DISPLAYING A FILESYSTEM HIERARCHY (320/320 points)

In this exercise, we will pretty-print directory structures.

The prelude gives the types that we will use to represent directory structures. A `node` in the `filesystem` is either a simple `File`, a `Dir`ectory that contains a nested `filesystem`, or a `Symlink`.
The latter, as on Unix systems, is a fake file that redirects to another file. For this, it provides the relative path to this target file. The path is the list of directory to traverse to get to the target file, followed by the later. If one has to go a directory up, we use the common `".."` directory name that represents the parent directory.
A `filesystem` is a list of named nodes. An example `filesystem` is given below, in the format that you will have to produce. Don't worry, we'll break this piece by piece.

```
/photos
| /march
| | photo_1.bmp
| | photo_2.bmp
| | photo_3.bmp
| | index.html
| /april
| | photo_1.bmp
| | photo_2.bmp
| | index.html
/videos
| video1.avi
| video2.avi
| video3.avi
| video4.avi
| best.avi -> video4.avi
| index.html
/indexes
| videos.html -> ../videos/index.html
| photos_march.html -> ../photos/march/index.html
| photos_april.html -> ../photos/april/index.html
| photos_may.html -> INVALID
```

This output was generated from the following OCaml structure.

```
[ "photos", Dir
    [ "march", Dir
        [ "photo_1.bmp", File ;
          "photo_2.bmp", File ;
          "photo_3.bmp", File ;
          "index.html", File ] ;
      "april", Dir
        [ "photo_1.bmp", File ;
          "photo_2.bmp", File ;
          "index.html", File ] ] ;
  "videos", Dir
    [ "video1.avi", File ;
      "video2.avi", File ;
      "video3.avi", File ;
      "video4.avi", File ;
      "best.avi", Symlink [ "video4.avi" ] ;
      "index.html", File ] ;
  "indexes", Dir
    [ "videos.html",
      Symlink [ ".." ; "videos" ; "index.html" ] ;
      "photos_march.html",
      Symlink [ ".." ; "photos" ; "march" ; "index.html" ] ;
      "photos_april.html",
      Symlink [ ".." ; "photos" ; "april" ; "index.html" ] ;
      "photos_may.html",
      Symlink [ ".." ; "photos" ; "may" ; "index.html" ] ] ]
```

2. As you can see in the example, the depth of a file in the `filesystem` (the number of nested folders that are its ancestors) is represented by a sequence of vertical lines. Write a function `print_file: int -> string -> unit` that prints a file name, with the given number of `"| "` in front of it.
3. Write a similar function `print_symlink: int -> string -> string list -> unit` that prints the link name, with the given number of `"| "` in front of it, and the relative path (preceded by an arrow `" -> "`).
4. Write a similar function `print_dir: int -> string -> unit` that prints the dir name, with the given number of `"| "` in front of it, and the prepended `'/'`.
5. Write a function `print_filesystem: filesystem -> unit` that traverses the filesystem, producing the same display as in the example. You will probably need an auxiliary, recursive function, and you will have to use the previous answers.
6. Write a function `resolve: string list -> string list -> string list`. It takes as parameters:
    1. The full path from the root to a symlink, including its name. In the given example, that could be for instance `[ "indexes" ; "photos_april.html" ]`.
    2. The relative path for this symlink. Here, that would be `[ ".." ; "photos" ; "april" ; "index.html" ]`.

    The function returns the full path from the root to the target of the symlink. Here, we should get `[ "photos" ; "april" ; "index.html" ]`. Note that it may not be as easy as it seems, so you may think about it before plunging into the code.
7. Write a function `file_exists : filesystem -> string list -> bool` that tells if a file exists in the `filesystem`. The path is the full absolute path to the file, and the target must be a `File`, not a `Dir` or a `Symlink`.
8. Update your function `print_filesystem: filesystem -> unit` so that it replaces the printed relative path by `"INVALID"` when the symlink cannot be resolved to an existing file.

