

```

/* As a demo, Dr. Li used various strange data structures. Hopefully it enriches and
*expand your knowledge. But you don't have to follow Dr Li's strange defs to finish this
*project
* Xuechao Li
* Project3 practice
*/

#include <fstream>
#include <iostream>
#include <vector>
using namespace std;

/* declare your user-defined functions */
bool check_file(string);

/* do not forget to describe each function */
vector<int> read_file(string);

void write_file(string, vector<int>);

vector<int> merge(vector<int>, vector<int>);

/*
* Display the values of a given vector.
*
* Param: file Name of file to display. (string)
* Param: v Vector containing values to display. (vector<int>)
*/
void to_string(string, vector<int>);

/*
* Merge the numbers in two specified files and write all the numbers
* to a specified third file sorted in ascending order.
*
* Return: 1 Program completed successfully. (int)
*/
int main() {
    /* declare your variables */
    .....

    /* Get name of file one. */
    do {
        /* user friendly interfaces */
    } while (cin.fail() || !check_file(file1));
}

```

```

/* Get and display numbers from file one. */
numbers1 = read_file(file1);
.....

/* Get name of file two. */
do {
    /* user friendly interfaces */
} while (cin.fail() || !check_file(file2));

/* Get and display numbers from file two. */
numbers2 = read_file(file2);

/* Combine vectors and display the sorted result. */
numbers3 = merge(numbers1, numbers2);

/* Get name of output file. */
do {
    /* user friendly interfaces */
} while (cin.fail());

/* Write combined vector to output file. */
write_file(file3, numbers3);

return.....;
}

bool check_file(string file) {
    /* Input file stream. (ifstream) */
    ifstream stream;

    /* Check whether file exists. */
    stream.open(file.c_str());
    if (stream.fail()) {
        .....
    }
    stream.close();

    return true;
}

vector<int> read_file(string file) {
    /* Input file stream. (ifstream) */
    ifstream stream;

    /* Vector containing numbers from file. (vector<int>) */
    vector<int> v;

    /* Integer read from file. (int) */

```

```

    int i;

    /* Add each number in the file to a vector. */
    stream.open(.....);
    while (stream.good()) {
        .....
    }

    return .....;
}

void write_file(string file, vector<int> v) {
    /* Output file stream. (ofstream) */
    .....
}

vector<int> merge(vector<int> v1, vector<int> v2) {
    variables .....

    /* Compare both vectors. */
    while (.....) {
        if (v1 value > v2 value) {
            .....
        } else {
            .....
        }
    }

    /* Add any remaining numbers from vector one. */
    if (still have values in v1) {
        copy/paste
    }

    /* Add any remaining numbers from vector two. */
    if (still have values in v1) {
        copy/paste
    }

    return .....;
}

void to_string(string file, vector<int> v) {
    /* Vector iterator number. (unsigned short) */
    unsigned short i;

    /* Display the numbers contained in a vector. */
}

```