

## Kern's CS161 Database Project

### **Background**

This is a program for processing and displaying reports about fires in Oregon. You can tell it to display single fires, a subset of fires, aggregate data about certain fires, or every fire in the dataset.

### **1. The FireReport struct**

Each fire incident report is stored somewhere within an array of structs. The struct is called FireReport, and it contains the following information:

<b>Variable Name</b>	<b>Data Type</b>	<b>Description</b>	<b>Example</b>
fireIdentifier	string	Fire unique ID	19-974047-19
fireName	string	Colloquial name	Eden Bench Raws
fireSizeInAcres	double	Area of fire	14.5
fireDateAndTime	string	Date/time of fire	06/01/2019 01:34:00 PM
fireGeneralCause	string	Reported cause	Lightning
fireLandType	string	Category of land	State

## 2. Reading Data

The program's *main* begins with a function called *GetFireReportsFromFile*, which first looks for a database file called "data.csv" (customizable by changing `DATA_FILENAME` in `main.cpp`). If no problems are encountered when trying to open the file, *GetFireReportsFromFile* will then call the function *CreateFireReportFromString* on every single line of the file. This will use `getline()` with a comma delimiter to separate the data into different strings. These strings are then used to populate a `FireReport` instance within the array.

Next, if the file is read without problems, *main* will run a `While` loop which continuously takes user inputs so that different types of reports can be chosen:

```
Which type of reports would you like to view?  
You can type 'All', 'Single', 'Extrema', 'Aggregate', or 'Quit'.
```

This works by accepting a string from `cin` and then examining it for known commands. The strings are converted to lowercase so that the user doesn't have to worry about case sensitivity. Some commands are detected using a direct `'=='` comparison, and some are done by using `'string.find() != string::npos'`, depending on how unambiguous the command needs to be.

If an unknown command is entered, the program will start over at the beginning. If the user types "quit" or "exit" at the main prompt, the program will stop.

## 2.5. Printing All Records

If the user selects "all", every valid record will be displayed (up to a maximum, hardcoded as a constant integer):

Fire Name	Fire ID	Date/Time	Fire Size	Land Type	Cause
Honeycutt Dr 380	19-712A41-19	05/03/2019 04:37:00 PM	0.25 Acres	Private	Debris Burning
Left Fork Fielder Crk Rd	19-712A32-19	04/24/2019 08:17:00 AM	0.01 Acres	Private	Recreation
Eagles Rest	19-771164-19	05/11/2019 01:05:00 PM	0.01 Acres	Not Specified	Debris Burning
Whiskey Creek	19-954049-19	05/03/2019 03:25:00 PM	0.73 Acres	Private	Arson
Crooked Ag	19-582075-19	05/02/2019 03:43:00 PM	0.73 Acres	Private	Debris Burning
Limber Fire	19-581111-19	04/30/2019 04:00:00 AM	0.02 Acres	Private	Equipment Use
High Heaven Fire	19-531086-19	04/30/2019 01:35:00 PM	0.02 Acres	Private	Debris Burning
Dog Creek 132	19-982059-19	05/28/2019 06:14:00 PM	0.02 Acres	Private	Lightning
Susan Cr. Rd.	19-733208-19	05/04/2019 03:50:00 PM	18.27 Acres	Private	Debris Burning
Raichl	19-521035-19	03/18/2019 01:00:00 PM	0.50 Acres	Private	Debris Burning
Alder Creek	19-721139-19	04/24/2019 02:46:00 PM	0.01 Acres	Private	Recreation
Speaker Road	19-712A28-19	04/17/2019 11:00:00 AM	0.01 Acres	Private	Smoking
Sandy Creek	19-722151-19	05/03/2019 02:24:00 PM	0.06 Acres	Private	Debris Burning
85100 Territorial	19-781095-19	05/07/2019 03:38:00 PM	0.03 Acres	State	Debris Burning
Little Applegate 4001	19-711540-19	05/24/2019 07:14:00 PM	1.50 Acres	Private	Debris Burning
Fortune Branch	19-732214-19	05/13/2019 07:30:00 AM	0.01 Acres	BLM	Debris Burning
Jacksonville Reservoir R	19-711516-19	03/11/2019 03:57:00 PM	2.00 Acres	Private	Debris Burning

