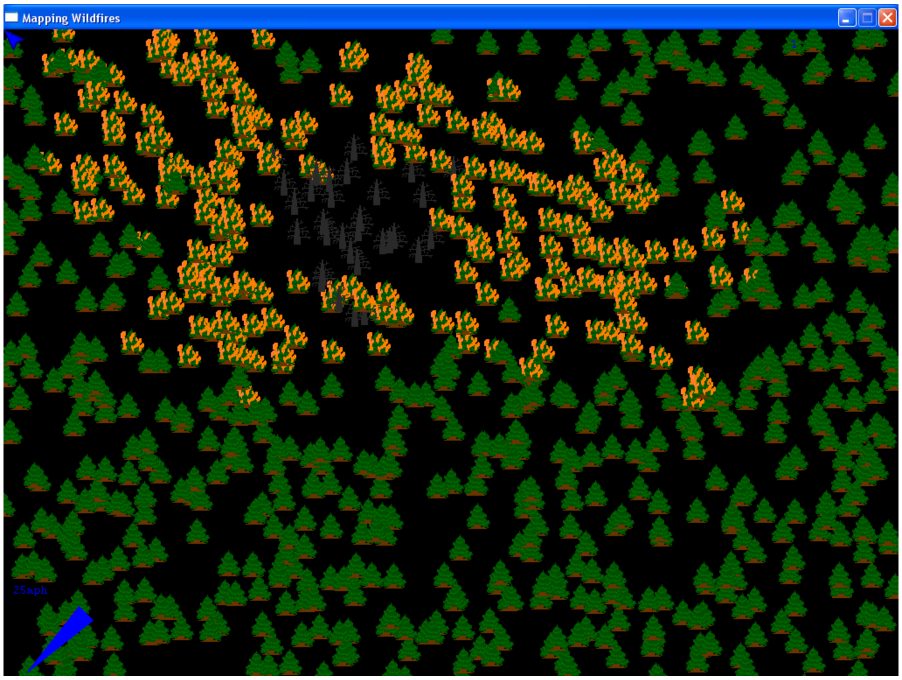
Apendix A: Screenshots

These Screenshots show the progression of a forest fire in our program without wind or topography.

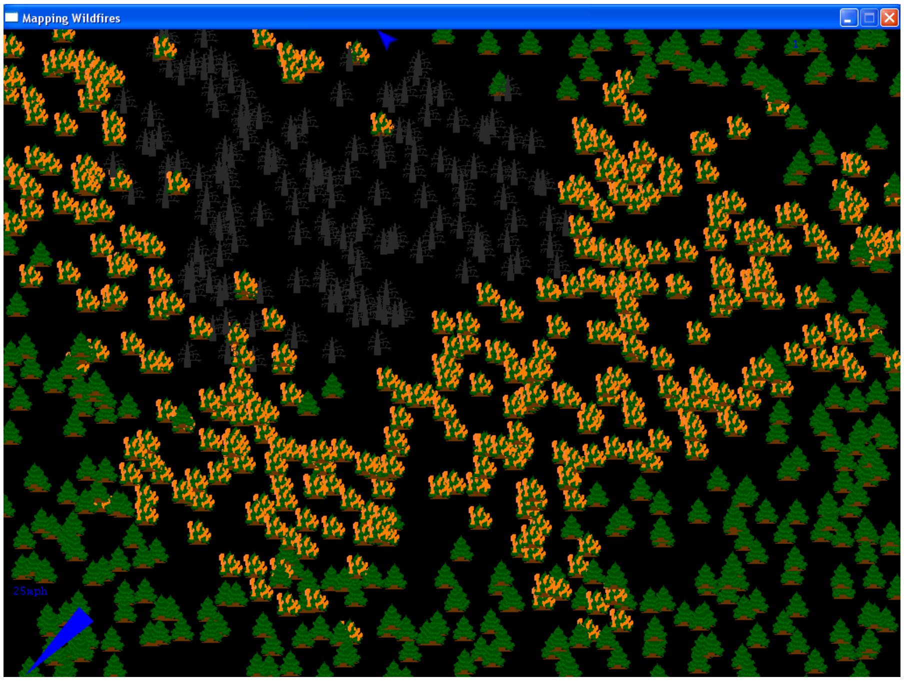
1:



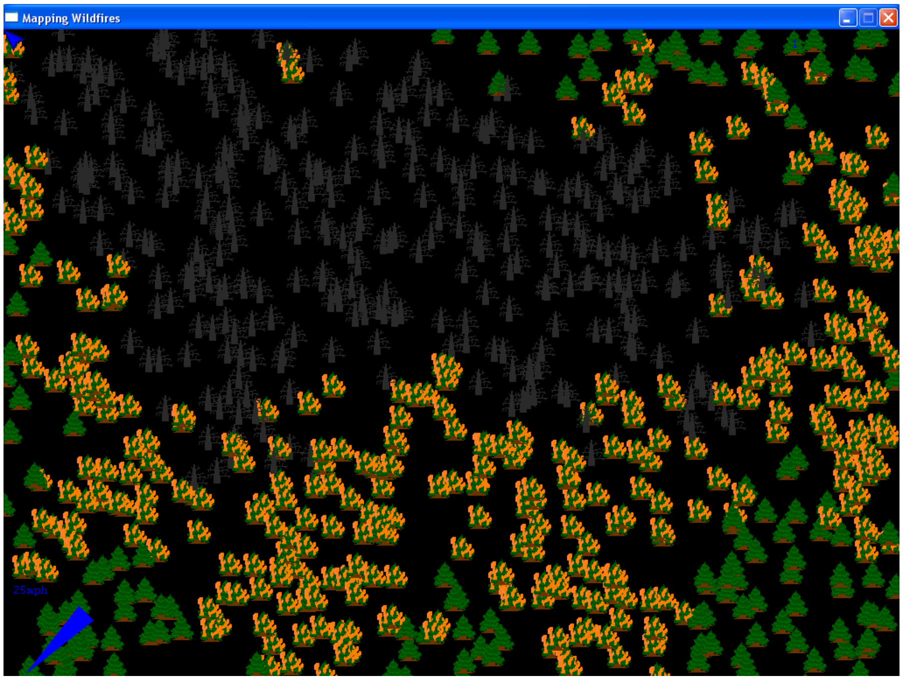
2:



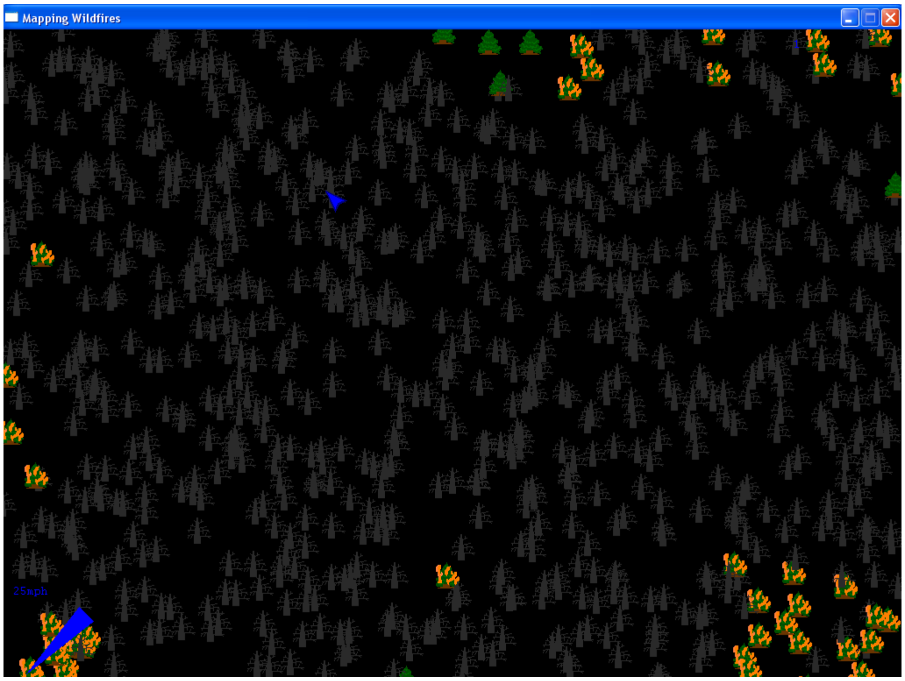
3:



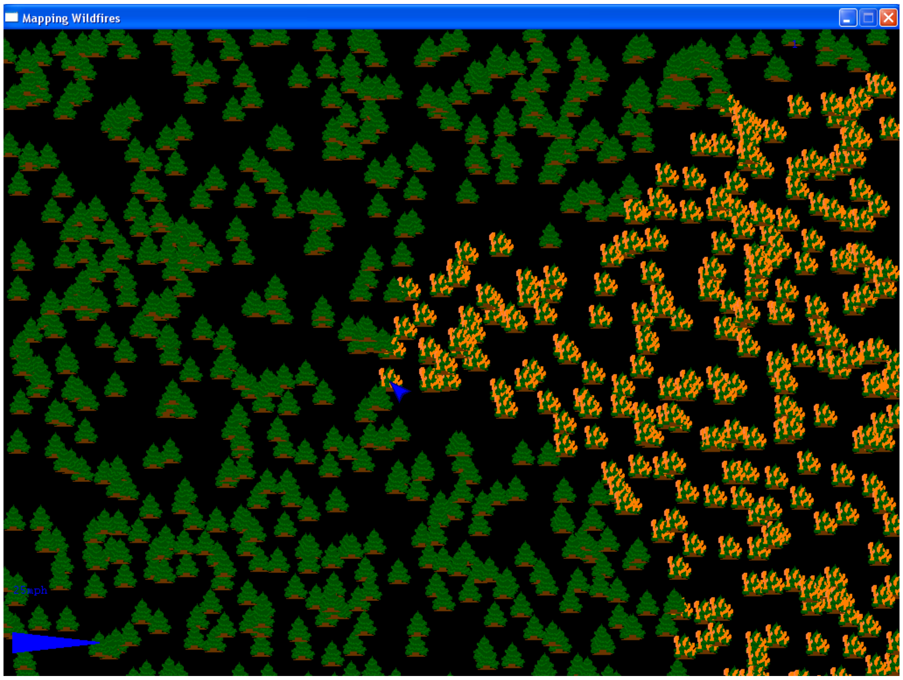
4:



5:



This shows the effect of wind on a fire:



Apendix B: Source Code

This is our source code, written in Blitz Basic (www.blitzbasic.com):

```
;;;;;;;;;;;;;;;;;
;Mapping Wildfires by Ruben Hamming-Green and Chris Hughes
;;;;;;;;;;;;;;;;;;

AppTitle "Mapping Wildfires"
SetBuffer BackBuffer()
Graphics 1024, 768, 0, 2

AutoMidHandle True

SeedRnd MilliSecs()

;Loading all the images
Treei = LoadAnimImage(CurrentDir() + "Images/Tree.bmp", 32, 32, 0, 1);healthy
BTreei = LoadAnimImage(CurrentDir() + "Images/BTree.bmp", 32, 32, 0, 2);burning
DTreei = LoadAnimImage(CurrentDir() + "Images/DeadTree.bmp", 32, 32, 0, 1);dead
Cursori = LoadImage(CurrentDir() + "Images/Cursor.bmp")
Raini = LoadAnimImage(CurrentDir() + "Images/Rain.bmp", 32, 32, 0, 2)
Compassi = LoadAnimImage(CurrentDir() + "Images/Compass.bmp", 100, 100, 0, 8)
;Terrain effecting images
MapA = LoadImage(CurrentDir() + "Images/MapA.bmp")
TopoA = LoadImage(CurrentDir() + "Images/MapA.bmp")

HandleImage TopoA, 0, 0
;;Pay attention to these
Type Tree
Field X, Y, image, health, onfire, ID, catchprob#, Frame, kind, Elevation;onfire- 1 burning, 0 not burning; kind- kind of try
End Type
```

```
Type Wind
Field Direction, MPH, Image ;Direction is 1-8, starting with North, going to NE
