Technical Description: Incident Report System

I. Task Description

Student 1 - Administrator Role

- Add, delete, and modify incidents (location, description, severity)

- View and export all reported incidents

- Mark incidents as resolved

Student 2 - Reporter Role

- Create new incident reports

- View list of unresolved incidents

- Filter incidents by severity or location

II. Data Structures Used by the Team

The following classes will be implemented:

- Incident:

  - string id

  - string description

  - string location

  - int severity (scale from 1 to 5)

  - bool resolved

- Date:

  - int day

  - int month

  - int year

- Report:

  - Incident[] incidents

  - Date reportDate

Relationships:

- The Report class has a composition relationship with Incident

- The Incident class uses Date to track report creation time

III. File Structure

incidents.txt

Stores all incident reports:

<number_of_incidents>

<incident_id> <description> <location> <severity> <resolved_flag>

...

unresolved.txt

Contains only unresolved incident entries:

<incident_id> <description> <location> <severity>

...

report_log.txt

Contains report snapshots:

<report_date>

<incident_id_list_for_that_day>

...

IV. Interacting with Executables

Application 1 (Administrator):

./admin.exe view_incidents

- View all recorded incidents

./admin.exe add_incident <id> <description> <location> <severity>

- Add a new incident

./admin.exe delete_incident <id>

- Delete a specific incident

./admin.exe modify_incident <id> <field> <new_value>

- Modify the description/location/severity of an incident

./admin.exe resolve_incident <id>

- Mark an incident as resolved

./admin.exe export_reports

- Export report log to a file

Application 2 (Reporter):

./reporter.exe report_incident <id> <description> <location> <severity>

- Report a new incident

./reporter.exe view_unresolved

- View all unresolved incidents

./reporter.exe filter_incidents <by_location | by_severity> <value>

- Filter incidents accordingly