THE GIVEN PRELUDE

```
type filesystem =
  (string * node) list
and node =
  | File
  | Dir of filesystem
  | Symlink of string list
```

YOUR OCAML ENVIRONMENT

```
6
7   let rec print_file lvl name = match lvl with
8     | 0 -> print_string name
9     | _ -> print_string "|" ; print_file (lvl - 1) name
10  ;;
11
12  let print_symlink lvl name path =
13    print_file lvl name ; print_string " -> " ; print_path path
14  ;;
15
16  let rec print_dir lvl name = match lvl with
17    | 0 -> print_string "/" ; print_string name
18    | _ -> print_string "|" ; print_dir (lvl - 1) name
19  ;;
20
21  let print_filesystem root =
22    let rec print_filesysteme lvl items = match items with
23      | [] -> ()
24      | hd::tl -> match hd with
25        | (name, file) -> match file with
26          | File -> print_file lvl name ; print_string "\n" ;
27              print_filesysteme lvl tl
28          | Symlink path -> print_symlink lvl name path ; print_string "\n" ;
29              print_filesysteme lvl tl
30          | Dir system -> print_dir lvl name ; print_string "\n" ;
31              print_filesysteme (lvl + 1) system ;
32              print_filesysteme lvl tl
33    in
```


Exercise complete (click for details) — 320 pts

v Exercise 1: `print_path` — Completed, 50 pts

Found `print_path` with compatible type.

Computing `print_path [".."; "funny.txt"]`

Expected output — 5 pts
```
../funny.txt
```

Computing `print_path ["opt"; "report.doc"]`

Expected output — 5 pts
```
opt/report.doc
```

Computing
```
print_path
  [".."; "local"; "opt"; ".."; ".."; "Users"; "Users"; "htaccess.wav"]
```

Expected output — 5 pts
```
../local/opt/../../Users/Users/htaccess.wav
```

Computing
```
print_path
  ["Users"; "local"; "local"; "local"; "My Documents"; "My Documents"; "..";
   ".passwd.wav"]
```

Expected output — 5 pts
```
Users/local/local/local/My Documents/My Documents/../.passwd.wav
```

Computing `print_path ["My Documents"; ".."; ".."; "My Documents"; ".."; "id_rsa.ppt"]`

Expected output — 5 pts
```
My Documents/../../My Documents/../id_rsa.ppt
```

Computing `print_path ["opt"; "Users"; "opt"; "config.html"]`

Expected output — 5 pts
```
opt/Users/opt/config.html
```

Computing `print_path ["Users"; "htaccess.txt"]`

Expected output — 5 pts
```
Users/htaccess.txt
```

Computing
```
print_path
  ["local"; "My Documents"; ".."; "My Documents"; "opt"; ".."; ".report.ppt"]
```

Expected output — 5 pts
```
local/My Documents/../My Documents/opt/../.report.ppt
```

Computing `print_path ["Users"; "passwd.html"]`

Expected output — 5 pts
```
Users/passwd.html
```

Computing `print_path ["opt"; ".."; "opt"; ".."; "local"; ".id_rsa.doc"]`

Expected output — 5 pts
```
opt/../opt/../local/.id_rsa.doc
```

v Exercise 2: `print_file` — Completed, 50 pts

Found `print_file` with compatible type.

Computing `print_file 2 "config.ppt"`

Expected output — 5 pts
```
||config.ppt
```

Computing `print_file 0 "funny.html"`

Expected output — 5 pts
```
funny.html
```

Computing `print_file 2 "config.doc"`

Expected output — 5 pts

```
  |||passwd.wav
Computing print_file 2 "report.html"
```
Expected output 5 pts
```
  ||report.html
Computing print_file 2 "id_rsa.txt"
```
Expected output 5 pts
```
  ||id_rsa.txt
Computing print_file 0 "funny.ppt"
```
Expected output 5 pts
```
  funny.ppt
Computing print_file 0 "htaccess.wav"
```
Expected output 5 pts
```
  htaccess.wav
Computing print_file 4 ".passwd.doc"
```
Expected output 5 pts
```
  ||||.passwd.doc
Computing print_file 3 "funny.txt"
```
Expected output 5 pts
```
  |||funny.txt
```