End Type
;;

;;Ignore these
Type TrackBlaze
Field BX, BY, EX, EY, Red, Green, Blue, TotFrame
End Type

Type BlueSquare
Field X, Y
End Type

Type Cursor
Field Image
End Type

Global onfi = 0, ID = 0;onfi- whether trees are on fire or not
Global timer = 0, LKey = 38, LineDisp = 1, TotFrame = 0, fireout = 0, colorsassigned, Greencol, RKey = 19
Global Raining = 0, RainFrame = 0, rainx = 0, rainy = 0

Cursor.Cursor = New Cursor
Cursor\Image = Cursori
HandleImage Cursor\Image, 0, 0

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;
;Here to change variables
```

```
Global Wind.Wind = New Wind
Wind\Image = Compassi
HandleImage Wind\Image, 0, 0
Wind\Direction = 2;Rand(1,8)
Wind\MPH = 25
;;;;;;;;;;;;;;;;;;;;
Global treecount = 800 ;how many trees there are
Global LeftX = 0, RightX = ImageWidth(MapA) ;1024 ;dimensions of forest
Global TopY = 0, BottomY = ImageHeight(MapA);768
Global catchprobability# = .75 ;likelihood of trees catching fire
;How far it can burn in various directions (neutral settings)
Global Rightburn = 50
Global LeftBurn = 50
Global UpBurn = 50
Global DownBurn = 50
WindDist()
;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;;;;;;;;;;(experimental)
;make trees grow on image
;;;;;;;;;;;

;CopyPixel xpix, ypix, FrontBuffer
HandleImage MapA, 0, 0
DrawImage MapA, 0, 0



;make the trees
For tr = 0 To treecount
```

```
    Tree.Tree = New Tree
    ;;;;;;;;;;
    Tree\X = Rand(LeftX, RightX)
    Tree\Y = Rand(TopY, BottomY)
    ;drawtree = 0
    ;For ypix = 0 To BottomY
    ;For xpix = 0 To RightX
    ;GetColor xpix, ypix
