Project 4: File System

Implement a FUSE file system for converting HTML files between the following character encodings: ISO8859-1, ISO8859-9, UTF-8, UTF-16, UTF-32. The file system will operate on an underlying folder which contains a number of HTML files in these encodings. On the top level folder of the FUSE file system every encoding will be represented as a folder and these folders will contain converted versions of the original HTML files in that encoding.

For example, let us assume that the original folder contains two HTML files: x.html and y.html. The FUSE file system will have the following structure:

```
/
  UTF-8/
    x.html
    y.html
  UTF-32/
    x.html
    y.html
  ...
```

The "/UTF-8/x.html" file will be the UTF-8 encoded version of the original "x.html" file and the "/UTF-32/x.html" file will be the UTF-32 encoded version.

Note that the conversion between encodings will be done "on-the-fly", that means that you will not generate new disk files for the converted versions, you will generate the converted files every time they are requested.

The encoding of an HTML file can be found in the "head" section as in:

```
<html>
  <head>
    <meta charset="iso8859-9" />
    ...
```

You can assume that all input HTML files have properly formatted and correct input encoding declarations. You should also note that in addition to do the encoding conversion, you have to fix the meta tag in the converted file.

You can test your program by opening these files in a web browser and checking whether the browser correctly identifies the encoding (menu items: view -> character encoding or tools -> page info) and whether it displays the correct symbols or not.

You can use the libiconv or glib libraries or any library you like for handling the encoding conversion.