Name: _____

Vectors and Pointers

Problems to do in Lab. Do not report this.

Problem 1. Menu Driven Program, improved with vectors

The menu driven program must be improved this way:

- It must use vectors, instead of arrays. So, there is no limit for the size.
- It must include an option to display students on a range of ages.

To be reported on canvas. Create a PDF. Include screenshots of code and execution. Include copy-pasteable text of code. Be careful with variable names and indentation. You must use the templates.

Problem 2. Number Analysis Program

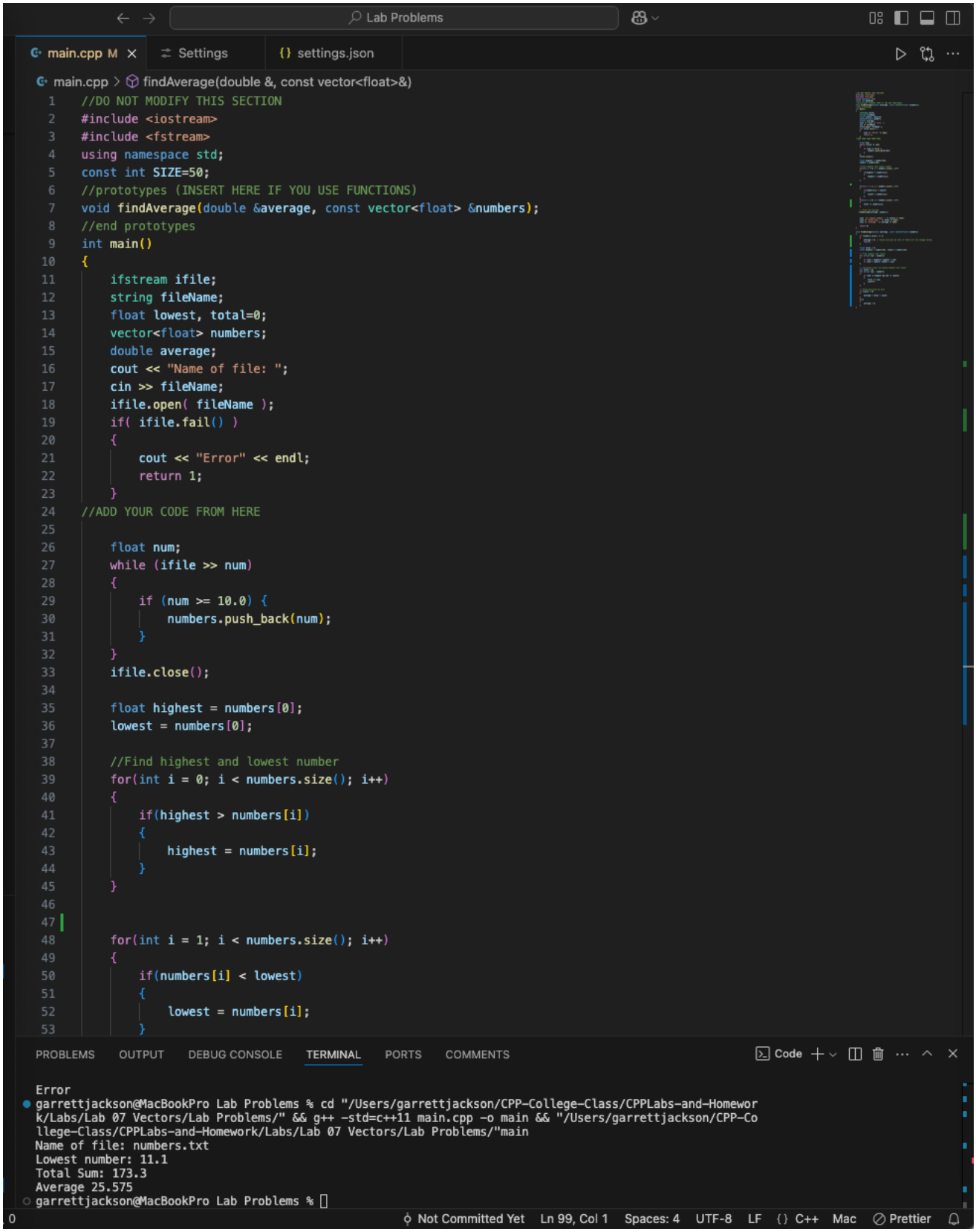
Write a program that asks the user for a file name. Assume the file contains a series of float numbers, each written on a separate line (maximum 20). The program should read the contents of the file into a vector, only the numbers greater or equal than 10.0 and then display the following data:

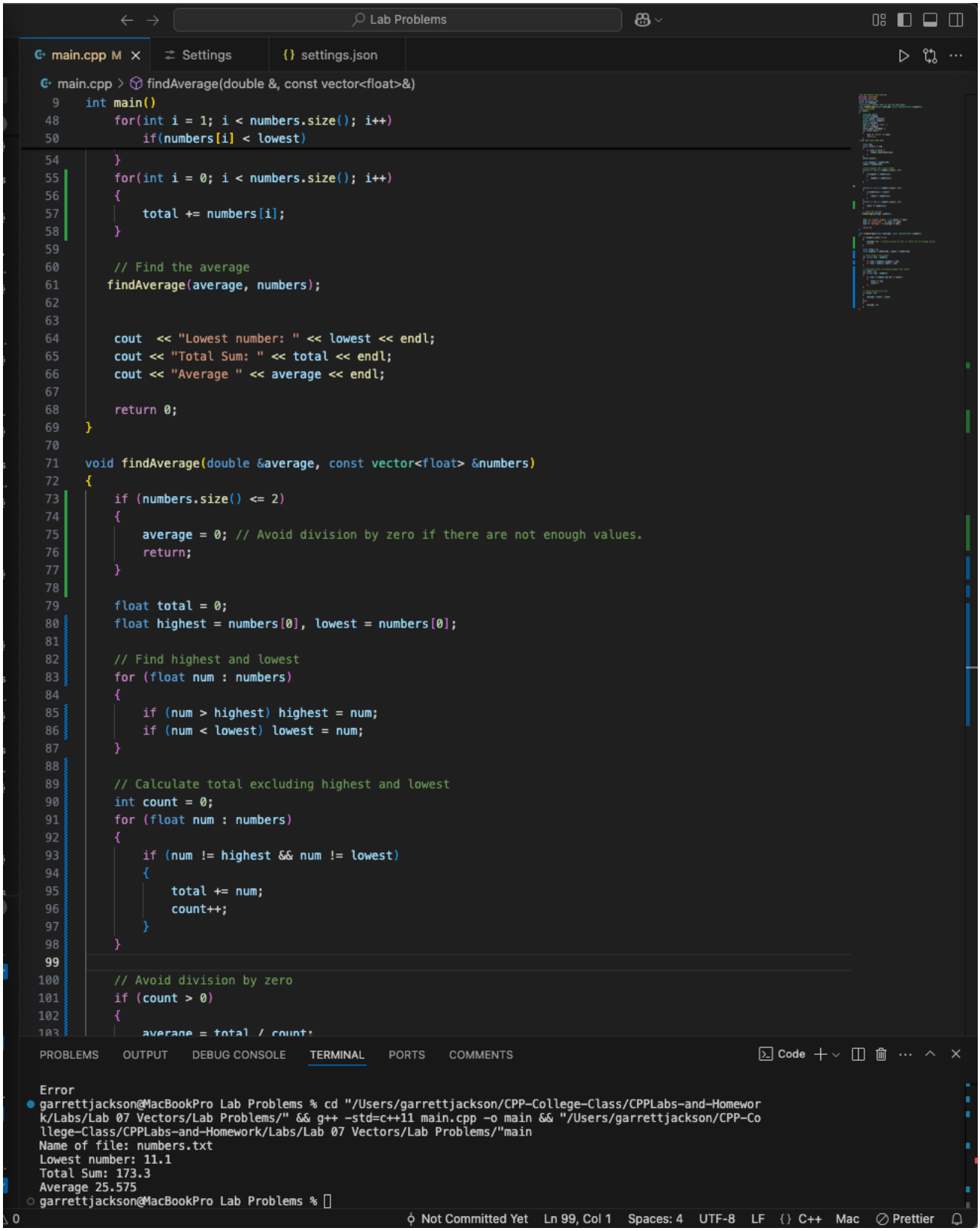
- The lowest number in the vector
- The total of the numbers in the vector
- The average of the numbers in the array excluding the greatest and the lowest.

