MONSOON RAINS IN THE

THE NEW MEXICO SUPERCOMPUTING CHALLENGE

SOUTHWESTERN U.S.

Team 83
Oñate High School
Final Report
April 4, 2007

TEAM MEMBERS
Brittany Atteberry
Jason Li
Petrina Strader
Erica Swenson
Natalie Wells

SPONSORING TEACHERS
Donald Downs
Josefina Domínguez

Table of Contents

Executive Summary

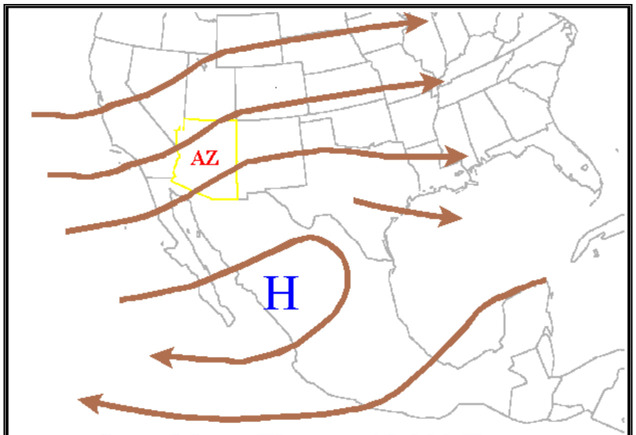
A much ignored fact about the American Southwest is that it experiences a monsoon circulation. Characterized by a shift in wind direction, a wet summer season, and a dry winter, our region's farming economy relies heavily on the timing and variance of the rains it brings, which influences all of crop production. A quick overview of the science of monsoons: During the scorching summers, the southwestern deserts heat up more quickly than the surrounding Pacific Ocean and Gulf of Mexico. This hot air rises and creates an area of low pressure, and moist air rushes in from the oceans to replace it. Inundated with moisture, the skies over New Mexico, Arizona, Utah, and Colorado (the Four Corners states) feature sudden and intense rainstorms in the summer months. Because of its importance to our farming industry, we decided to explore the possibility of harsher monsoon rains due to increased land temperatures. Though we attribute these elevated temperatures to global warming, it is not the problem of global warming that we attempt to solve; we merely apply its effects to our model. We hypothesized that more intense land heating would cause a stronger inflow of moist air, which would enable more thunderstorms to form, resulting in increased rainfall. The model we created is a one dimensional energy balance model, which calculates incoming and outgoing radiation. Our model simulates the monsoon season by adjusting the specific heat of the air, and simulates increased carbon dioxide levels by changing the air's emissivity. We obtained a contour elevation map, and drew a vertical line along the border of the Four Corners states from Mexico to Wyoming. We placed points of latitude along that line, and found the elevation and latitude of each of those points. Furthermore, our model features an air plate and a land plate in order to simplify the relationship between the Earth and the atmosphere. Running our program for a year for each latitude point, we found that increased emissivity in turn raised surface temperature, especially in the four corners area. These results supported our hypothesis and our assumption that more carbon dioxide increases air emissivity.

Introduction

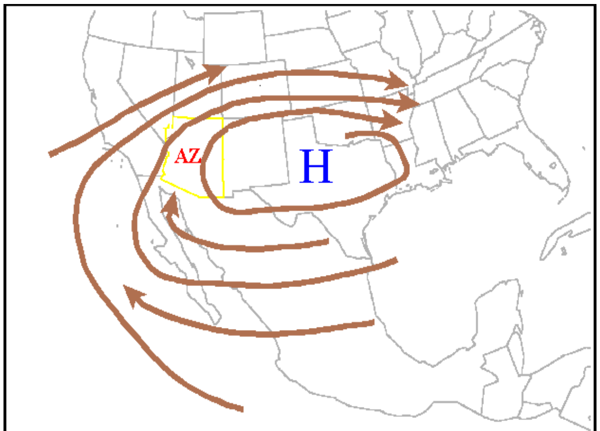
The monsoon season has an enormous influence on our local culture and economy. The smallest irregularity in the timing and variance of rain can affect all of crop production. Local farmers regularly irrigate their crops, so too little rain for a summer would not be a serious threat. However, if it rains too much, plants will rot and that season's crop will go to waste. Though this is tolerable for one season, a few successive years of more intense summer rains would obviously pose a serious problem to the local economy of our farmers. With this knowledge, we created a model that displays the effect of global warming on surface temperatures and thus the monsoon season in order to explore the possibility of harsher summer rains due to elevated land temperatures.

Monsoons

What is a monsoon? The word "monsoon" comes from the Arabic "mausin" which means "season of winds." It earns this name because the monsoon season features a signature shift in wind direction. So, the term monsoon does not describe the rains that this shift brings, but the actual wind reversal itself. Monsoon circulations occur in India, Australia, and Africa. In the case of the North American Monsoon, winds shift from a northerly or northwestern direction to a southerly or southeastern direction. The ensuing surge of moisture from the Gulf of Mexico and the Pacific is due to the uneven heating that takes place over the ocean and land masses involved. Water has a higher specific heat than the materials found on land, and thus has a higher heat capacity. As a result, land is heated by the sun more quickly and can reach a higher temperature than the oceans bordering it. Here the wind shift occurs. The heated air over the arid deserts rises, creating an area of low pressure, called a thermal low. Moist air from the gulfs surges in to replace it. In addition, a subtropical high pressure system, known as the Bermuda high, moves northward from northern Mexico to rest over Texas. The following pictures illustrate the movement of this subtropical high pressure system.



June Mean Flow at 18,000 Feet



July Mean Flow at 18,000 Feet

Global Warming

Global warming is not the problem we are solving. We are simply applying its effects to our working model by influencing the properties of the atmosphere and observing any changes it makes to our results. A human-enhanced greenhouse effect is generally believed to be the cause of global warming. It is important to note that the greenhouse effect happens naturally, and without it the average earth temperature would be about 30 degrees Celsius lower than it actually is. The primary natural greenhouse gases are water vapor and carbon dioxide, the former composing 0 to 4% of the atmosphere and the latter 0.04%. Water vapor accounts for 36% of the greenhouse effect, and carbon dioxide accounts for 12%. However, humans have been releasing increased amounts of carbon dioxide into the atmosphere, so the theory holds that more outgoing radiation will be trapped and global temperature will rise.

Rain Formation Process

The sun emits shortwave solar radiation, which the Earth receives, although some is reflected by the atmosphere, clouds, or the surface itself. Albedo is the reflectivity of a surface; it represents the fraction of solar radiation that is reflected right back into space unchanged. The higher the albedo value, the more radiation it reflects. The Earth in turn emits long wave thermal infrared radiation, some of which is reflected by water vapor and carbon dioxide in the atmosphere. When moist air moves into the low pressure area above the intensely heated desert lands, it expands, rises, and begins to cool, a process known as adiabatic cooling. An unsaturated parcel of air (a parcel that has not yet condensed) will cool as it rises according to the dry adiabatic lapse rate, which is the negative of the rate a parcel of air cools as it rises in height.