    ;If ColorRed() = 255; And ColorGreen() < 255 And ColorBlue() < 255
    ;drawtree = 1
    ;EndIf
    ;Next
    ;Next
    ;If drawtree = 0
    ;Delete Tree
    ;EndIf
    ;;;;;;;;;;;
    Tree\Image = Treei
;       If onfi = 0
;       Tree\onfire = 1
;       Tree\Image = BTreei
;       onfi = 1
;       BlueSquare.BlueSquare = New BlueSquare
;       BlueSquare\X = Tree\X; + 13
;       BlueSquare\Y = Tree\Y; + 13
;       EndIf
    Tree\health = 50 ;will be adjusted depending on tree\kind
    Tree\ID = ID
    Tree\Catchprob# = catchprobability
    ID = ID + 1
    Tree\Frame = 0
```

```
        Tree\Elevation = 0 ;(standard)
   DrawImage Tree\Image, Tree\X, Tree\Y
Next
ID = 0

For Tree.Tree = Each Tree
        .redotreeposition
        If ImagesCollide(Tree\Image, Tree\X, Tree\Y, Tree\Frame, MapA, 0, 0, 0)
         Tree\X = Rand(LeftX, RightX)
         Tree\Y = Rand(TopY, BottomY)
         Goto redotreeposition
        EndIf
Next

;Cls
MaskImage TopoA, 255, 255, 0
DrawImage TopoA, 0, 0
Flip
For Tree.Tree = Each Tree
GetColor Tree\X, Tree\Y
Tree\Elevation = ColorRed()
Next

;SetBuffer ImageBuffer()

For x = LeftX To RightX
For y = TopY To BottomY
GetColor x, y
If ColorRed() = 50 Or ColorRed() = 100  Or ColorRed() = 150 Or ColorRed() = 200 Or ColorRed() = 250 Or ColorRed() = 255
Color 255, 0, 0
```

```
Plot x, y

EndIf

Next
Next

WaitKey

SetBuffer BackBuffer()

;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;
;mainloop
;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;
While Not KeyDown(1)
Cls
timer = timer + 1

If KeyHit(LKEY)
LineDisp = LineDisp + 1
If LineDisp > 1
LineDisp = 0