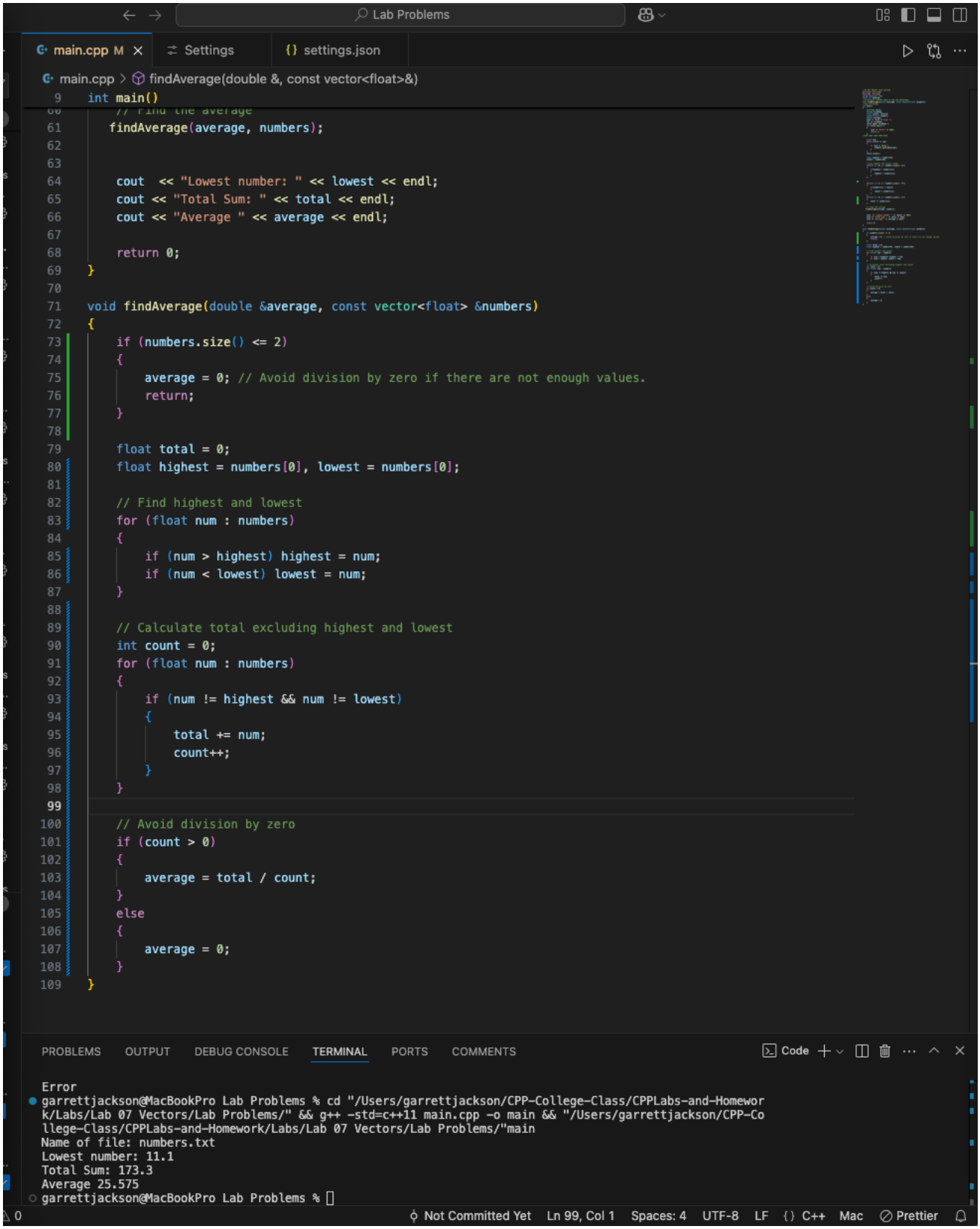
TEMPLATE:

```
//DO NOT MODIFY THIS SECTION
#include <iostream>
#include <fstream>
using namespace std;
const int SIZE=50;
//prototypes (INSERT HERE IF YOU USE FUNCTIONS)

//end prototypes
int main()
{
    ifstream ifile;
    string fileName;
    float lowest, total=0;

    vector<float> numbers;
    double average;
    cout << "Name of file: ";
    cin >> fileName;
    ifile.open( fileName );
```

```
    if( ifile.fail() )
    {
        cout << "Error" << endl;
        return 1;
    }
//ADD YOUR CODE FROM HERE
```


Lab Problems
main.cpp M
Settings
settings.json
main.cpp > findAverage(double &, const vector<float>&)
//DO NOT MODIFY THIS SECTION
#include <iostream>
#include <fstream>
using namespace std;
const int SIZE=50;
//prototypes (INSERT HERE IF YOU USE FUNCTIONS)
void findAverage(double &average, const vector<float> &numbers);
//end prototypes
int main()
{
ifstream ifile;
string fileName;
float lowest, total=0;
vector<float> numbers;
double average;
cout << "Name of file: ";
cin >> fileName;
ifile.open(fileName);
if(ifile.fail())
{
cout << "Error" << endl;
return 1;
}
//ADD YOUR CODE FROM HERE
float num;
while (ifile >> num)
{
if (num >= 10.0) {
numbers.push_back(num);
}
}
ifile.close();
float highest = numbers[0];
lowest = numbers[0];
//Find highest and lowest number
for(int i = 0; i < numbers.size(); i++)
{
if(highest > numbers[i])
{
highest = numbers[i];
}
}
for(int i = 1; i < numbers.size(); i++)
{
if(numbers[i] < lowest)
{
lowest = numbers[i];
}
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
COMMENTS
Code
Error
garrettjackson@MacBookPro Lab Problems % cd "/Users/garrettjackson/CPP-College-Class/CPPLabs-and-Homework/Labs/Lab 07 Vectors/Lab Problems/" && g++ -std=c++11 main.cpp -o main && "/Users/garrettjackson/CPP-College-Class/CPPLabs-and-Homework/Labs/Lab 07 Vectors/Lab Problems/"main
Name of file: numbers.txt
Lowest number: 11.1
Total Sum: 173.3
Average 25.575
garrettjackson@MacBookPro Lab Problems %
0
Not Committed Yet
Ln 99, Col 1
Spaces: 4
UTF-8
LF
C++
Mac
Prettier