v Exercise 3: `print_symlink` Completed, 50 pts

Found `print_symlink` with compatible type.

```
Computing
  print_symlink
    3
    ".id_rsa.html"
    [".."; ".."; "My Documents"; ".."; "Users"; ".."; "htaccess.doc"]
```
Expected output 5 pts
```
  |||.id_rsa.html -> ../../My Documents/../Users/../htaccess.doc
Computing print_symlink 0 "report.wav" [".."; ".."; ".."; "config.html"]
```
Expected output 5 pts
```
  report.wav -> ../../../config.html
Computing print_symlink 0 "config.txt" [".."; ".id_rsa.txt"]
```
Expected output 5 pts
```
  config.txt -> ../.id_rsa.txt
Computing
  print_symlink
    2
    "htaccess.wav"
    [".."; ".."; ".."; "local"; "opt"; ".."; "local"; "report.doc"]
```
Expected output 5 pts
```
  ||htaccess.wav -> ../../../local/opt/../local/report.doc
Computing print_symlink 3 ".funny.ppt" [".."; ".."; "Users"; "passwd.ppt"]
```
Expected output 5 pts
```
  |||.funny.ppt -> ../../Users/passwd.ppt
Computing print_symlink 0 "id_rsa.txt" ["passwd.wav"]
```
Expected output 5 pts
```
  id_rsa.txt -> passwd.wav
Computing
  print_symlink
    3
    "report.wav"
    ["local"; "local"; ".."; "My Documents"; "My Documents"; "funny.txt"]
```
Expected output 5 pts
```
  |||report.wav -> local/local/../My Documents/My Documents/funny.txt
Computing
  print_symlink
    2
    "htaccess.doc"
    ["opt"; ".."; "My Documents"; ".."; "Users"; "Users"; "opt"; "config.ppt"]
```
Expected output 5 pts
```
  ||htaccess.doc -> opt/../My Documents/../Users/Users/opt/config.ppt
Computing
  print_symlink
    2
    "report.doc"
    ["Users"; "My Documents"; ".."; "opt"; ".."; ".."; "opt"; "passwd.ppt"]
```
Expected output 5 pts
```
  ||report.doc -> Users/My Documents/../opt/../../opt/passwd.ppt
Computing
  print_symlink
    3
    "config.html"
    ["Users"; ".."; "local"; "local"; "htaccess.html"]
```
Expected output 5 pts
```
  |||config.html -> Users/../local/local/htaccess.html
```

v Exercise 4: `print_dir` Completed, 50 pts

```
  |/My Documents
```
Computing `print_dir 0 "Users"`

Expected output 5 pts
```
  /Users
```
Computing `print_dir 2 "local"`

Expected output 5 pts
```
  ||/local
```
Computing `print_dir 3 "local"`

Expected output 5 pts
```
  |||/local
```
Computing `print_dir 3 "opt"`

Expected output 5 pts
```
  |||/opt
```
Computing `print_dir 2 "My Documents"`

Expected output 5 pts
```
  ||/My Documents
```
Computing `print_dir 0 "My Documents"`

Expected output 5 pts
```
  /My Documents
```
Computing `print_dir 4 "Users"`

Expected output 5 pts
```
  ||||/Users
```
Computing `print_dir 1 "opt"`

Expected output 5 pts
```
  |/opt
```
Computing `print_dir 2 "opt"`

Expected output 5 pts
```
  ||/opt
```

∨ Exercise 5: `print_filesystem` Completed, 50 pts

Found `print_filesystem` with compatible type.

Computing
```
  print_filesystem
    [("config.doc", File); ("funny.html", File); ("id_rsa.ppt", File);
     ("passwd.doc", Symlink ["funny.html"])]
```
Expected output 5 pts
```
  config.doc
  funny.html
  id_rsa.ppt
  passwd.doc -> funny.html
```
Computing
```
  print_filesystem
    [(".config.html", File);
     ("Users",
      Dir
       [("My Documents",
         Dir
          [("config.doc", Symlink ["id_rsa.doc"]); ("config.txt", File);
           ("htaccess.html", File); ("htaccess.wav", File);
           ("id_rsa.doc", File);
           ("report.wav", Symlink [".."; "Users"; "funny.html"])]);
        ("Users",
         Dir
          [(".funny.doc", Symlink [".."; ".."; ".config.html"]);
           ("funny.html", File)]);
        ("opt",
         Dir
          [(".funny.ppt", File);
           ("report.txt", Symlink [".."; ".."; ".config.html"])])]);
     ("id_rsa.html", Symlink [".config.html"])]
```
Expected output 5 pts
```
  .config.html
  /Users
  |/My Documents
  ||config.doc -> id_rsa.doc
  ||config.txt
  ||htaccess.html
  ||htaccess.wav
  ||id_rsa.doc
  ||report.wav -> ../Users/funny.html
  |/Users
  ||.funny.doc -> ../../.config.html
  ||funny.html
  |/opt
  ||.funny.ppt
  ||report.txt -> ../../.config.html
  id_rsa.html -> .config.html
```
Computing `print_filesystem [("funny.ppt", File)]`