"Adiabatic" rules that no heat will be gained or lost outside the parcel. As the moist air expands and reaches a high enough elevation, it reaches its dew point and condenses. Ice crystals form, droplets grow rapidly around them, and we have cloud formation (called the Bergeron Process). Finally, the water droplets reach a large enough size to fall to earth, and precipitation occurs.

Problem Definition

We utilize a one-dimensional energy balance model to calculate incoming and outgoing radiation at any day at any time at any latitude. This computational model incorporates Euler's Method in order to solve differential equations numerically. We simulate the monsoon season by adjusting the specific heat of the air due to a greater influx of moisture, and simulate increased carbon dioxide levels by changing the air's emissivity.

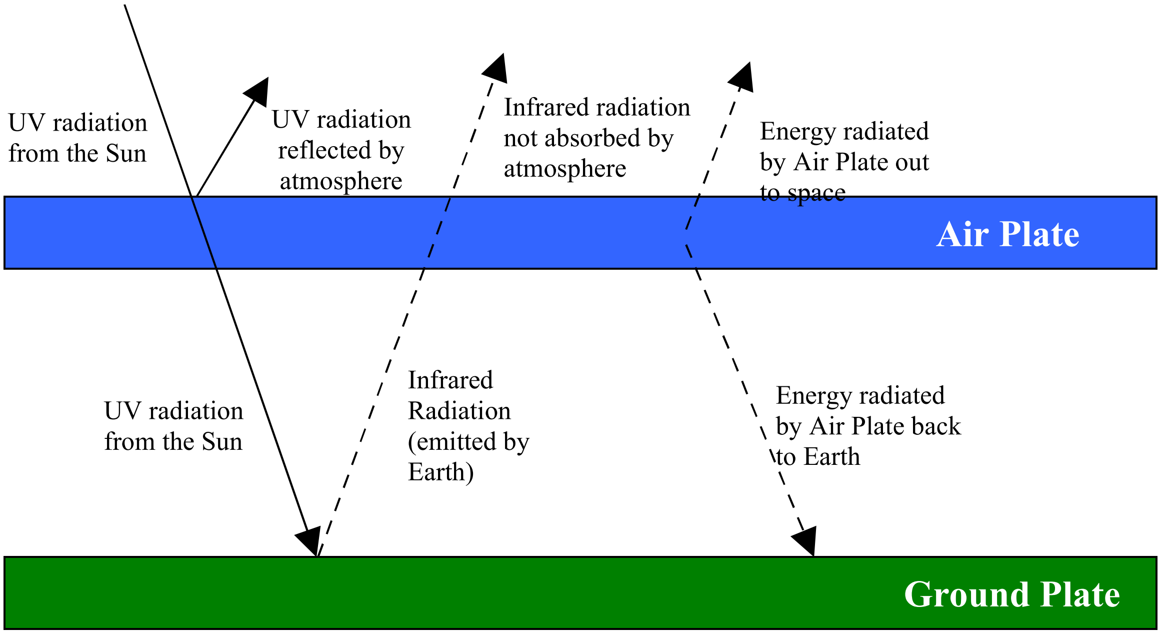
Project Set-Up

We began by obtaining a contour elevation map of the Four Corners states (Colorado, New Mexico, Utah, Arizona), the area where the monsoon circulation predominantly occurs and thus the area that it most affects. The map reflects a change in elevation about every 550 meters. These changes are represented by colored zones. It is important to note that all climate models generally deal with very coarse data, especially when dealing with elevation. This is because the modeling process is so large that detailed information processing can take up much time and effort. With a map in our hands, we then drew a vertical line on the map traveling along the border of the four states, which we will call the Border Line. We placed 115 points along this line spaced approximately 8.0465 meters apart. We used the map's elevation data to create a table recording the elevation of every point on the Border Line. In order to find the latitude of each point, we found the actual latitude of the points on the southern border of New Mexico and the northern border of Colorado. We found the difference between these two latitude values and divided that number by the number of points between the borders, which came out to .08406 degrees latitude. This value applied for the distance between all the points because all the points were spaced the same distance apart on the actual map (6.5mm). We obtained the actual latitude of the southern border of New Mexico, and added .08406 degrees to get the latitude of the second point. We added .08406 again to find the third point, and so on. With a table for the Border Line recording elevation and latitude for each point, we were ready to start programming.

Assumptions

We made several assumptions in order to simplify our model:

- The Earth's surface is a plate that emits and absorbs radiation.
- The air is also a plate, emitting and absorbing radiation.
- No latitudinal zone influences its neighboring zones.
- Increased carbon dioxide levels increases emissivity.



Hypothesis

Increased summer air temperatures in the Four Corners region will result in stronger monsoonal rains due to lower pressure over the intensely heated deserts, which will draw in more moisture from the oceans. Elevated carbon dioxide levels will contribute to this.

Program Initialization

We created a one dimensional energy balance model. This computational model calculates rates of incoming and outgoing radiation on any day and at any time at any latitude. Furthermore, we established a latitudinal 14° Celsius lapse rate, which we established by comparing the average January temperatures of the Mexican and Wyoming borders.

Step 1: Calculate the surface temperature for January 1st for each latitude point based on 14° Celsius lapse rate.

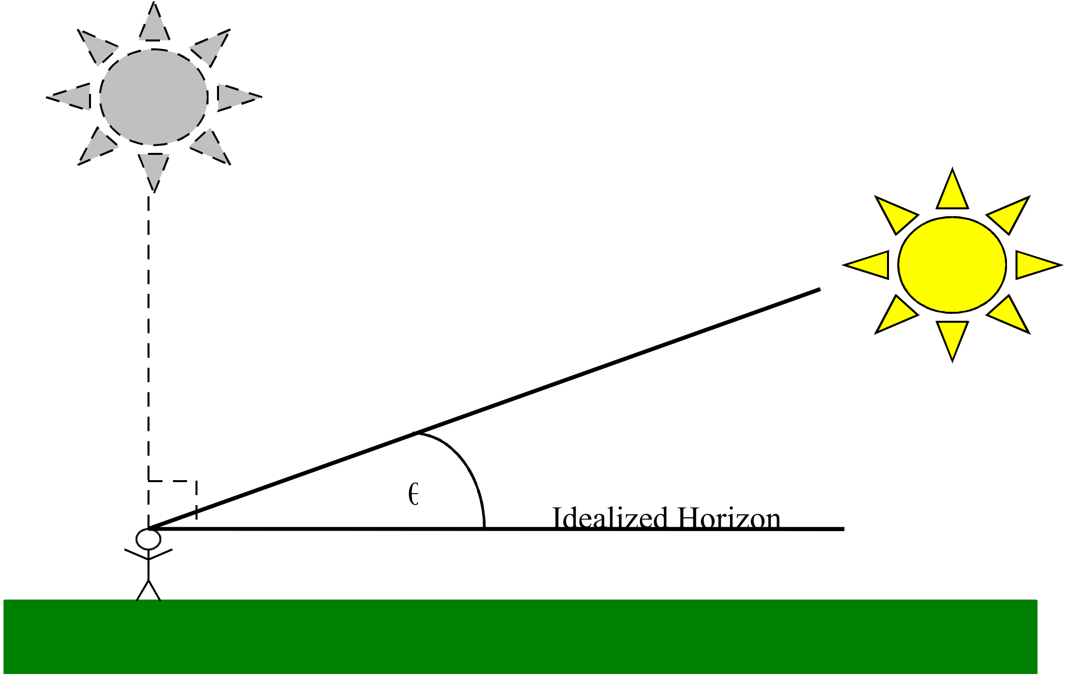
Step 2: Calculate the air temperature for each latitude based on the dry adiabatic lapse rate of 9.8° Kelvin per kilometer of elevation.

