1. What I was wondering was if we needed to right our own algorithms for this or if we should use existing C++ libraries.

   Answer: Please do not use any library to implement your sorting function in this assignment.

2. I'm not very familiar with the C++ but it looks like in the algorithms library there is a min heap class you can implement. So for example would it be a viable solution to use the min heap to combine and sort the numbers for us or should we write our own?

   Answer: The algorithm is very simple because the two arrays are sorted ones. You do **NOT** need the min heap class to implement your sorting function.

3. What kind of input can I expect? Should I program for a specific input format, a list with one number on each line in each file, no blank lines. Or can I not assume this much, should I try to program for several possible input formats, i.e.. separated by spaces, comma separated (CSV), etc.

   Answer: Please use two simple text files, each of which has a list with one number on each line. No blank lines please.

4. Here is the error message,
   ***0003_hw4.cpp: In function ‘int main()’:
   ***0003_hw4.cpp:44:15: error: no matching function for call to ‘std::basic_ifstream<char>::open(std::string&)’

   It says something is wrong on line 44.
   ===

```
  30 int main()
  31 {
  32
  33 int iArray1[MAX_SIZE];
  34 int iArray1_size = 0;
  35 int iArray2[MAX_SIZE];
  36 int iArray2_size = 0;
  37 string developer("Brian Ponder");
  38 string inFile;
  39 ifstream in;
  40
  41 cout << "***Welcome to " << developer << "\'s sorting program ***"
  42         << "Enter the first input file name: ";
  43    cin  >> inFile;
  44 in.open(inFile);
```

```
45 while(in >> iArray1[i]){
46 iArray1_size++;
47 i++;
48
49 }
```

===

Answer: The line 44 is "in.open(inFile);"

Below is the definition of ifstream::open
public member function
void open (const char * filename, ios_base::openmode mode = ios_base::in);

As we can see here, the first parameter is a pointer of char* type.
but you passed in the inFile of string type.

you need a type convention.

please refer to string::c_str(...);