Expected output 5 pts

```
   [("Users",
     Dir
      [(".id_rsa.html", File); ("config.doc", Symlink ["htaccess.html"]);
       ("funny.txt", File); ("htaccess.html", File);
       ("opt",
        Dir
         [(".htaccess.wav", File); ("config.html", File);
          ("htaccess.txt", File); ("id_rsa.ppt", Symlink [".."; "funny.txt"]);
          ("report.html", Symlink [".."; "htaccess.html"]);
          ("report.txt", Symlink [".."; "htaccess.html"])]);
       ("passwd.wav", Symlink [".."; "local"; "My Documents"; "funny.doc"])]);
    ("local",
     Dir
      [("My Documents",
        Dir
         [("funny.doc", File);
          ("passwd.ppt", Symlink [".."; ".."; "report.txt"]);
          ("report.ppt", File)]);
       ("config.txt", File); ("htaccess.ppt", File)]);
    ("report.txt", File)]
```

Expected output 5 pts

```
/Users
|.id_rsa.html
|config.doc -> htaccess.html
|funny.txt
|htaccess.html
|/opt
||.htaccess.wav
||config.html
||htaccess.txt
||id_rsa.ppt -> ../funny.txt
||report.html -> ../htaccess.html
||report.txt -> ../htaccess.html
|passwd.wav -> ../local/My Documents/funny.doc
/local
|/My Documents
||funny.doc
||passwd.ppt -> ../../report.txt
||report.ppt
|config.txt
|htaccess.ppt
report.txt
```

Computing

```
print_filesystem
  [("My Documents",
    Dir
     [("config.doc", File); ("config.ppt", Symlink ["config.doc"]);
      ("funny.html", File);
      ("id_rsa.wav", Symlink [".."; "Users"; ".funny.wav"]);
      ("passwd.txt", File)]);
   ("Users", Dir [(".funny.wav", File)])]
```

Expected output 5 pts

```
/My Documents
|config.doc
|config.ppt -> config.doc
|funny.html
|id_rsa.wav -> ../Users/.funny.wav
|passwd.txt
/Users
|.funny.wav
```

Computing

```
print_filesystem
  [(".htaccess.doc", File); ("config.doc", Symlink ["id_rsa.ppt"]);
   ("config.txt", File); ("htaccess.wav", File);
   ("id_rsa.html", Symlink ["config.txt"]); ("id_rsa.ppt", File)]
```

Expected output 5 pts

```
.htaccess.doc
config.doc -> id_rsa.ppt
config.txt
htaccess.wav
id_rsa.html -> config.txt
id_rsa.ppt
```

Computing

```
print_filesystem
  [("config.html", File); ("funny.html", File); ("passwd.ppt", File);
   ("passwd.txt", Symlink ["funny.html"]);
   ("report.wav", Symlink ["passwd.ppt"])]
```

Expected output 5 pts

```
config.html
funny.html
passwd.ppt
passwd.txt -> funny.html
report.wav -> passwd.ppt
```

Computing

```
      ("opt",
       Dir
        [(".report.doc", File); ("htaccess.txt", File);
         ("passwd.doc", Symlink [".."; ".."; "id_rsa.txt"])])]);
    ("funny.txt", File); ("id_rsa.txt", File)]
```

Expected output 5 pts

```
/Users
|.id_rsa.wav
|/opt
||.report.doc
||htaccess.txt
||passwd.doc -> ../../id_rsa.txt
funny.txt
id_rsa.txt
```

Computing

```
print_filesystem
  [("My Documents",
    Dir
     [(".htaccess.ppt", File); ("funny.wav", Symlink [".htaccess.ppt"]);
      ("id_rsa.doc", File); ("report.html", Symlink ["id_rsa.doc"])])]
```

Expected output 5 pts

```
/My Documents
|.htaccess.ppt
|funny.wav -> .htaccess.ppt
|id_rsa.doc
|report.html -> id_rsa.doc
```

Computing

```
print_filesystem
  [(".config.html", File); (".funny.html", File); ("report.txt", File)]
```

Expected output 5 pts

```
.config.html
.funny.html
report.txt
```

v Exercise 6: `resolve` Completed, 10 pts

Found `resolve` with compatible type.