Step 3: Calculate the atmospheric pressure of each latitude zone, dependent on air temperature and altitude.

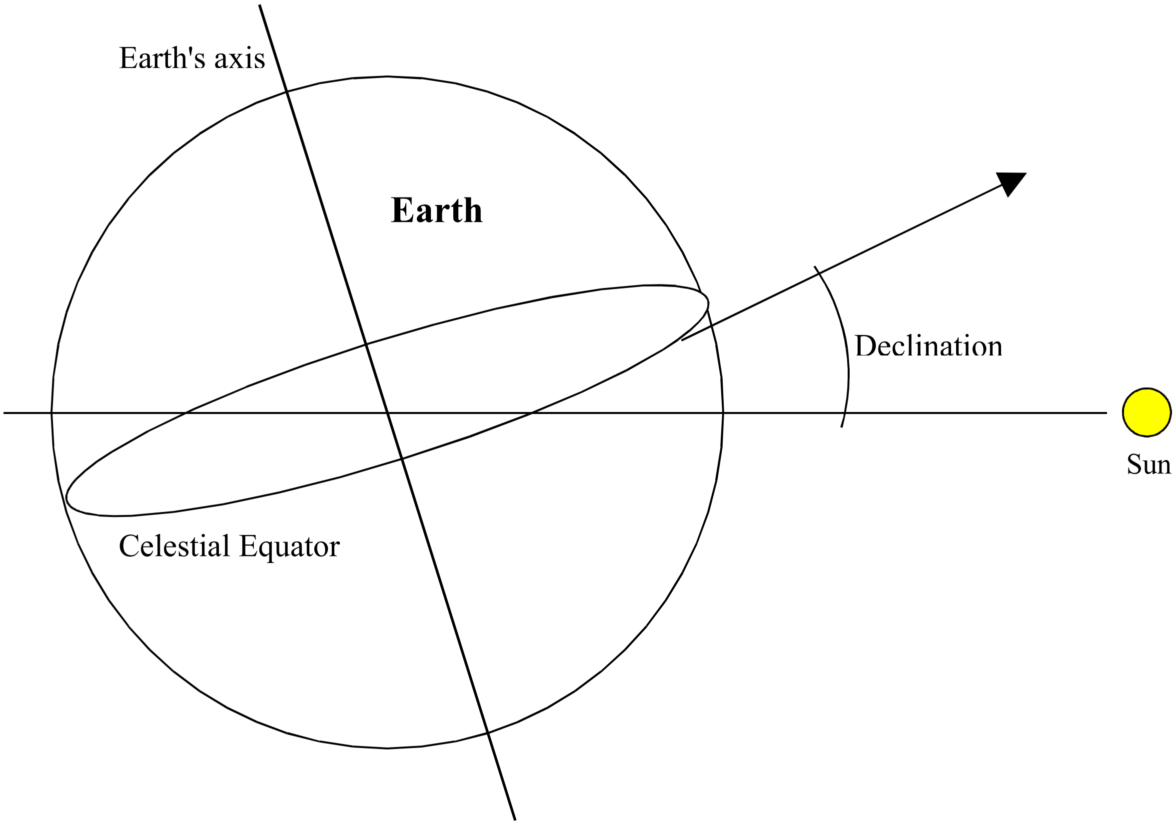
Main Loop

This loop runs an entire year for each latitude zone, each being 1 square meter in size.

Step 4: Calculate the angle of the sun. The angle of the sun relates to the sun's declination and elevation. Solar elevation is the angle between the sun and the idealized horizon, so this angle varies at different times of the day. For example, it would be measured as 0° at sunrise and sunset, and 90° at noon at the equator, as shown below:

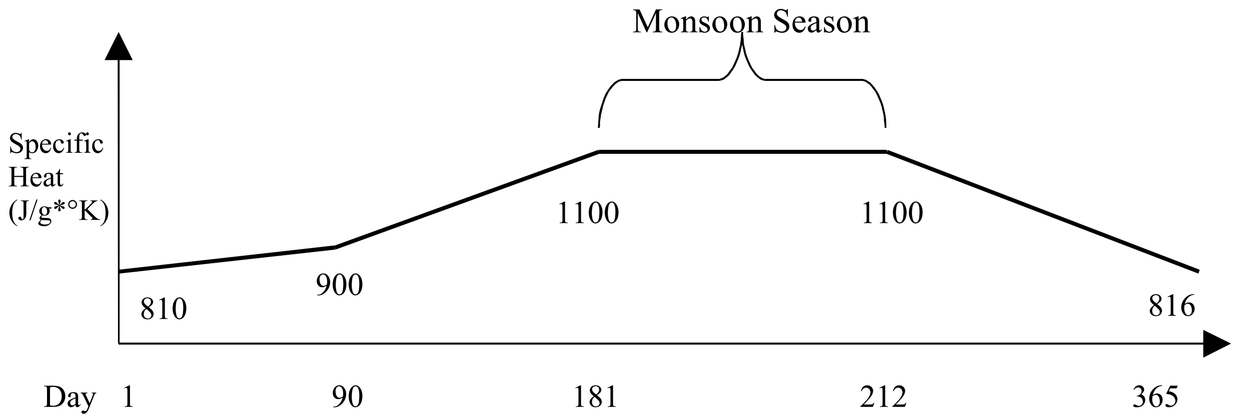


Declination also describes the angle of the sun, but according to the earth's equator. This angle varies little during the day, but much throughout the year. This greatly influences incoming solar radiation, because it determines how directly the rays of the sun will hit the Earth.



Step 5: Calculate incoming solar radiation.

