

Auroral Alert

New Mexico Adventures in
Supercomputing Challenge

Final Report

March 16, 2004
Team 045

Manzano High School

Team Members

Sean Gentry
Jamal Osman
Matt Tlush

Teachers

Mr. Schum
Mr. Matthews

Table of Contents

- 1) Executive Summary

- 2) Main Body
 - a) Introduction
 - b) Description of the Problem
 - c) Results
 - d) Recommendations
 - e) Acknowledgements

- 3) Bibliography

- 4) Appendices
 - a) Appendix A: Program
 - b) Appendix B: Graph 1

Summary

You look up into the sky and see these swirling, glowing colors floating in the sky. “What are those?” You wonder. Well those colors are caused by highly charged protons and electrons ionizing the atmosphere which cause those colors to be displayed. The particles come from solar flares on the sun which, are in turn caused by the differences in rotations in the sun. The poles rotate faster than the equator because the sun is boiling gas, and isn’t solid. The differences in rotation speeds cause the sun’s magnetic field to be warped and break through the surface of the sun, causing sunspots. The fields continue to stretch and make solar flares by drawing pieces of the sun away from its surface. The particles interact with the magnetosphere which causes the magnetosphere to stretch three times longer than normal. The particles get drawn into the magnetic poles of the earth where the magnetosphere originates. Then the particles interact with the atmosphere and cause the Aurora Borealis.

Our project is to find out what size of solar flare will bring the aurora down to a certain latitude. That is what are code does.

Introduction

Our group has learned many important skills required for success in the Adventures in Supercomputing Challenge. We spent many weeks learning the general aspects of HTML. We spent two to three weeks setting up both MHS AiS Challenge Linux accounts and accessing our New Mexico AiS Challenge Unix accounts. We discussed and researched general math algorithms for computing distances and velocities in simple motion problems, volumes of rectangular and cylindrical railroad cars, and creating mailing record databases. We spent five weeks learning the basic concepts of C++ programming code. Combining the math algorithms with programming skills, we then wrote the C++ code to compute the physics problems of motion, and creating a mailing record database. Basically, our instructor provided us with nominal information and starting templates to write the program code for the above sample problems.

The purpose of this project, which has changed a number of times since the start back in September, is to create a mathematical model of the Auroral Borealis. On October 21 and the days following two very dynamic centers of activity emerged on the sun. One of the largest solar flares occurred causing radio blackouts and mild communication disruption. Associated with this solar flare was a cloud of energized particles known as a coronal mass ejection (CME). This CME, moving at over five million miles per hour impacted Earth's magnetic field causing a large Auroral Borealis. This is what spawned our idea for our project. We wanted to see if it was possible to mathematically model an

Auroral Borealis and what Intensity of CME would it take for the Auroral Borealis to Grow.

Description

To come up with a math model we used a chart we found on a NASA website. The chart had the Kp-index and the latitude. The Kp-index is a chart that scientists use to calculate the location of the auroral oval. From there we used Microsoft Excel to graph out the chart. Then we calculated a best-fit line equation, which we used for our code. For the code we entered a variable and the computer calculated the latitude for the given Kp.

Results

We have no results at this time. But we do have working code.

Recommendations

This project went through some drastic changes during the seven months since the start. Early on we hoped to have a great big project with all the bells and whistles. The truth is our project at the last moment has changed from what we hoped to have done in the beginning to a project that we crammed as much energy into just to get it finished. If we had more time we would have created a 3D model of our results. Also we would have liked to make the program more user friendly with more options to solve the problem.

Acknowledgments

We would like to thank Mr. Schum and Mr. Matthews for all the help they gave us in our project. Without them we would haven't received any training in using C++ and HTML.

Appendix A

```

/*Team 45: This program asks the user to input a Kp-index number
and then using a quadratic equation it displays the latitude of the Kp-
indices */
#include <windows.h>
#include <stdlib.h>
#include <iostream.h>
#include <math.h>
#include <fstream.h>
#include <iomanip.h>

void programoverview();
void calc();

int main()
{

    int kp,L;
    programoverview();
    calc();
    cout<<"\nPlease enter a K-indices of 0-9:"<<endl;
    cin>>kp;
    L=(-0.25*kp)*pow(0,2) + (0.8*kp) + (48.95);
    cout<<"The Latitude of your given K-indices is:"<<L<<" "<<"degrees
latitude"<<endl;
    MessageBox (NULL, "Press ok to view the excel graph" , "Auroral Alert",
0 + MB_OKCANCEL);
    system("pause");
    return 0;

}

void programoverview()
{
    cout<<"In this program you input a K-indices and the program outputs
the"<<
"\nlatitude that the auroral oval will be at."<<endl;
}

void calc()
{

}

```

Appendix B