EndIf
EndIf

If KeyHit(1)
End
EndIf
```

```
If MouseHit(1)
For Tree.Tree = Each Tree
If (MouseX() > Tree\X - 16) And (MouseX() < Tree\X + 16) And (MouseY() < Tree\Y + 16) And (MouseY() > Tree\Y - 16)
Tree\Onfire = 1
Tree\Image = BTreei
fireout = 0
Exit
EndIf
Next
EndIf

If KeyHit(RKEY)
raining = raining + 1
If raining = 2
raining = 0
EndIf

EndIf

;If enough time has elapsed, keeps it from going to fast.
If timer > 25
timer = 0
If colorsassigned = 0 And fireout = 0
TotFrame = TotFrame + 1
EndIf

If fireout = 0
fireout = 1
For t = 0 To treecount
For Tree.Tree = Each Tree
```

```
If Tree\ID = ID
trx = Tree\X;making variables to use
try = Tree\Y
tro = Tree\Onfire
trv = Tree\Elevation
If Tree\onfire = 1
fireout = 0
EndIf
Exit
EndIf
Next
ID = ID + 1
If ID > treecount
ID = 0
EndIf
;make others burn
If tro = 1
For Tree.Tree = Each Tree
    If Tree\onfire = 0 And Tree\Health > 0
    ;Determines Probability
    catchprob# = DetermineProbability(trx, try, trv, Tree\X, Tree\Y, Tree\Elevation, Tree\Catchprob#, 50)
     catch# = Rnd(0, 1)
     If catch <= catchprob
     Tree\onfire = 1
     TrackBlaze.TrackBlaze = New TrackBlaze
     TrackBlaze\BX = trx; + 16
     TrackBlaze\BY = try; + 16
     TrackBlaze\EX = Tree\X; + 16
     TrackBlaze\EY = Tree\Y; + 16
     TrackBlaze\TotFrame = TotFrame
```

```
             EndIf
            EndIf
Next
EndIf
Next
EndIf

For Tree.Tree = Each Tree
;tree images
If Tree\onfire = 1 And Tree\health > 0
Tree\Image = BTreei
Tree\Health = Tree\Health - 1
ElseIf Tree\onfire = 1 And Tree\Health = 0
Tree\Image = DTreei
Tree\onfire = 0
Tree\Frame = 0
EndIf