Computing

```
resolve
  ["opt"; "Users"; "Users"; "local"; "id_rsa.ppt"]
  [".."; "Users"; "My Documents"; ".passwd.wav"]
```

Correct value `["opt"; "Users"; "Users"; "Users"; "My Documents"; ".passwd.wav"]` 1 pt

Computing `resolve ["My Documents"; "config.txt"] ["local"; "passwd.wav"]`

Correct value `["My Documents"; "local"; "passwd.wav"]` 1 pt

Computing

```
resolve
  ["opt"; "My Documents"; "opt"; "Users"; "local"; "id_rsa.doc"]
  [".."; "opt"; ".."; "report.doc"]
```

Correct value `["opt"; "My Documents"; "opt"; "Users"; "report.doc"]` 1 pt

Computing

```
resolve
  ["opt"; "My Documents"; "Users"; "htaccess.ppt"]
  ["local"; ".funny.doc"]
```

Correct value `["opt"; "My Documents"; "Users"; "local"; ".funny.doc"]` 1 pt

Computing

```
resolve
  ["opt"; "My Documents"; "Users"; "My Documents"; "local"; "passwd.html"]
  [".."; "Users"; "local"; "local"; "My Documents"; ".htaccess.txt"]
```

Correct value 1 pt

```
["opt"; "My Documents"; "Users"; "My Documents"; "Users"; "local"; "local";
 "My Documents"; ".htaccess.txt"]
```

Computing

```
resolve
  ["opt"; "My Documents"; "Users"; "opt"; "Users"; "id_rsa.wav"]
  [".."; ".."; "Users"; "local"; "funny.wav"]
```

Correct value `["opt"; "My Documents"; "Users"; "Users"; "local"; "funny.wav"]` 1 pt

Computing `resolve ["report.doc"] [".."; "opt"; "config.txt"]`

Correct value `["opt"; "config.txt"]` 1 pt

Computing `resolve ["local"; "My Documents"; "config.html"] ["Users"; "id_rsa.ppt"]`

Correct value `["local"; "My Documents"; "Users"; "id_rsa.ppt"]` 1 pt

Computing

```
resolve
  ["opt"; "opt"; "local"; "My Documents"; "funny.txt"]
  [".."; ".."; "Users"; ".."; "My Documents"; "htaccess.html"]
```

Correct value `["opt"; "opt"; "My Documents"; "htaccess.html"]` 1 pt

Computing

```
resolve
  ["My Documents"; "local"; "passwd.doc"]
  ["opt"; ".."; "My Documents"; ".."; "local"; "report.wav"]
```

Correct value `["My Documents"; "local"; "local"; "report.wav"]` 1 pt

v Exercise 7: `file_exists` Completed, 10 pts

```
   [("passwd.ppt", Symlink ["htaccess.wav"]); ("htaccess.wav", File)]
   ["Users"; "local"; "Users"; "opt"; "report.doc"]
```
Correct value false 1 pt

```
Computing
  file_exists
    [("opt",
      Dir
       [(".id_rsa.html", File); ("funny.html", File);
        ("htaccess.ppt", Symlink ["funny.html"]);
        ("passwd.doc", Symlink ["funny.html"])])]
    ["report.wav"]
```
Correct value false 1 pt

```
Computing
  file_exists
    [("funny.ppt", File); ("htaccess.wav", File)]
    ["Users"; "Users"; "My Documents"; "report.txt"]
```
Correct value false 1 pt

```
Computing
  file_exists
    [("My Documents",
      Dir
       [("My Documents",
         Dir
          [(".config.wav", File);
           ("config.html", Symlink [".."; "Users"; "htaccess.html"]);
           ("funny.doc", File);
           ("id_rsa.html", Symlink [".."; "Users"; ".report.ppt"]);
           ("passwd.doc", File); ("passwd.wav", File)]);
        ("Users",
         Dir
          [(".config.txt", Symlink ["report.ppt"]); (".id_rsa.txt", File);
           (".report.ppt", File); ("config.ppt", Symlink [".report.ppt"]);
           ("htaccess.html", File);
           ("passwd.html", Symlink [".."; "My Documents"; "funny.doc"]);
           ("report.ppt", File)]);
        ("funny.html", File);
        ("opt",
         Dir
          [("htaccess.txt", File);
           ("id_rsa.ppt", Symlink [".."; "funny.html"]); ("report.doc", File)])])]
    ["My Documents"; "opt"; "report.doc"]
```
Correct value true 1 pt

```
Computing
  file_exists
    [("config.txt", File); ("local", Dir [("htaccess.html", File)]);
     ("passwd.wav", Symlink ["config.txt"])]
    ["config.txt"]
```
Correct value true 1 pt

```
Computing
  file_exists
    [(".funny.txt", File);
     ("Users",
      Dir
       [(".id_rsa.doc", File); ("config.ppt", File); ("htaccess.html", File)]);
     ("local",
      Dir
       [(".id_rsa.doc", File);
        ("My Documents",
         Dir
          [(".config.txt", File);
           ("htaccess.html", Symlink [".."; ".."; ".funny.txt"]);
           ("htaccess.wav", File); ("report.txt", File)]);
        ("Users",
         Dir
          [(".funny.wav", File);
           ("report.wav", Symlink [".."; "My Documents"; "report.txt"])]);
        ("config.ppt", Symlink [".."; "Users"; "config.ppt"]);
        ("local",
         Dir
          [(".id_rsa.doc", File);
           (".passwd.wav", Symlink [".."; ".id_rsa.doc"]);
           ("funny.txt", Symlink [".."; "Users"; ".funny.wav"]);
           ("id_rsa.doc", File); ("passwd.ppt", Symlink [".."; ".id_rsa.doc"]);
           ("report.html", File)])])]
    ["local"; "My Documents"; ".config.txt"]
```
Correct value true 1 pt

```
Computing
  file_exists
    [("My Documents",
      Dir
       [(".config.html", File);
        (".funny.html", Symlink [".."; "local"; "opt"; ".htaccess.doc"]);
        ("config.ppt", File); ("funny.txt", Symlink [".config.html"]);
        ("passwd.doc", File); ("report.txt", File)]);
```

```
       ("opt",
        Dir
         [(".htaccess.doc", File); (".id_rsa.html", File);
          ("passwd.txt", Symlink [".."; "id_rsa.wav"]); ("report.wav", File)]);
       ("report.ppt", File)])]
   ["My Documents"; "opt"; "local"; "local"; "Users"; "opt"; "Users";
    "config.txt"]
```

