```
/*
 * Student.cpp
 *
 *  Created on: Sep 26, 2016
 *      Author: Kenny Do
 */

#include <iomanip>
#include <iostream>
#include <cstdlib>
#include <time.h>
#include <fstream>
#include <string.h>

#include "myDate.h"
#include "MergeSort.h"
#include "StudentComparators.h"
#include "StudentStruct.h"

using namespace std;

/**
 * Returns a random name from names.txt
 */
char* readName() {
    fstream infile("names.txt");
    char cName[256];
    int random = rand() % 100 + 1;
    int lineNumber = 0;
    while(infile.getline(cName, 256)) {
        ++lineNumber;
        if(lineNumber == random) {
            break;
        }
    }
    infile.close();
    return strdup(cName);
}

/**
 * Menu for the selecting which sort to be displayed
 */
int menu() {
    cout << "1) Display original list" <<endl;
    cout << "2) Display list sorted by Student ID" <<endl;
    cout << "3) Display list sorted by Student Age" <<endl;
    cout << "4) Display list sorted by Student Name" <<endl;
    cout << "5) Display list sorted by Student Grade" <<endl;
    cout << "6) Exit" <<endl;
    char choice;
    choice = cin.get();
    cin.ignore(256, '\n');
    choice -= 48;
    while((int) choice < 1 || (int) choice > 6 ) {
        //Ï = 139 in char
        if(choice = 139) {
            //Terminates cleanly
            return 6;
        }
        cout << "Please pick a valid choice." <<endl;
        choice = cin.get();
        choice -= 48;
        cin.ignore(256, '\n');
    }
    return choice;
}
```

```
/**
 * Print the student
 */
void printStudent(Student & student) {
    cout << setw(20) << left << student.idNum <<
            setw(20) << student.name <<
            setw(20) << student.birthdate.toString() <<
            setw(20) << student.grade << '\n';
}

/**
 * Print an array of students
 */
void printArrayOfStudents(Student student[], const int & size) {
    cout << setw(20) << left << "Student ID" <<
            setw(20) << "Name" <<
            setw(20) <<  "Birthday" <<
            setw(20) << "Grade" << '\n';
    for(int i = 0; i < size;i++) {
        printStudent(student[i]);
    }
}

/**
 * Generate a list of students up to the size
 */
void generateStudents(Student student[], int size) {
    for(int i = 0; i < size; i++) {
        student[i] = {rand() % 8999 + 1000, readName(),
        myDate::getRandomDayBetween(myDate(1,1,1990), myDate(12,31,1994)), rand() % 50 + 50};
    }
}

int main() {
    srand(time(NULL));
    const int size = 10;

    Student student[size];
    generateStudents(student, size);

    Student stuID[size];
    Student stuAge[size];
    Student stuName[size];
    Student stuGrade[size];
    copy(student, student + size, stuID);
    copy(student, student + size, stuAge);
    copy(student, student + size, stuName);
    copy(student, student + size, stuGrade);

    Student * stupt[4];
    stupt[0] = stuID;
    stupt[1] = stuAge;
    stupt[2] = stuName;
    stupt[3] = stuGrade;
    MergeSort::mergeSort(stupt[0], size,StudentComparators::compareByStudentID());
    MergeSort::mergeSort(stupt[1], size,StudentComparators::compareByStudentAge());
    MergeSort::mergeSort(stupt[2], size,StudentComparators::compareByStudentName());
    MergeSort::mergeSort(stupt[3], size,StudentComparators::compareByStudentGrade());
    int choice = menu();
    while(choice != 6) {
        switch(choice) {
        case 1:
            printArrayOfStudents(student, size);
            break;
        case 2:
            printArrayOfStudents(stupt[0], size);
            break;
        case 3:
            printArrayOfStudents(stupt[1], size);
            break;
        case 4:
            printArrayOfStudents(stupt[2], size);
            break;
        case 5:
            printArrayOfStudents(stupt[3], size);
            break;
        }
        cout << endl;
        choice = menu();
    }
    return 0;
}
```