

The graph in Figure 1 below shows the average global temperature since the year 1880. The data points are global temperature deviations for the month of May, converted to Fahrenheit. Figure 2 shows the same data with the calculated best fit line. The coefficient calculated where $a = 0.0127$, $b = 56.357$. The mean squared error of the model is 0.099 . The best fit line was extended to the year 2033 (when I will turn 40) and the projected global temperature will be 58.3°F .

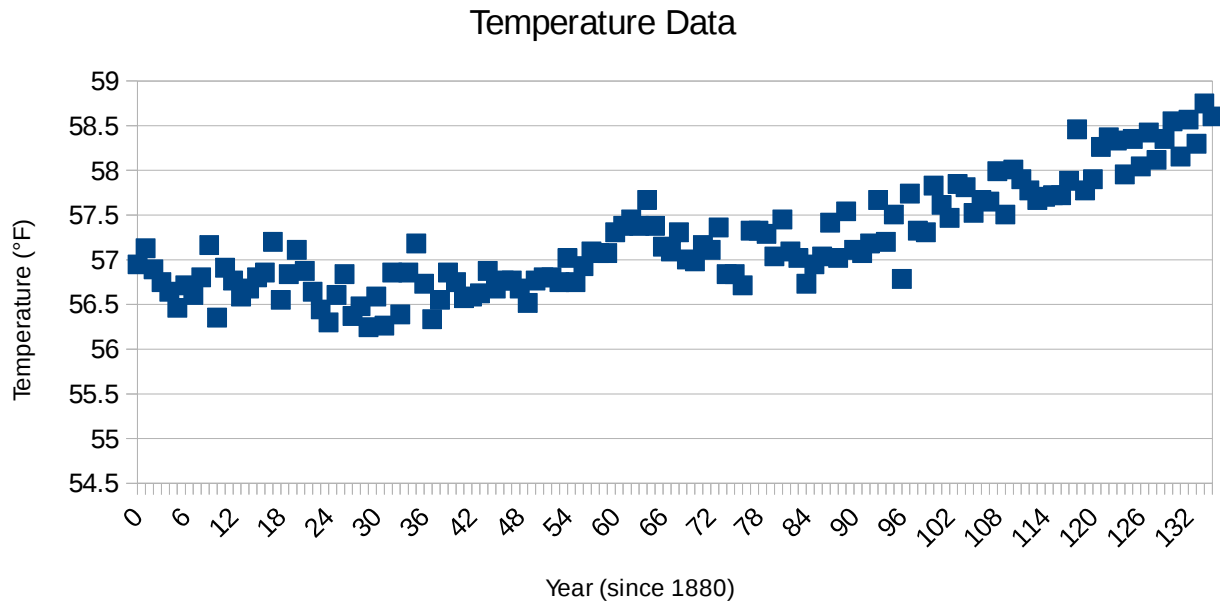


Figure 1: Global temperature averages from 1880 to present

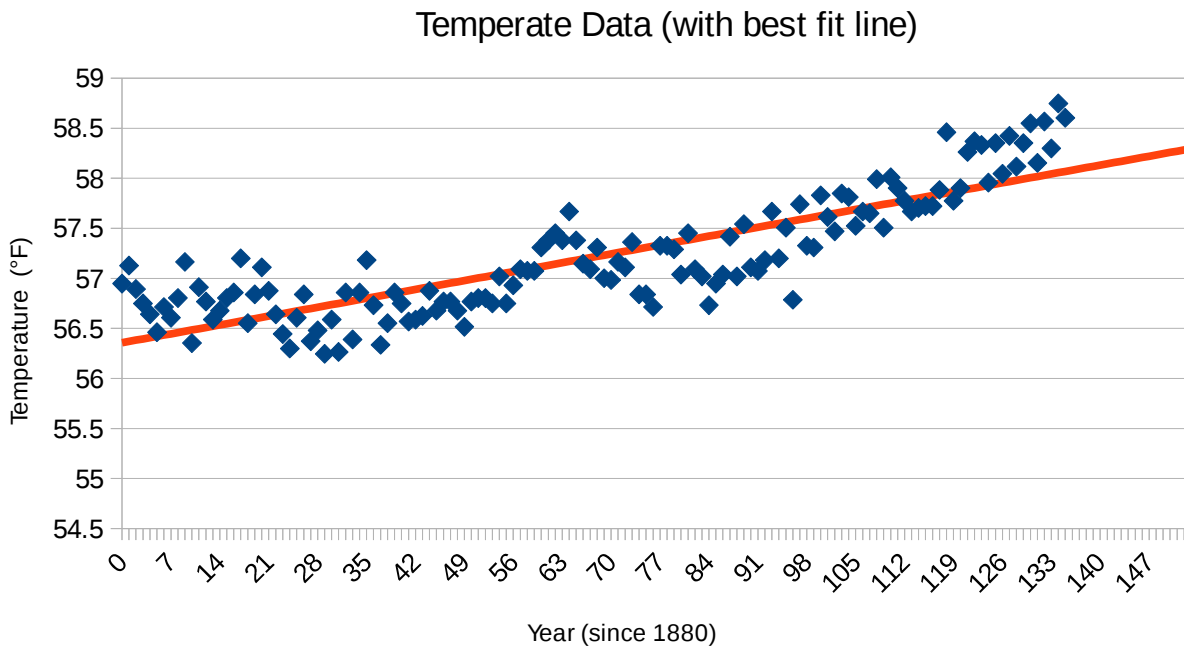


Figure 2: Temperature data with best fit line

Appendix: Program for calculating coefficients a & b

```
#include <iostream>
#include <fstream>
#include <string>
#include <sstream>
#include <stdlib.h>
using namespace std;

int main(){
    int N=0;
    double tempValues[256];
    string line;

    // 1. Read in the data file
    ifstream file;
    file.open("tempData.txt");
    while(getline(file, line)){

        // if line starts with a year value (1***, 2***)
        if(!line.substr(0,1).compare("1") || !line.substr(0,1).compare("2")){
            string token;
            istringstream iss(line);
            int i=0;
            // split the line into tokens on spaces
            while(getline(iss, token, ' ')){
                if(!token.empty()){
                    i++;
                    // 6th available token is 5th month (may)