Correct value `false` 1 pt

```
Computing
  file_exists
    [("My Documents",
      Dir
       [("Users", Dir [(".config.wav", File); (".report.html", File)]);
        ("funny.txt", File);
        ("id_rsa.doc", Symlink [".."; "local"; "Users"; "id_rsa.txt"])]);
     ("Users",
      Dir
       [("funny.doc", File);
        ("local",
         Dir
          [(".passwd.wav",
            Symlink [".."; ".."; "local"; "Users"; "id_rsa.txt"]);
           (".report.html", Symlink [".."; ".."; "config.ppt"]);
           ("htaccess.ppt", File); ("passwd.txt", File); ("report.doc", File)])]);
     ("config.ppt", File); ("funny.doc", Symlink ["config.ppt"]);
     ("local",
      Dir
       [("Users",
         Dir
          [("id_rsa.txt", File);
           ("passwd.ppt", Symlink [".."; ".."; "Users"; "funny.doc"])]);
        ("funny.ppt", File); ("htaccess.wav", File);
        ("report.doc", Symlink [".."; "My Documents"; "Users"; ".config.wav"])])]
    ["My Documents"; "opt"; "My Documents"; "opt"; "config.ppt"]
```

Correct value `false` 1 pt

```
Computing
  file_exists
    [("My Documents",
      Dir
       [("id_rsa.html", File);
        ("local",
         Dir
          [(".id_rsa.doc", Symlink ["htaccess.ppt"]); ("config.txt", File);
           ("funny.ppt", File); ("htaccess.ppt", File)])]);
     ("config.doc", Symlink ["htaccess.ppt"]); ("htaccess.ppt", File)]
    ["opt"; "Users"; "opt"; "My Documents"; "local"; "id_rsa.html"]
```

Correct value `false` 1 pt

```
Computing
  file_exists
    [("My Documents",
      Dir
       [("My Documents",
         Dir
          [(".funny.ppt", File);
           (".id_rsa.ppt", Symlink [".."; "Users"; "htaccess.txt"])]);
        ("Users",
         Dir
          [(".htaccess.html", Symlink [".."; ".."; "id_rsa.txt"]);
           ("config.html", Symlink [".."; "My Documents"; ".funny.ppt"]);
           ("funny.ppt", File); ("htaccess.txt", File);
           ("id_rsa.txt", Symlink [".."; ".."; "opt"; "opt"; ".config.ppt"]);
           ("passwd.doc", Symlink [".."; ".."; "id_rsa.txt"]);
           ("passwd.wav", File); ("report.html", File)]);
        ("config.wav", File); ("report.html", Symlink [".."; "id_rsa.txt"])]);
     ("id_rsa.txt", File);
     ("opt",
      Dir
       [("opt",
         Dir
          [(".config.ppt", File);
           ("htaccess.doc",
            Symlink [".."; ".."; "My Documents"; "Users"; "htaccess.txt"])])]);
     ("passwd.html", Symlink ["id_rsa.txt"])]
    ["id_rsa.txt"]
```

