

\$Id: ifstream.mm,v 1.13 2022-02-07 13:14:02-08 - - \$  
 /afs/cats.ucsc.edu/courses/csel111-wm/Assignments/asg3-listmap-templates/discussion-misc.d  
<https://www2.ucsc.edu/courses/csel111-wm/Assignments/asg3-listmap-templates/discussion-misc.d/>

## 1. Files in C++ — ifstream

Files can be read in C++ by using `<fstream>`'s class `ifstream`.

Following is a sample loop that can be used to iterator over `vector<string>` `filenames` and print the contents of the files. Assume a function `void catfile (istream&)` to use `getline`, etc., just as it would read `cin`. The complete program is in the file `catfile.cpp` in the `misc/` directory.

```
for (const auto& filename: filenames) {
    ifstream infile (filename);
    if (infile.fail()) {
        status = EXIT_FAILURE;
        cerr << progname << " : " << filename << " : " << strerror (errno) << endl;
    } else {
        catfile (infile, filename);
    }
}
```

## 2. Dissection of the code

- (a) `string filename` is the name of the file to open.
- (b) `ifstream infile (filename);`  
 An `ifstream` is used to read a file. This statement shows a ctor call with the filename as an argument. If the file can be opened for input, OK. If not, there is no exception. The open just quietly fails.
- (c) `infile.fail()` checks to see if the open failed. It is always necessary to check on success or failure.
- (d) `cerr << progname << " : " << filename << " : " << strerror (errno) << endl;`  
 Failure must always be reported properly in three parts:
  1. The name of the program reporting the failure. This should always be `basename(argv[0])`, which should be saved by `main` when the program starts.
  2. The name of the object or file causing trouble.
  3. The specific reason for failure.
    - `errno` is an external variable with an integer code indicating the precise reason for the failure.
    - `strerror(errno)` converts that into a human-readable string.
- (e) If the open succeeds, the file is read, processed according to whatever the program requires.
- (f) Closing the file is necessary whenever the file is opened. When an `ifstream` goes out of scope, the destructor is run, as for all objects. The destructor of an `ifstream` closes the file.

### 3. `basename(3)`

The function `basename(3)` is available from `<libgen.h>`. In the case that you are compiling on a system lacking that library, here is a drop-in replacement:

```
string basename (const string &name) {  
    return name.substr (name.rfind ('/') + 1);  
}
```

`rfind` returns the position of the rightmost character in the string. In the case of failure, it returns `string::npos`, which is the largest possible `size_t` value. Thus, `npos + 1` overflows to 0.