Lab 13: University Course Management System

Description:
You are tasked with creating a C++ program to manage courses and students in a university using STL set and map containers. The program should allow the user to add courses, enroll students, display course details, save data to files, and load data from files. You will need to create a Person base class and derived classes Student and Instructor. The program should also demonstrate inheritance and polymorphism.

Tasks:

1. **Create a Person base class:**
    - The Person class should have the following private attributes:
        - name (string): The name of the person.
        - id (int): The ID of the person.
    - The class should have a constructor to initialize these attributes.
    - The class should have public methods to:
        - Get the name and ID of the person.
        - Display the person's details.
2. **Create derived classes Student and Instructor:**
    - The Student class should inherit from Person and add the following private attribute:
        - coursesEnrolled (set<string>): A set of course codes the student is enrolled in.
    - The Instructor class should inherit from Person and add the following private attribute:
        - coursesTeaching (set<string>): A set of course codes the instructor is teaching.
    - Both classes should have methods to:
        - Add a course to the respective set.
        - Display the person's details along with their courses.

3. **Manage courses and enrollments:**
    - Use an STL map to store courses. The key should be the course code (string), and the value should be the course name (string).
    - Create a function *addCourse* that allows the user to add a new course to the map.
    - Create a function *enrollStudent* that allows the user to enroll a student in a course. The function should add the course code to the student's coursesEnrolled set.
    - Create a function *assignInstructor* that allows the user to assign an instructor to a course. The function should add the course code to the instructor's coursesTeaching set.
4. **Save and load data to/from files:**
    - Create a function *saveData* that writes the details of courses, students, and instructors to separate files (courses.txt, students.txt, instructors.txt).
    - Create a function *loadData* that reads the details from the files and populates the map and sets.

Requirements:

- Use file streams (*ofstream* and *ifstream*) to handle file operations.
- Use the STL set and map containers to manage courses and enrollments.
- Ensure proper error handling for file operations (e.g., check if the file opens successfully).
- Use inheritance and polymorphism to design the Person, Student, and Instructor classes.

Sample Input/Output

1. Add Course
2. Enroll Student
3. Assign Instructor
4. Display Data
5. Save Data
6. Load Data
7. Exit
Enter your choice: 1
Enter course code: CS101
Enter course name: Introduction to Computer Science
Course added successfully!

1. Add Course
2. Enroll Student
3. Assign Instructor
4. Display Data
5. Save Data
6. Load Data
7. Exit
Enter your choice: 2
Enter student ID: 1
Enter course code: CS101
Enter student name: John Doe
Student enrolled successfully!

1. Add Course
2. Enroll Student
3. Assign Instructor
4. Display Data
5. Save Data
6. Load Data
7. Exit
Enter your choice: 3
Enter instructor ID: 101
Enter course code: CS101
Enter instructor name: Dr. Smith

Instructor assigned successfully!

1. Add Course
2. Enroll Student
3. Assign Instructor
4. Display Data
5. Save Data
6. Load Data
7. Exit
Enter your choice: 4
Courses:
Code: CS101, Name: Introduction to Computer Science
Students:
Name: John Doe, ID: 1
Courses Enrolled: CS101
Instructors:
Name: Dr. Smith, ID: 101
Courses Teaching: CS101

1. Add Course
2. Enroll Student
3. Assign Instructor
4. Display Data
5. Save Data
6. Load Data
7. Exit
Enter your choice: 5
Data saved to files!

1. Add Course
2. Enroll Student
3. Assign Instructor
4. Display Data
5. Save Data
6. Load Data
7. Exit
Enter your choice: 7
Exiting program...