cat

Jon Sporring

November 21, 2019

1 Lærervejledningn

Emne working with files

Sværhedsgrad Easy

2 Introduktion

`cat` is a UNIX-program, which concatenates (i.e. joins) files. The program exists on both Linux and macOS.

When passing two text files to `cat`, e.g. `a.txt` and `b.txt`, the program prints the contents of file `a.txt` followed by the contents of `b.txt` to the screen.

UNIX also has an inverse version of `cat`, `tac`, which prints the files in reverse order.

For example, if the file `a.txt` contains the characters `aaa\nbbb\n` and the file `b.txt` contains the characters `ccc\nddd\n`, then

```
cat a.txt b.txt
```

will output `aaa\nbbb\nccc\nddd\n` to the screen. In contrast,

```
tac a.txt b.txt
```

will output `ddd\nccc\nbbb\naaa\n` to the screen.

In the following assignments you are to write a (functional) implementation of `cat` in F#.

3 Opgave(r)

1. Make a function,

   ```
   readFile : filename:string -> string option
   ```

   which takes a filename and returns the contents of the text file as a `string option`. If the file does not exist, the function should return `None`.

2. Make a function,

   ```
   printFile : filenname:string -> bool
   ```

   which prints the content of the file with the name `filename`. If no error occurs, then the function must return `true`, and otherwise `false`.

3. First write a function,

   ```
   cat : filenames:string list -> string
   ```

   which takes a list of filenames. The function should use `readFile` (exercise 1) to read the contents of the files. The contents of the files should be merged into a single `string`, which the function returns.

   Then write a program, `cat`, which takes a list of filenames as command line arguments, calls the `cat` function with this list and prints the resulting string to the screen using `printFile` (exercise 2).

4. First write a function,

   ```
   tac : filenames:string list -> string
   ```

   which takes a list of files, concatenates them and prints the result line by line in reverse order (i.e. the opposite of `cat` on a line-by-line basis), and returns the result as a string.

   Then write a program, `tac`, which takes a list of filenames as command line arguments, calls the `tac` function with this list and prints the resulting string to the screen using `printFile` (exercise 2).