### 3. Printing A Single Record

If the user selects "single", they can choose to select by name or ID:

```
Which criterion would you like to select by?  
You can type 'Name', 'ID', or 'Back'.
```

```
Searching for a fire with name "lightning mcqueen":  
Fire Name | Fire ID | Date/Time | Fire Size | Land Type | Cause  
-----  
Lightning McQueen | 19-582072-19 | 03/28/2019 05:00:00 PM | 1.50 Acres | Private | Lightning
```

```
Searching for a fire with ID "19-711519-19":  
Fire Name | Fire ID | Date/Time | Fire Size | Land Type | Cause  
-----  
Medco B | 19-711519-19 | 05/04/2019 11:05:00 PM | 345.00 Acres | Private | Under Invest
```

The 'single' function works similarly in either case. All records in the struct array are examined using a For loop, and the first one with a matching name or ID is printed.

#### 4. Printing Matching Records

If the user selects "matching", they can choose to select by cause or by date:

```
Which criterion would you like to select by?
You can type 'Cause', 'Date', or 'Back'.
```

In either case, the logic is the same. After outputting the column headers, the program will loop through the array of FireReports. The input may be case-sensitive due to time constraints. Any records which match the given criterion will be listed:

```
Here are the fires from "05/03/2019":
Fire Name      | Fire ID      | Date/Time      | Fire Size    | Land Type    | Cause
-----
Honeycutt Dr 380 | 19-712A41-19 | 05/03/2019 04:37:00 PM | 0.25 Acres | Private      | Debris Burning
Whiskey Creek   | 19-954049-19 | 05/03/2019 03:25:00 PM | 0.73 Acres | Private      | Arson
Sandy Creek     | 19-722151-19 | 05/03/2019 02:24:00 PM | 0.06 Acres | Private      | Debris Burning
Fourth Creek    | 19-721150-19 | 05/03/2019 11:57:00 AM | 0.10 Acres | Private      | Debris Burning
La Duke STAT    | 19-771152-19 | 05/03/2019 01:40:00 PM | 0.20 Acres | Private      | Equipment Use
COIDC 0100      | 19-955098-19 | 05/03/2019 02:43:00 PM | 1.50 Acres | Private      | Miscellaneous
Mill Creek      | 19-954050-19 | 05/03/2019 06:11:00 PM | 0.01 Acres | Private      | Debris Burning
```

## 5. Printing Min/Max Records

If the user selects "extrema", they can choose to view the largest and smallest sized fires by a given cause, or a given type of land:

```
Which criterion would you like to select by?  
You can type 'Cause', 'Land', or 'Back'.
```

The input may be case-sensitive due to time constraints. The function will then create two empty FireReports and two doubles: one of each for the minimum and maximum fire incidents. While looping through the main FireReport array, it will check to see if any relevant record is larger than the current maximum or smaller than the current minimum. If so, that record is stored in the appropriate variables. The records are then printed:

```
Here are the largest and smallest fires caused by "Lightning":  
Fire Name      | Fire ID      | Date/Time      | Fire Size      | Land Type      | Cause  
-----  
Dog Creek 132   | 19-982059-19 | 05/28/2019 06:14:00 PM | 0.02 Acres | Private      | Lightning  
Eden Bench Raws | 19-974047-19 | 06/01/2019 01:34:00 PM | 14.50 Acres | State       | Lightning  
  
Here are the largest and smallest fires which occurred on land type "County":  
Fire Name      | Fire ID      | Date/Time      | Fire Size      | Land Type      | Cause  
-----  
Post Canyon     | 19-954051-19 | 05/10/2019 04:30:00 PM | 0.01 Acres | County       | Recreation  
Upper Hog Creek | 19-712A24-19 | 03/20/2019 04:13:00 PM | 2.00 Acres | County       | Recreation
```

## 6. Printing Aggregate Data

If the user types "aggregate" as a command, no specific records will actually be returned. The user will choose to search by cause type or by land type, and then the average fire size for that criterion will be displayed:

```
Which criterion would you like to average by?  
You can type 'Cause', 'Land', or 'Back'.
```

```
Display the average fire size by which type of land? (Case sensitive)  
State  
On average, fires on "State" land spread to a size of 1.40364 acres.
```

```
Display the average fire size by which cause? (Case sensitive)  
Smoking  
On average, fires caused by "Smoking" spread to a size of 0.5075 acres.
```