File Structures (2022)

Lab3

Some notes:

File mode flag	Default open mode
ios::in	Input mode. Information will be read from the file. If the file does not exist, it will not be created and the open function will fail.
ios::out	Output mode. Information will be written to the file. By default, the file's contents will be deleted if it already exists.
ios::app	Append mode. If the file already exists, its contents are preserved and all output is written to the end of the file. By default, this flag causes the file to be created if it does not exist.

Error handling functions

Function	Meaning
bool bad()	Returns: **(1)** If an invalid operation is attempted or any unrecoverable, error has occurred, and **(0)** otherwise
bool good()	Returns: **(1)** If no error has occurred. This means, everything is okay with the stream object and we can proceed to perform I/O operations. When it returns (**zero)**, no further operations carried out.
bool fail()	Returns non-zero **(true)** when an input or output operation has failed.
bool is_open()	Returns true if the stream is open.
bool eof()	Returns non-zero **(true value)** if end-of-file is encountered while reading; otherwise returns (**zero)** (false value).
clear()	Resets the error state so that further operations can be attempted.

Exercise 1:

Write c++ program to test the above error functions.

Exercise 2:

Write a c++ program that uses

- The **tellp()** function to know the position of writing pointer (when just open the file, while writing and after writing terminate).
- The **tellg()** function to know the position of reading pointer (when just open the file, while reading and after reading terminate).

Exercise 3:

Write a c++ program that uses the **seekp()** function to jump around to different locations in the file, writing a character after each stop.

Exercise 4:

Write a c++ program that uses the **seekg()** function to jump around to different locations in the file, retrieving a character after each stop.

Exercise 5:

Write a c++ program to compute the file size.

Write c++ program to read data from the file created in previous exercise **“file1.txt”** using **get(ch)** function and write the data into a new file named **“file2.txt”** after converting each letter from lower case to upper.