Step 6: Calculate the air heat capacity for each latitude point. The air's heat capacity depends on the time of year, and we simulate the monsoon circulation by adjusting the specific heat of the air. The specific heat of a given amount of a substance is the measure of heat energy required to raise its temperature by one degree. Below is an example graph of the specific heat over a year for the Four Corners Region.



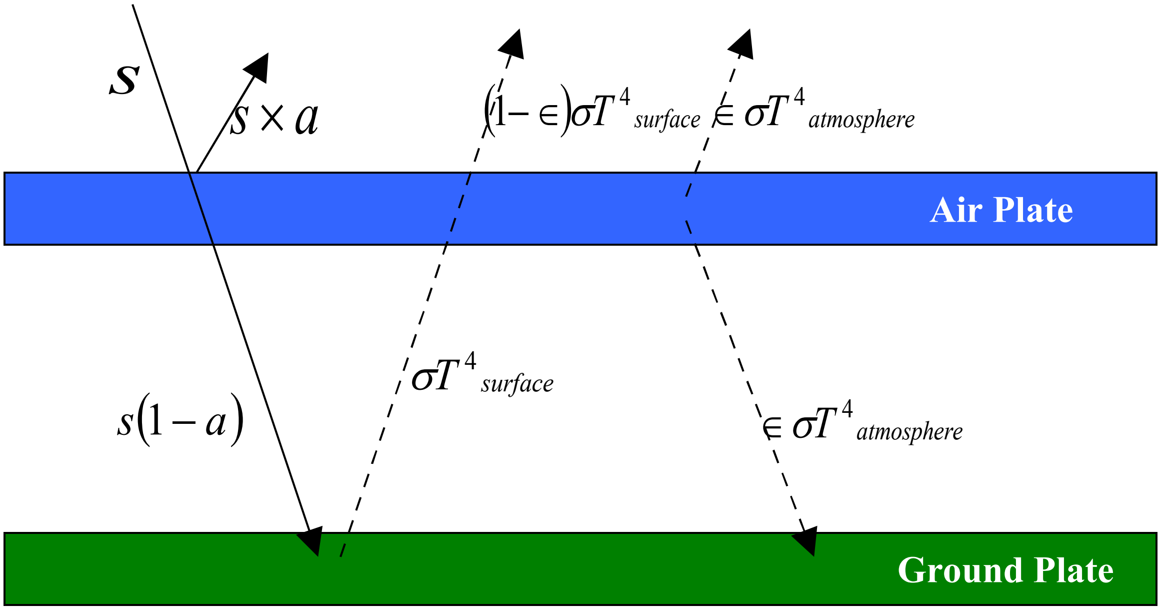
Step 7: Calculate the rate of incoming solar radiation reaching the surface of the Earth, as well as outgoing radiation being reflected from the air plate back to the surface. We accumulated this over time by multiplying by Δt (change in time). We used the Stefan-Boltzmann Law to convert the rate of incoming radiation to change in temperature (ΔT). The law states that the total energy that a blackbody radiates is proportional to the fourth power of that body's temperature. Then we adjust the surface temperature.

Step 8: All incoming radiation is in turn radiated back up to the air layer as infrared radiation. Calculate the rate of outgoing radiation and adjust surface temperature.

Step 9: Some of the infrared radiation that the Earth emits is absorbed by the air layer, and some passes through. This depends upon the emissivity of the air plate, which is influenced by the gases in the air (including carbon dioxide). The air's emissivity is the ratio of the energy it radiates to the energy that a blackbody radiates at the same temperature. It measures the ability of a substance to absorb and radiate energy. Update the air temperature.

Step 10: The air plate emits infrared radiation evenly in both directions (half into space, half back towards the Earth plate). Calculate these rates and adjust the temperature of the air layer.

Below is a picture that describes the process:

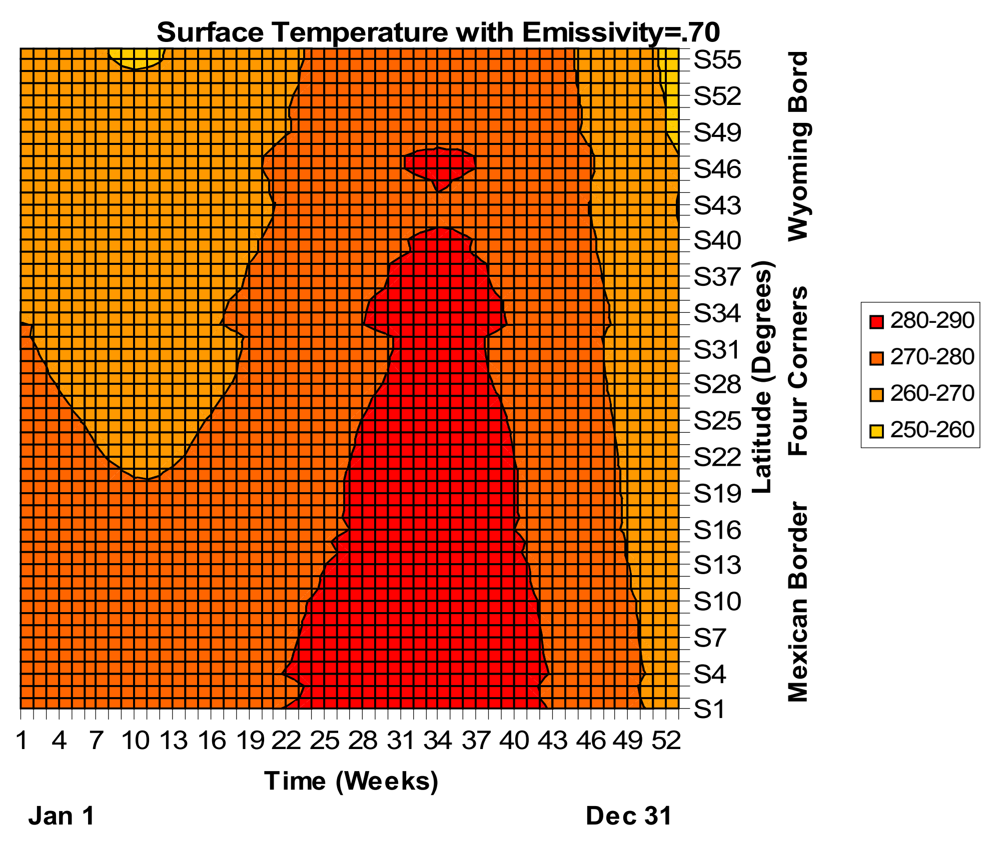


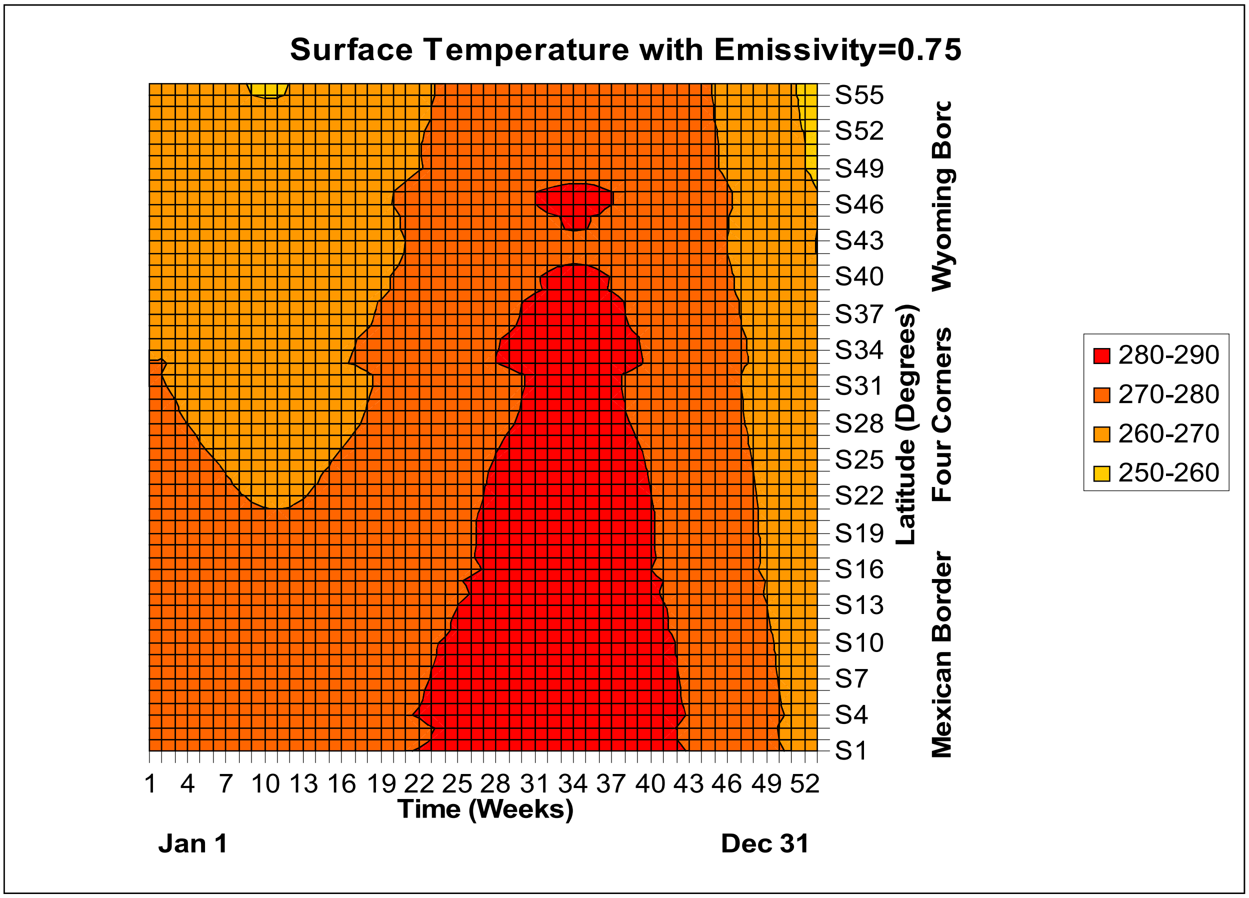
- s is rate of incoming solar radiation adjusted for time of day and day of year
- a is albedo
- σ is Stefan-Boltzmann Constant
- $\in$ is the emissivity
- $T_{surface}$ is the temperature of the surface in degrees Kelvin
- $T_{atmosphere}$ is the temperature of the atmospheric plate in degrees Kelvin

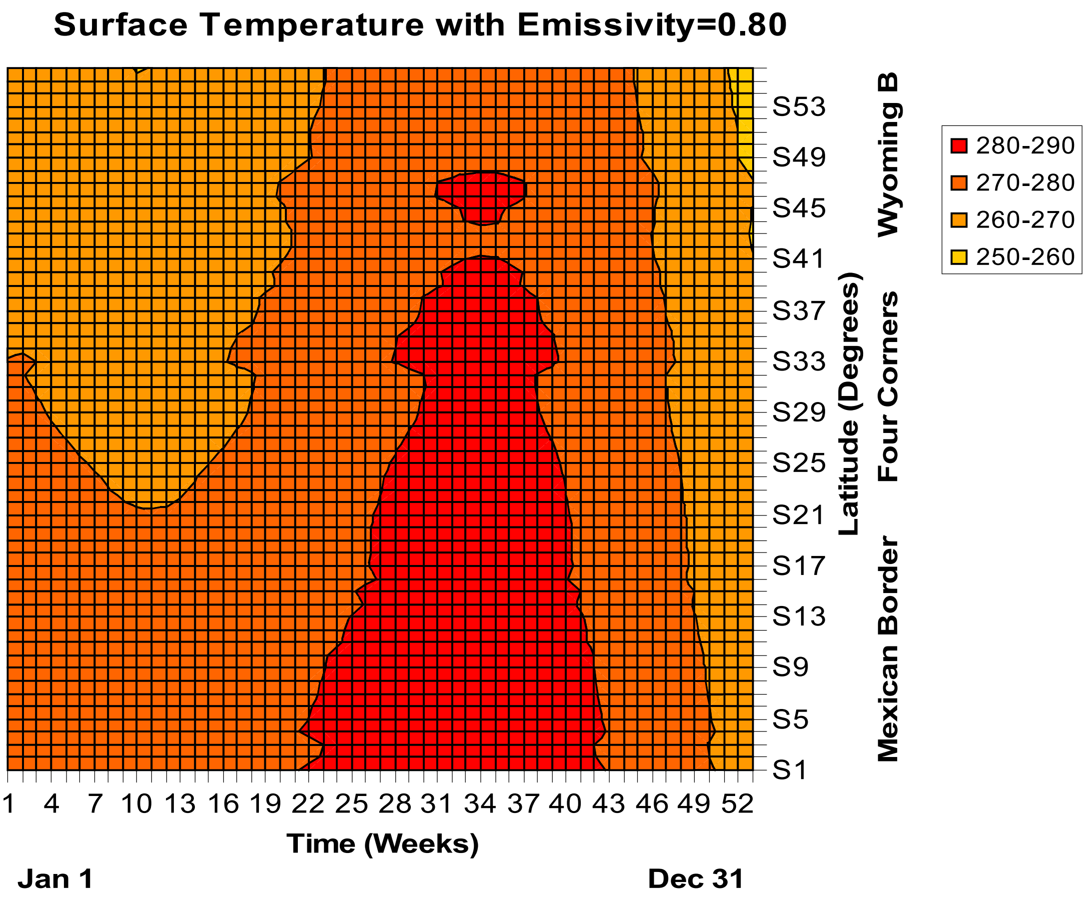
After the model works, we change the emissivity of the air layer to simulate increased levels of carbon dioxide, and run the model again.