Lab Problems
main.cpp M
Settings
settings.json
main.cpp > findAverage(double &, const vector<float>&)
int main()
for(int i = 1; i < numbers.size(); i++)
if(numbers[i] < lowest)
}
for(int i = 0; i < numbers.size(); i++)
{
total += numbers[i];
}
// Find the average
findAverage(average, numbers);
cout << "Lowest number: " << lowest << endl;
cout << "Total Sum: " << total << endl;
cout << "Average " << average << endl;
return 0;
}
void findAverage(double &average, const vector<float> &numbers)
{
if (numbers.size() <= 2)
{
average = 0; // Avoid division by zero if there are not enough values.
return;
}
float total = 0;
float highest = numbers[0], lowest = numbers[0];
// Find highest and lowest
for (float num : numbers)
{
if (num > highest) highest = num;
if (num < lowest) lowest = num;
}
// Calculate total excluding highest and lowest
int count = 0;
for (float num : numbers)
{
if (num != highest && num != lowest)
{
total += num;
count++;
}
}
// Avoid division by zero
if (count > 0)
{
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
Code
Error
garrettjackson@MacBookPro Lab Problems % cd "/Users/garrettjackson/CPP-College-Class/CPPLabs-and-Homework/Labs/Lab 07 Vectors/Lab Problems/" && g++ -std=c++11 main.cpp -o main && "/Users/garrettjackson/CPP-College-Class/CPPLabs-and-Homework/Labs/Lab 07 Vectors/Lab Problems/"main
Name of file: numbers.txt
Lowest number: 11.1
Total Sum: 173.3
Average 25.575
garrettjackson@MacBookPro Lab Problems %
Not Committed Yet Ln 99, Col 1 Spaces: 4 UTF-8 LF C++ Mac Prettier




```
//DO NOT MODIFY THIS SECTION

#include <iostream>
```

```
#include <fstream>
using namespace std;
const int SIZE=50;
//prototypes (INSERT HERE IF YOU USE FUNCTIONS)
void findAverage(double &average, const vector<float> &numbers);
//end prototypes
int main()
{
    ifstream ifile;
    string fileName;
    float lowest, total=0;
    vector<float> numbers;
    double average;
    cout << "Name of file: ";
    cin >> fileName;
    ifile.open( fileName );
    if( ifile.fail() )
    {
        cout << "Error" << endl;
        return 1;
    }
//ADD YOUR CODE FROM HERE

    float num;
    while (ifile >> num)
    {
        if (num >= 10.0) {
            numbers.push_back(num);
        }
```

```
    }
    ifile.close();

    float highest = numbers[0];
    lowest = numbers[0];

    //Find highest and lowest number
    for(int i = 0; i < numbers.size(); i++)
    {
        if(highest > numbers[i])
        {
            highest = numbers[i];
        }
    }


    for(int i = 1; i < numbers.size(); i++)
    {
        if(numbers[i] < lowest)
        {
            lowest = numbers[i];
        }
    }
    for(int i = 0; i < numbers.size(); i++)
    {
        total += numbers[i];
    }

    // Find the average
```

```
   findAverage(average, numbers);



    cout  << "Lowest number: " << lowest << endl;
    cout << "Total Sum: " << total << endl;
    cout << "Average " << average << endl;

    return 0;
}

void findAverage(double &average, const vector<float> &numbers)
{
    if (numbers.size() <= 2)
    {
        average = 0; // Avoid division by zero if there are not enough values.
        return;
    }

    float total = 0;
    float highest = numbers[0], lowest = numbers[0];

    // Find highest and lowest
    for (float num : numbers)
    {
        if (num > highest) highest = num;
        if (num < lowest) lowest = num;
    }

    // Calculate total excluding highest and lowest
```

```
    int count = 0;
    for (float num : numbers)
    {
        if (num != highest && num != lowest)
        {
            total += num;
            count++;
        }
    }

    // Avoid division by zero
    if (count > 0)
    {
        average = total / count;
    }
    else
    {
        average = 0;
    }
}
```