

Assignment UP-03

In syscall.h

Function modified:

```
static int create (const char *file, unsigned initial_size)
    Creates a file and returns status
static int remove (const char *file)
    Removes a file and returns status
static int open (const char *file)
    Opens a file and search for a null file pointer and assign fd to that file
static int filesize (int fd)
    Returns file size
static int read (int fd, void *buffer, unsigned size)
    Reads from file and keyboard input
static int write (int fd, const void *buffer, unsigned size)
    Write from buffer to stdout or file
static void seek (int fd, unsigned position)
    Takes the file pointer to given position
static int tell (int fd)
    Returns position of file pointer
static void close (int fd)
    Close given file with file descriptor fd
```

ALGORITHM:

1. First it validates the given file. Creates a new file called **file** initially **initial_size** bytes in size. Then returns status as 0 or 1.
2. First it validates the pointer to given file. Then it removes the file by calling `filesys_remove` and returns the status.
3. This function opens a file whose pointer is given. Then it returns fd for that file by allocating it to `thread->files` array. The index at which the pointer is stored in array is the fd for that file. If no space is free then it returns 12 indicating failure.
4. First this function validates the fd given for the file. If valid it returns the file size using `file_length` func. Otherwise returns -1.
5. This function reads from file fd and stores data in buffer. Finally it returns the bytes read.

If `fd = 1` means `STD_OUT` so in this case we do not read. Otherwise find the file with given `fd` value or read from `STDIN` using `input_getch()`. `file_read()` functions reads from a given file .

6. This functions writes data from buffer to file `fd`. Finally it returns the bytes write.
If `fd = 0` means `STDIN` so in this case we do not write. Otherwise find the file with given `fd` value or write to Console using `putbuf()`. `file_write()` functions write to a given file .
7. Seek function first validates the given file pointer to the file. Changes the next byte to be read or written in open file `fd` to position, expressed in bytes from the beginning of the file. For this it uses function `file_seek()` .
8. It validates the given `fd`. Then it returns the position of the next byte to be read or written in open file `fd`, expressed in bytes from the beginning of the file.
9. This function closes the given file and deallocates the `fd` of that file.