Next
;endif timer
EndIf

DrawImage TopoA, 0, 0

;draw the trees
For Tree.Tree = Each Tree
 If Tree\onfire = 1 And (timer = 6 Or timer = 12 Or timer = 18 Or timer = 24)
            Tree\Frame = Tree\Frame + 1
            If Tree\Frame > 1
            Tree\Frame = 0
            EndIf
 EndIf
```

```
MaskImage Tree\Image, 255, 255, 255
DrawImage Tree\Image, Tree\X, Tree\Y, Tree\Frame
;Color 0, 0, 255
;Text Tree\X, Tree\Y, Tree\Elevation
Next

If Raining = 1
;If timer = 5 Or timer = 10 Or timer = 15 Or timer = 20 Or timer = 25
;RainFrame = RainFrame + 1
;EndIf
;If RainFrame = 2
;RainFrame = 0
;EndIf
rainx = rainx - Rand(1, 1)
rainy = rainy + Rand(2, 2)
TileImage Raini, rainx, rainy, RainFrame
EndIf


If fireout = 1
colorsassigned = 1
EndIf

;Draw Windspeed
Color 0, 0, 255
Text 10, 634, Wind\MPH + "mph"
DrawImage Wind\Image, 10, 650, Wind\Direction-1


;Rect Bluesquare\X, BlueSquare\Y, 6, 6
Text 900, 10, TotFrame
```

```
DrawImage Cursor\Image, MouseX(), MouseY()
Flip



Wend

Function DetermineProbability#(BX,BY,BE,EX,EY,EE,Prob#,Health)
 ;;;;;;;;;;;;;;;;;;;;;;;;;;
 ;Adjust Prob here
 ;;;;;;;;;;;;;;;;;;;;;;;;;;
 If EX < BX + RightBurn And EX > BX - LeftBurn And EY < BY + DownBurn And EY > BY - UpBurn
  Prob# = WindProb(EY,BY,EX,BX,Prob)
  Prob# = DistanceProb(Prob)
  Prob# = RainProb(Prob)
  Prob# = TopoProb(Prob, BE, EE)
  catchproba# = Prob#/Health; to make the probability spread out over the tree's life
 Else
  catchproba# = -1
 EndIf

Return catchproba#
End Function



;;;;;;;;;;;;;;
;Adjusts how far trees have to be away from burning tree to catch fire
;;;;;;;;;;;;;;;