Correct value `true` 1 pt

v Exercise 8: `print_filesystem` Completed, 50 pts

Found `print_filesystem` with compatible type.

```
Computing
  print_filesystem
    [("My Documents",
      Dir [("funny.doc", Symlink ["report.ppt"]); ("report.ppt", File)]);
     ("passwd.txt", File)]
```

Expected output 5 pts

passwd.txt

Computing `print_filesystem [(".funny.doc", File)]`

Expected output 5 pts

```
.funny.doc
```

Computing

```
print_filesystem
  [("config.doc", File);
   ("htaccess.html", Symlink ["local"; "Users"; "report.wav"]);
   ("local",
    Dir
     [("Users", Dir [("report.wav", File)]); ("funny.html", File);
      ("passwd.txt", File)]);
   ("opt",
    Dir
     [(".htaccess.txt", File);
      ("config.txt", Symlink [".."; "local"; "passwd.txt"]);
      ("opt",
       Dir
        [(".funny.html",
          Symlink
           ["My Documents"; "opt"; ".."; "opt"; "My Documents";
            "htaccess.wav"]);
         (".id_rsa.doc", File); ("htaccess.ppt", File); ("id_rsa.wav", File)]);
      ("report.ppt", File)])]
```

Expected output 5 pts

```
config.doc
htaccess.html -> local/Users/report.wav
/local
|/Users
||report.wav
|funny.html
|passwd.txt
/opt
|.htaccess.txt
|config.txt -> ../local/passwd.txt
|/opt
||.funny.html -> INVALID
||.id_rsa.doc
||htaccess.ppt
||id_rsa.wav
|report.ppt
```

Computing

```
print_filesystem
  [("Users",
    Dir
     [(".passwd.doc", Symlink ["Users"; "id_rsa.doc"]);
      ("Users",
       Dir
        [("config.wav", File); ("id_rsa.doc", File);
         ("passwd.ppt", Symlink ["config.wav"])]);
      ("funny.wav", File);
      ("htaccess.wav", Symlink ["Users"; "local"; ".."; "id_rsa.txt"]);
      ("report.html", File)]);
   ("opt",
    Dir
     [("My Documents",
       Dir
        [(".config.doc", File); (".funny.html", File);
         (".htaccess.ppt", File); ("passwd.ppt", File)])])]
```

Expected output 5 pts

```
/Users
|.passwd.doc -> Users/id_rsa.doc
|/Users
||config.wav
||id_rsa.doc
||passwd.ppt -> config.wav
|funny.wav
|htaccess.wav -> INVALID
|report.html
/opt
|/My Documents
||.config.doc
||.funny.html
||.htaccess.ppt
||passwd.ppt
```

Computing

```
print_filesystem
  [(".id_rsa.ppt", File); (".passwd.ppt", File);
   ("My Documents",
    Dir
     [(".htaccess.txt", Symlink ["local"; "Users"; ".report.txt"]);
      (".passwd.doc", Symlink ["htaccess.wav"]); (".report.txt", File);
      ("funny.doc",
       Symlink
        ["local"; "My Documents"; ".."; ".."; "local"; "My Documents";
```

```
        [(".config.doc", File); ("config.wav", Symlink [".."; "config.ppt"]);
         ("htaccess.doc", File)]);
       ("report.html", File)]);
    ("funny.html", Symlink [".passwd.ppt"])]
```

Expected output 5 pts

```
.id_rsa.ppt
.passwd.ppt
/My Documents
|.htaccess.txt -> INVALID
|.passwd.doc -> htaccess.wav
|.report.txt
|funny.doc -> INVALID
|htaccess.wav
|/opt
||.config.doc
||config.wav -> INVALID
||htaccess.doc
|report.html
funny.html -> .passwd.ppt
```

Computing

```
print_filesystem
  [("config.doc", Symlink [".."; "Users"; ".."; ".."; "passwd.ppt"]);
   ("opt",
    Dir
     [(".report.txt", File); ("funny.html", File); ("htaccess.wav", File);
      ("passwd.ppt", Symlink [".."; "report.wav"])]);
   ("report.wav", File)]
```

Expected output 5 pts

```
config.doc -> INVALID
/opt
|.report.txt
|funny.html
|htaccess.wav
|passwd.ppt -> ../report.wav
report.wav
```