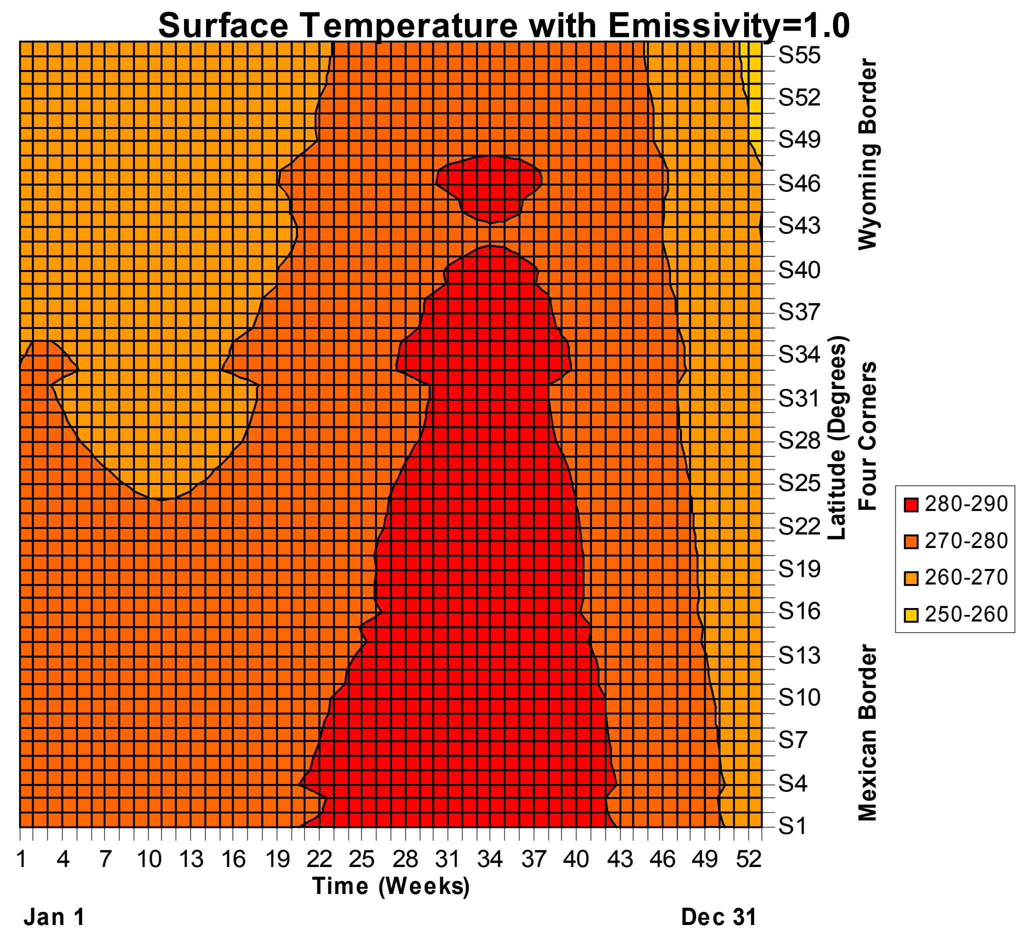
Results

The final results from our program came in the form of surface and air temperatures for each latitude zone. We established a base value of .75 for the emissivity of the air. This value was our control, as it is the Earth's average emissivity. We then raised it to .80 and monitored the surface temperature for any changes (we focused on the surface temperature more than the air temperature, as it is the intense land heating that attributes to the monsoon). The variances were miniscule, with temperatures rising only two hundredths of a degree in Kelvin. We then plunged into 1.0 emissivity (which is the value for a true black body), and once again only observed a small increase of 0.05° Kelvin. The following graphs illustrate our data. Though the graphs look similar, each graph is slightly affected by the change in emissivity. We observed a spike in temperatures corresponding to the months of the year that the monsoon rains predominantly occur. Also, we found that two bulges of hotter temperatures form, which correspond to lower elevations (Arizona and New Mexico have lower average elevations than Colorado and Utah as well). Our graphs also displayed steadily hotter temperatures as the latitude dropped.




Surface Temperature with Emissivity=0.75
S55
S52
S49
S46
S43
S40
S37
S34
S31
S28
S25
S22
S19
S16
S13
S10
S7
S4
S1
Latitude (Degrees)
Wyoming Borc
Four Corners
Mexican Border
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 52
Time (Weeks)
Jan 1
Dec 31
280-290
270-280
260-270
250-260


Surface Temperature with Emissivity=0.80
S53
S49
S45
S41
S37
S33
S29
S25
S21
S17
S13
S9
S5
S1
Latitude (Degrees)
Wyoming B
Four Corners
Mexican Border
280-290
270-280
260-270
250-260
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 52
Time (Weeks)
Jan 1
Dec 31


Surface Temperature with Emissivity=1.0
S55
S52
S49
S46
S43
S40
S37
S34
S31
S28
S25
S22
S19
S16
S13
S10
S7
S4
S1
Latitude (Degrees)
Wyoming Border
Four Corners
Mexican Border
280-290
270-280
260-270
250-260
1
4
7
10
13
16
19
22
25
28
31
34
37
40
43
46
49
52
Time (Weeks)
Jan 1
Dec 31

Conclusions

Though our results only showed a change of 0.05 degrees Kelvin at 1 emissivity, it is important to note that the Earths rising temperatures that have been observed only rise in increments of 0.2 degrees Celsius over decades. Our graphs also confirmed the influence of latitude on incoming solar radiation, for the higher latitudes (Utah and Colorado) received less heating. Furthermore, before a thunderstorm develops a cold layer of air acts as a chilly cap over the warmer surface air. If a section of surface air rises in temperature, it creates a "hole" in the cold air layer above it, and a thundercloud rapidly forms. We believe that a more emissive atmosphere would cause more of these hotspots that form into thunderstorms, and more storms means more rain more often. We conclude that as temperature over the Four Corners region increases due to increased atmospheric emissivity, the low pressure area during the summer will be more intense, and therefore will result in more intense and more frequent monsoon rains.

Recommendations

The air plate and earth plate concept greatly simplified our model, made it much more comprehensive, and did not detract from the validity of our model. Also, we initially created three longitudes with points of latitude along them: one traveling through the middle of Colorado and New Mexico, one along the border of the four states, and one through mid-Utah and Arizona. With the three longitudes, we would have included a larger range of elevations; however, for the sake of time, we were only able to focus on one line. Furthermore, we conducted an enormous amount of research and the project was still difficult and complex. With more time, we would extend the model so that neighboring latitude zones would influence one another, as they do in reality. We would also refine the numbers we used for specific heat, emissivity, and albedo, for these were more scientific guesses than the preferable confirmed experimental data for our region.

