```
//A high school admission system in C++


#include <iostream>
#include <vector>
#include <map>
#include <string>
using namespace std;

class Student {
private:
    static int nextAdmissionNumber;
    int admissionNumber;
    string name;
    int age;
    string address;
    string stream;
    string dormitory;

public:
    Student(string n, int a, string addr)
        : name(n), age(a), address(addr) {
        admissionNumber = nextAdmissionNumber++;
    }

    int getAdmissionNumber() const {
        return admissionNumber;
    }
    string getName() const {
        return name;
    }

    void setStream(string strm) {
        stream = strm;
    }

    string getStream() const {
        return stream;
    }

    void setDormitory(string dorm) {
        dormitory = dorm;
    }

    string getDormitory() const {
        return dormitory;
    }

    void display() const {
        cout << "Admission Number: " << admissionNumber << endl;
        cout << "Name: " << name << endl;
        cout << "Age: " << age << endl;
        cout << "Address: " << address << endl;
        cout << "Stream: " << stream << endl;
        cout << "Dormitory: " << dormitory << endl;
    }
};
```

```cpp
int Student::nextAdmissionNumber = 1;

class Class {
private:
    string stream;
    vector<Student> students;

public:
    Class() = default; // Default constructor

    Class(string strm) : stream(strm) {}

    void addStudent(const Student &student) {
        students.push_back(student);
    }

    void displayStudents() const {
        cout << "Class: " << stream << endl;
        for (const auto &student : students) {
            student.display();
            cout << "---------------" << endl;
        }
    }
};

class Dormitory {
private:
    string name;
    vector<Student> students;

public:
    Dormitory() = default; // Default constructor

    Dormitory(string n) : name(n) {}

    void addStudent(const Student &student) {
        students.push_back(student);
    }

    void displayStudents() const {
        cout << "Dormitory: " << name << endl;
        for (const auto &student : students) {
            student.display();
            cout << "---------------" << endl;
        }
    }
};

class Administration {
private:
    vector<Student> students;
    map<string, Class> classes;
    map<string, Dormitory> dormitories;

public:
    void admitStudent(const string &name, int age, const string &address) {
        Student student(name, age, address);
        students.push_back(student);
    }

    void assignClass(int admissionNumber, const string &stream) {
        for (auto &student : students) {
            if (student.getAdmissionNumber() == admissionNumber) {
                student.setStream(stream);
                if (classes.find(stream) == classes.end()) {
                    classes[stream] = Class(stream);
                }
                classes[stream].addStudent(student);
                return;
            }
        }
    }

    void assignDormitory(int admissionNumber, const string &dormitory) {
        for (auto &student : students) {
            if (student.getAdmissionNumber() == admissionNumber) {
                student.setDormitory(dormitory);
                if (dormitories.find(dormitory) == dormitories.end()) {
                    dormitories[dormitory] = Dormitory(dormitory);
                }
                dormitories[dormitory].addStudent(student);
                return;
            }
        }
    }
```

```
    void displayAllStudents() const {
        for (const auto &student : students) {
            student.display();
            cout << "---------------" << endl;
        }
    }

    void displayClass(const string &stream) const {
        if (classes.find(stream) != classes.end()) {
            classes.at(stream).displayStudents();
        } else {
            cout << "No students in this class." << endl;
        }
    }

    void displayDormitory(const string &dormitory) const {
        if (dormitories.find(dormitory) != dormitories.end()) {
            dormitories.at(dormitory).displayStudents();
        } else {
            cout << "No students in this dormitory." << endl;
        }
    }
};

int main() {
    Administration admin;

    // Admit students
    admin.admitStudent("PETER Komen", 15, "123 Maple St");
    admin.admitStudent("Bob Mukhwana", 16, "456 Oak St");
    admin.admitStudent("Charlie Wainaina", 15, "789 Pine St");

    // Assign classes
    admin.assignClass(1, "EAST");
    admin.assignClass(2, "WEST");
    admin.assignClass(3, "CENTRAL");

    // Assign dormitories
    admin.assignDormitory(1, "MT KENYA  HOUSE");
    admin.assignDormitory(2, "MT ELGON HOUSE");
    admin.assignDormitory(3, "ATLANTIC HOUSE");

    // Display all students
    cout << "All Students:" << endl;
    admin.displayAllStudents();

    // Display specific class
    cout << "\n STUDENT WITH DISABILITIES:" << endl;
    admin.displayClass("NO STUDENT WITH DISABILITY");

    // Display specific dormitory
    cout << "\n NEW DORM:" << endl;
    admin.displayDormitory("FOR THOSE WITH SPECIAL NEEDS");

    return 0;
}
```