

COMP242-Project II

In this project, you will implement an equation file browser application using linked stack and cursor array implementations of a linked list.

The equation file browser interface will look like the following:

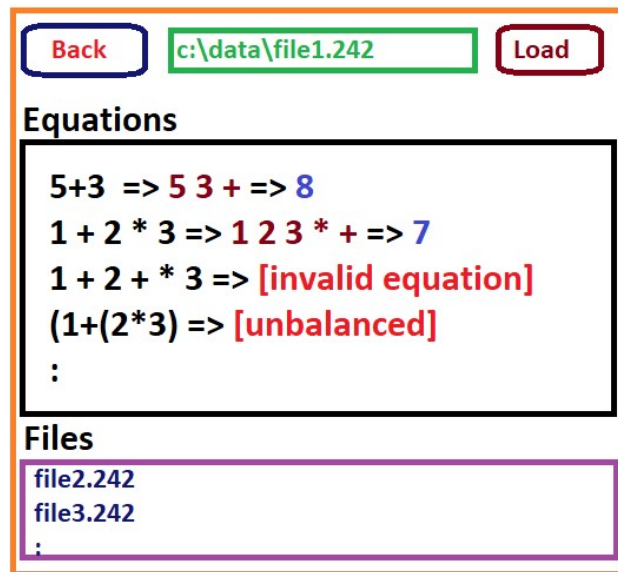


Fig 1: Main file browser GUI

The equation file is a special text file that has a **.242** extension. The equations in this file are in infix format. A typical equation file looks like the following:

```

file1.242 - Notepad
File Edit Format View Help
<242>
  <files>
    <file> c:\data\file2.242 </file>
    <file> c:\data\file3.242 </file>
  </files>

  <equations>
    <equation>5 + 3</equation>
    <equation>1 + 2 * 3</equation>
    <equation>1 + 2 + * 3</equation>
    <equation>( 1 + ( 2 * )</equation>
  </equations>
</242>
Ln 1, Col 1    100%  Windows (CRLF)  UTF-8
  
```

The file has to start with the tag **<242>** and ends with the tag **</242>**. Within the start and end tags there are 2 optional sections:

- **Files section:** it starts with **<files>** tag and ends with **</files>** tag.
- **Equations section:** it starts with **<equations>** tag and ends with **</equations>** tag.

And so what inside the files and equations sections (e.g. **<equation>** **<file>**).

- At start, the user has to click the **Load** button. The load button will open a **file chooser** to select an equation file. The selected file will be displayed in the label next to the load button (e.g. `c:\users\file1.242`) and then the file contents will be loaded (as shown in **Fig 1**) **if the file is valid** (e.g. tags are balanced) as follow:
 - In the equations section you need to load the infix equations -if any-; make sure they are valid and balanced (using stack). If valid, then convert them to postfix equations (using stack), and finally you need to evaluate the postfix equation and print the results (also using stack).
 - In the files section, you need to load the files -if any-.
 - If the file is in valid, show a meaningful error message.

(Note: for more about file chooser, check this out <https://jenkov.com/tutorials/javafx/filechooser.html>).

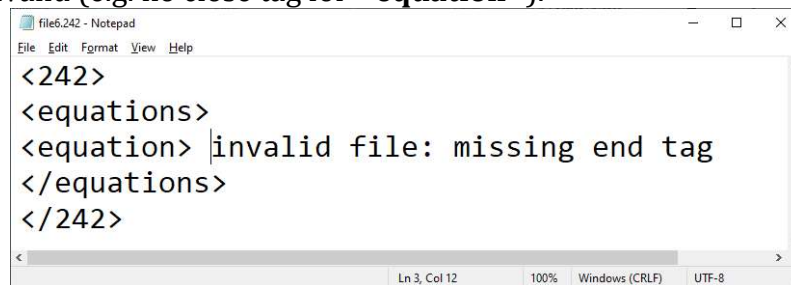
- Valid file:** you need to check and indicate whether the file tags are balanced or not. Meaning that each **start tag** has an **end tag** and they are closed in order. The tags are:

```

➤ <242>           </242>
➤ <equations>      </equations>
➤ <files>          </files>
➤ <equation>       </equation>
➤ <file>           </file>

```

This can be done using a stack as well similar to the balanced delimiters in equations. For example the following file is invalid (e.g. no close tag for **<equation>**).



```

file6.242 - Notepad
File Edit Format View Help
<242>
<equations>
<equation> invalid file: missing end tag
</equations>
</242>
Ln 3, Col 12 100% Windows (CRLF) UTF-8

```

- If you click in a file from the file sections, it will be loaded. (E.g. if we click the **file2.242** in **Fig 1** the file will be displayed as in **Fig 2**). And so on.

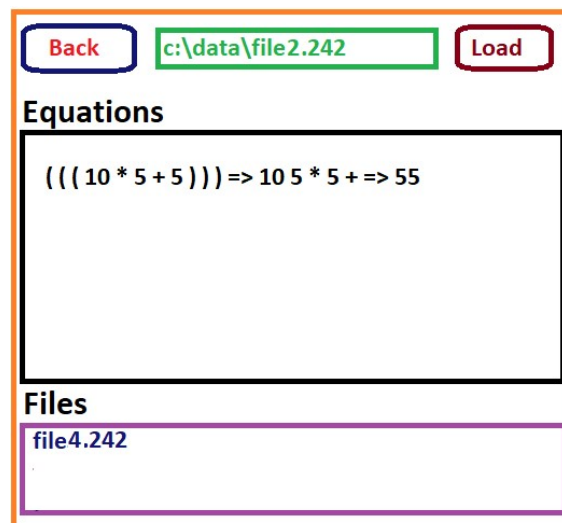
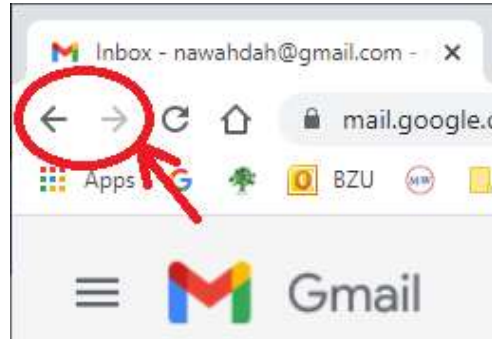


Fig 2: Loading file2.242

- While in Fig 2, if you click the **Back** button, it will go back to the previous loaded file (e.g. in this case **file1.242**). And so on. If you reached the 1st loaded file the back button should be disabled. (Note this is similar to the history buttons found in all the web browsers. See the following screenshot from the chrome browser:



When the user click the previous button (left arrow in chrome), the application will reload the previous URL (if applicable)). This is one application of stack as well.

**To make the project more interesting: we need you to implement the stack using a cursor array.
Yahooooo!**

Good Luck!