Achievements

We believe our most significant achievement was simply completing the project by the deadline. Attempting a climate model is a massive undertaking, and we were able to program our own original energy balance model using java in an academic year. Our research and understanding of weather processes made it possible for us to achieve this, as well as our inspired method of setting up the project using the contour map to establish the Border Line with latitudinal points and the concept of the air and ground plates. It was an extremely difficult project and required much sorting-out of a lot of information, for, as we learned, modeling Earth's processes is an incredibly complicated task, and the phenomena themselves are all the more intricately interwoven. Additionally, our team functioned more smoothly this year, and our increased communication helped us finish the project as well. We were able to promote each others strengths, and everyone made up an invaluable part of the team in order to reach a mutual goal.

Sources

Book Sources:

- Henderson-Sellers, Ann and Kendal McGuffie. A Climate Modeling Primer. Third Edition. West Sussex, England: John Wiley & Sons, Ltd., 2005.
- Bohren, Craig F. Atmospheric Thermodynamics. New York: Oxford University Press, Inc., 1998.
- Trenberth, Kevin E., ed. Climate System Modeling. New York: The Press Syndicate of the University of Cambridge, 1992.
- Bloomfield, Louis A. How Things Work: The Physics of Everyday Life. New York: John Wiley & Sons, Inc., 1997.
- McKnight, Tom L. and Darrel Hess. Physical Geography. Pearson Education, Inc., 2005.
- New Mexico Atlas & Gazetteer. First Edition. Yarmouth, Maine: DeLorme, 1998.
- Utah Atlas & Gazetteer. First Edition. Freeport, Maine: DeLorme, 1993.

Internet Sources:

- "Basics of the Arizona Monsoon and Desert Meteorology." AZ Thunderstorm Chase Project. Arizona State University. 30 November 2006. <http://geography.asu.edu/aztc/monsoon.html#monsoon>.
- "Climate Science Explained." ClimatePrediction.net. 28 October 2006. <http://climateprediction.net/index.php>.
- Dr. Mitchell, David. "DRI's Mexican Monsoon Page." Desert Research Institute. 19 November 2006. <http://www.dri.edu/Projects/Monsoon/>.
- "Earth Temperature and Site Geology." Geo4VA. Virginia Department of Mines Minerals and Energy. 14 March 2007. <http://www.geo4va.vt.edu/A1/A1.htm>.
- Schwartz, Stephen E. "Heat Capacity, Time Constant, and Sensitivity of Earth's Climate System." Brookhaven National Laboratory. 17 July 2006. 3 March 2007. <http://www.ecd.bnl.gov/pubs/BNL-76939-2006-AB.pdf>.
- "Image: Instrumental Temperature Record.png." Global Warming Art. Wikipedia. 2 April 2007. <http://en.wikipedia.org/wiki/Image:Instrumental_Temperature_Record.png>.

- Dr. Pidwirney, Michael. "Introduction to the Atmosphere." PhysicalGeography.net. 18 March 2007. Michael Pidwirney, University of British Columbia Okanagan. 13 Feb 2007. <http://www.physicalgeography.net/fundamentals/7a.html>.
- "Solar Radiation." Squ1.org. Square One Research. 1 February 2007. <http://squ1.org/wiki/Solar_Radiation>.
- “The Arizona Monsoon.” National Weather Service – NWS Phoenix. 2 April 2007. National Oceanic and Atmospheric Administration, U.S. Department of Commerce. 18 November 2006. <http://www.wrh.noaa.gov/psr/general/monsoon/>.
- “What is the Mexican Monsoon?” National Weather Service – NWS Tucson. National Oceanic and Atmospheric Administration, U.S. Department of Commerce. 18 November 2006. < http://www.wrh.noaa.gov/twc/monsoon/mexmonsoon.php>.

Other:

- Kann, Deirdre. "New Mexico Summer Precipitation." New Mexico Skywatcher. 2.2 (Oct 2006): 4.

Acknowledgments

Our team would like to thank Dr. Daniel Dugas, a professor in the Geography Department at New Mexico State University, who supplied us with the contour elevation map and a plethora of helpful suggestions and recommendations. We appreciate his tolerance of our incessant emails.

Secondly we would like to thank Dr. Jack Wright, Head of the Geography Department at New Mexico State University. We met with him as well as Dr. Dugas before we had decided on a problem for our project. He offered invaluable guidance and both professors helped us get to a healthy start.

Furthermore, Ted Sammis, New Mexico's State Climatologist, also answered many of our questions, and Eleanor Walther commented on our Challenge interim.

Lastly we thank Steven Ewing, a chemistry teacher at Oñate High School, and our sponsoring teacher, Donald Downs, who guided us throughout our entire project and without whom we would have drowned in the dark sea that is climate modeling.