                    if(i==6){
                        tempValues[N] = (((atoi(token.c_str())) / 100.0 + 14.0) *
(9.0 / 5.0)) + 32.0;
                        N++;
                    }
                }
            }
        }
    }

    // 3. Calculate the coefficients

    int i;
    double a    = 0.0;
    double b    = 0.0;
    double beta = 0.0;
    double delta = 0.0;
    double gamma = 0.0;
    double alpha = 0.0;
```

```

for(i=0; i<N; i++){
    alpha += (i*i);
}
alpha = alpha / (double)N;

for(i=0; i<N; i++){
    beta += i;
}
beta = beta / (double)N;

for(i=0; i<N; i++){
    gamma += (i * tempValues[i]);
}
gamma = gamma / (double)N;

for(i=0; i<N; i++){
    delta += tempValues[i];
}
delta = delta / (double)N;

// Solve for
// - alpha * a - beta * b + gamma = 0
// - beta * a -      b + delta = 0

double det = (-1 * alpha)*(-1) - (-1 * beta)*(-1 * beta);

a = ((gamma)*(-1) - (-1 * beta)*(delta)) / det;
b = ((-1 * alpha)*(delta) - (gamma)*(-1 * beta)) / det;
a*=-1;
b*=-1;
cout << "a = " << a << endl;
cout << "b = " << b << endl;

// Calculate mse
double mse = 0.0;
for(i=0; i<N; i++){
    mse += (tempValues[i] - a * i - b)*(tempValues[i] - a * i - b);
}
mse = mse / (double)N;
cout << "mse = " << mse << endl;

// Predict global average when I turn 40
double temp40 = a * (1993 + 40 - 1880) + b;
cout << "Prediction for global average temperate when I am 40: \n" << temp40 << "°F" <<
endl;
}

```