Function WindDist()
```

```
End Function

;;;;;;;;;;;;;;;
;Adjusts probability depending on wind
;;;;;;;;;;;;;;;;
Function WindProb(EY,BY,EX,BX,Proba#)
;North
If Wind\Direction = 1
If BY > EY
Proba# = Proba# * 2000
Else
Proba# = Proba# * 1/2000
EndIf
EndIf
;NE
If Wind\Direction = 2
If BY > EY And BX < EX
Proba# = Proba# * 2000
Else
Proba# = Proba# * 1/2000
EndIf
EndIf
;East
If Wind\Direction = 3
If BX < EX
Proba# = Proba# * 2000
Else
Proba# = Proba# * 1/2000
EndIf
EndIf
```

```
;SE
If Wind\Direction = 4
If BX < EX And BY < EY
Proba# = Proba# * 2000
Else
Proba# = Proba# * 1/2000
EndIf
EndIf
;South
If Wind\Direction = 5
If BY < EY
Proba# = Proba# * 2000
Else
Proba# = Proba# * 1/2000
EndIf
EndIf
;SW
If Wind\Direction = 6
If BY < EY And BX > EX
Proba# = Proba# * 2000
Else
Proba# = Proba# * 1/2000
EndIf
EndIf
;West
If Wind\Direction = 7
If BX > EX
Proba# = Proba# * 2000
Else
Proba# = Proba# * 1/2000
EndIf
```

```
EndIf
;NW
If Wind\Direction = 8
If BX > EX And BY > EY
Proba# = Proba# * 2000
Else
Proba# = Proba# * 1/2000
EndIf
EndIf

Return Proba#
End Function

;;;;;;;;;;;;;;;
;Adjusts probability depending on distance between trees
;;;;;;;;;;;;;;;;
Function DistanceProb(Proba#)

Return Proba#
End Function



;;;;;;;;;;;;;;
;Adjusts Probability depending on whether its raining
;;;;;;;;;;;;;;;
Function RainProb(Proba#)

Return Proba#
End Function

;;;;;;;;;;;;;;
```

```
;Adjusts Probability depending on changes in altitude
;;;;;;;;;;;;;;;
Function TopoProb(Proba#, BE, EE)
If EE > BE
Proba# = Proba*1000
ElseIf EE < BE
Proba# = Proba#/1.9
EndIf
Return Proba#
End Function
```