Java Program

```
/*
 * Main.java
 *
 * Created on February 5, 2007, 12:04 PM
 *
 * To change this template, choose Tools | Template Manager
 * and open the template in the editor.
 */

package angleofincidence;

import java.io.*;
import java.util.Formatter;


public class Main
{
    public static final double SOLARCONSTANT=1370.0; /*This value is measured
    in watts per meter squared.*/

    public static final double STEFANBOLTZMANN=0.0000000567; /*This value is
    measured in watts squared per meter squared multiplied by the temperature,
    in Kelvin, to the fourth.*/


    /**************************************
    * Below are the altitudes for the 100 *
    * points of latitude on our line.     *
    **************************************/

    public static double altitude[]={1372, 1651, 2072, 2000,
    2100, 1500, 950, 1190, 1200, 1190, 1200, 1310, 1290, 1280,
    1290, 1340, 1320, 1340, 1280, 1828, 1750, 1651, 1700, 1750,
    1828, 2195, 2300, 2200, 1828, 2190, 2700, 2220, 2190, 2100,
    2100, 2150, 2000, 1900, 1850, 1800, 1850, 1900, 1950, 1850,
    1950, 1900, 1950, 2000, 2050, 2100, 2200, 2300, 2400, 2500,
    2650, 2700, 2750, 2600, 2700, 3650, 2700, 2500, 2550, 2920,
    1100, 1100, 1100, 1100, 1100, 1400, 1650, 1700, 1700, 1650,
    1650, 1670, 2200, 2300, 2000, 2000, 2400, 2500, 2700, 2750,
    2600, 2300, 2000, 2100, 1800, 1300, 1100, 1000, 1000, 1600,
    1800, 2000, 2800, 2700, 2500, 2400, 2300, 2400, 2400, 2200,
    2700, 2500, 2600, 2700, 2700, 2550, 2500, 2700};


    public static double atmosPressure[]=new double[altitude.length];

    public static double latitude[]=new double[altitude.length];


    /**************************************
    * All temperatures in our program are *
    * measured in Kelvin.                 *
    **************************************/
```

```
public static double surfaceTemp[]=new double[altitude.length];
public static double airTemp[]=new double[altitude.length];
public static double deltaTemp;

public static double albedo=.2;
public static double emissivity=0.70;
public static final double heatCapacitySurface=59287680;
public static double airHeatCap(double day, double atmosPress)
{
    /*******************************************
    * We can simulate the monsoon season in the  *
    * four corners area by manipulating the heat *
    * capacity of the air as shown below in the  *
    * if and else statements.                    *
    *******************************************/

    double airSpecificHeat;
    day=day%365; //This ensures that we never pass day 365.

    if(day<=90)
    {
        airSpecificHeat=816.0+(((900.0-816.0)/90.0)*(day));
    }

    else if(day<=151)
    {
        airSpecificHeat=900.0+(((1100.0-900.0)/61.0)*(day-90));
    }

    else if(day<=181)
    {
        airSpecificHeat=1100.0;
    }

    else if(day<=242)
    {
        airSpecificHeat=1100.0-(((1100-900.0)/31.0)*(day-181));
    }

    else
    {
        airSpecificHeat=900.0-(((900.0-816.0)/113.0)*(day-242));
    }

    double airHeatCapacity=atmosPress*airSpecificHeat;
    return airHeatCapacity;
}

/*******************************************
* The below method calculates the incoming *
* solar radiation on any given day of the  *
* year, calling a variable sun elevation.  *
*******************************************/

public static double incomingRadiation(double H)
{
```

```
        H=Math.toRadians(H);
        double incomingRadiation;
        if(H>0) {
            incomingRadiation=(SOLARCONSTANT*Math.sin(H));
        } else {
            incomingRadiation=0;
        }
        return incomingRadiation;
    }



    /*************************************
    * The below method gives us the sun’s *
    * declination on any day at any time, *
    * calling a variable day, which is    *
    * the number of days from the winter  *
    * solstice.                           *
    *************************************/

    public static double sunDeclination(double day)
    {
        double solsticeDec=Math.toRadians(23.5);
        double sunDeclination=(solsticeDec*(Math.sin(((day-81.0)/365.0)
                              *(2*Math.PI))));
        sunDeclination=Math.toDegrees(sunDeclination);

        return sunDeclination;
    }


    /************************************
    * The below method gives us the      *
    * elevation of the sun on any day    *
    * and time, calling variables of sun *
    * declination, latitude, and time.   *
    ************************************/

    public static double sunElevation(double latitude, double time, double 
delta)
    {
        delta=Math.toRadians(delta); //delta is the sun’s declination
        double beta=Math.toRadians(latitude);
        double tau=Math.toRadians((12.0-time)*15.0);


        double sunElevation=(Math.asin((Math.sin(beta)*Math.sin((delta)))
            +(Math.cos(beta)*Math.cos(delta)*Math.cos(tau))));
        sunElevation=Math.toDegrees(sunElevation);

        return sunElevation;
    }

    public static void main(String[] args) throws IOException, 
FileNotFoundException
    {
        /*************************************
```

```
        * The following code writes the data *
        * with which our program provides    *
        * us into a data file which can be   *
        * easily read into a program such    *
        * as Excel.                          *
        *************************************/

        PrintWriter outSurface=new PrintWriter(new
        FileOutputStream("SurfaceData.txt"), true);
        PrintWriter outAir=new PrintWriter(new
FileOutputStream("AirData.txt"), true);

        int j;
        for (j=0;j<surfaceTemp.length;++j)
        {
            surfaceTemp[j]=279.0 - (14.0/100.0)*j;
            airTemp[j]=surfaceTemp[j]-9.8*altitude[j]/1000.0;
        }

        double Po=10332.27; /*This value represents surface pressure.*/

        double g=9.807;     /*This value represents the dry vertical lapse
rate
                             in degrees Kelvin per kilometer.*/
        double R=287.04;    /*This value is the gas constant.*/

        int k;

        for(k=0;k<atmosPressure.length;++k)
        {
            double z=altitude[k];  /*This value z represents the altitude
            of the point at which we are finding the pressure.*/

            atmosPressure[k]=(Po*(Math.exp((-g*z)/(R*airTemp[k]))));
        }

        int i;
        for(i=0;i<latitude.length;++i)
        {
            double deltaT=0.1; /*The time is measured in hours.*/
            long ticksInWeek=Math.round(24.0*7/deltaT);
            double t;
            double day=1;
            latitude[i]=33.0+i*.08406;
            long count=0;

            for(t=0;t<=24*365;t+=deltaT)
            {
                double decl=sunDeclination(day+t/24.0);
                double elev=sunElevation(latitude[i], t%24.0, decl);
                double s=incomingRadiation(elev);

                double recipAirHeatCapacity=1.0/airHeatCap(day+t/24.0,
                                            atmosPressure[i]);
                double recipSurfaceHeatCapacity=1.0/heatCapacitySurface;
```

```
          /****************************************
          * The following calculates the change in *
          * surface temperature due to incoming    *
          * solar radiation and incoming radiation *
          * from the air plate.                    *
          ****************************************/

          deltaTemp=recipSurfaceHeatCapacity*(s*(1-albedo)

+emissivity*STEFANBOLTZMANN*(Math.pow(airTemp[i],4)))*deltaT*3600;
          surfaceTemp[i]+=deltaTemp;


          /****************************************
          * The following calculates the change in *
          * surface temperature due to outgoing    *
          * infrared radiation.                    *
          ****************************************/

          deltaTemp=recipSurfaceHeatCapacity*STEFANBOLTZMANN*
                  Math.pow(surfaceTemp[i],4)*deltaT*3600;
          surfaceTemp[i]-=deltaTemp;



          /******************************************
          * The following calculates the change in    *
          * air temperature due to incoming radiation *
          * from the surface.                         *
          ******************************************/

          deltaTemp=(emissivity)*recipAirHeatCapacity*STEFANBOLTZMANN*
                  Math.pow(airTemp[i],4)*deltaT*3600;
          airTemp[i]+=deltaTemp;


          /*****************************************
          * The following calculates the change in   *
          * air temperature due to outgoing infrared *
          * radiation from the air plate to space    *
          * and back to earth.                       *
          *****************************************/

          deltaTemp=2.0*emissivity*recipAirHeatCapacity*STEFANBOLTZMANN*
                  Math.pow(airTemp[i],4)*deltaT*3600;
          airTemp[i]-=deltaTemp;


          if(i%2==0 && (count%ticksInWeek==ticksInWeek/14))
          {
              outSurface.format("%7.2f  ", surfaceTemp[i]);
              outAir.format("%7.2f  ", airTemp[i]);
          }
          count++;

      }
```

```
            if(i%2==0)
            {
                outSurface.format("\n");
                outAir.format("\n");
            }
        }
        outSurface.close();
        outAir.close();
    }
}
```