Computing

```
print_filesystem
  [("My Documents",
    Dir
     [("My Documents",
       Dir
        [("funny.doc", File); ("htaccess.txt", File);
         ("passwd.txt", Symlink [".."; ".."; "local"; "id_rsa.doc"])]);
      ("Users",
       Dir
        [("config.ppt", File);
         ("report.wav",
          Symlink [".."; "local"; ".."; "Users"; ".."; ".."; "passwd.ppt"])]);
      ("funny.html", Symlink [".."; "local"; "passwd.wav"]);
      ("id_rsa.html", File); ("passwd.doc", File)]);
   ("id_rsa.txt", File);
   ("local",
    Dir
     [("config.txt",
       Symlink [".."; "My Documents"; "My Documents"; "htaccess.txt"]);
      ("funny.ppt", File); ("htaccess.wav", Symlink [".."; "id_rsa.txt"]);
      ("id_rsa.doc", File); ("passwd.wav", File)])]
```

Expected output 5 pts

```
/My Documents
|/My Documents
||funny.doc
||htaccess.txt
||passwd.txt -> ../../local/id_rsa.doc
|/Users
||config.ppt
||report.wav -> INVALID
|funny.html -> ../local/passwd.wav
|id_rsa.html
|passwd.doc
id_rsa.txt
/local
|config.txt -> ../My Documents/My Documents/htaccess.txt
|funny.ppt
|htaccess.wav -> ../id_rsa.txt
|id_rsa.doc
|passwd.wav
```

Computing

```
print_filesystem
  [("Users",
    Dir
     [(".config.txt",
       Symlink [".."; ".."; "opt"; ".."; ".."; ".."; "report.ppt"]);
      ("htaccess.ppt", Symlink [".."; "funny.html"]); ("id_rsa.wav", File);
      ("report.doc", File)]);
   ("funny.html", File);
```

```
    ("report.ppt", Symlink ["opt"; "funny.wav"])]
```

Expected output 5 pts

```
/Users
|.config.txt -> INVALID
|htaccess.ppt -> ../funny.html
|id_rsa.wav
|report.doc
funny.html
/opt
|config.html
|funny.wav
|passwd.txt -> ../funny.html
report.ppt -> opt/funny.wav
```

Computing

```
print_filesystem
  [("funny.txt", File);
   ("local",
    Dir
     [(".id_rsa.html",
       Symlink
        [".."; "opt"; "My Documents"; "local"; "Users"; "opt"; "..";
         "htaccess.html"]);
      ("My Documents",
       Dir
        [(".id_rsa.ppt", File); (".passwd.doc", File); ("funny.doc", File);
         ("htaccess.html",
          Symlink [".."; "local"; "My Documents"; "Users"; "id_rsa.html"]);
         ("report.ppt", File)]);
      ("htaccess.wav", File);
      ("local",
       Dir
        [("config.txt", File); ("funny.html", File); ("htaccess.txt", File);
         ("id_rsa.ppt", File); ("report.txt", Symlink ["funny.html"])])]);
   ("opt", Dir [(".id_rsa.wav", File)]);
   ("passwd.wav", Symlink ["funny.txt"]); ("report.txt", File)]
```

Expected output 5 pts

```
funny.txt
/local
|.id_rsa.html -> INVALID
|/My Documents
||.id_rsa.ppt
||.passwd.doc
||funny.doc
||htaccess.html -> INVALID
||report.ppt
|htaccess.wav
|/local
||config.txt
||funny.html
||htaccess.txt
||id_rsa.ppt
||report.txt -> funny.html
/opt
|.id_rsa.wav
passwd.wav -> funny.txt
report.txt
```

Computing

```
print_filesystem
  [("opt",
    Dir
     [("config.doc", File);
      ("local",
       Dir [(".report.wav", File); ("id_rsa.html", Symlink [".report.wav"])]);
      ("opt",
       Dir
        [(".funny.doc", File); ("config.ppt", Symlink [".."; "funny.txt"]);
         ("funny.doc", File); ("passwd.ppt", File);
         ("report.txt", Symlink ["My Documents"; "passwd.txt"])])])]
```

Expected output 5 pts

```
/opt
|config.doc
|/local
||.report.wav
||id_rsa.html -> .report.wav
|/opt
||.funny.doc
||config.ppt -> INVALID
||funny.doc
||passwd.ppt
||report.txt -> INVALID
```