Design Document: Asg1

Jinghao Shen
CruzID: jshen30

1 Goals

The goal for this assignment is to implement a simple single-threaded RPC server that will support the math functions and file functions. The server will respond to a standard RPC protocol, which is given, providing the results of several file system functions(read, write, list, unlink, status) that the server will implement. The program will run the server in a directory, and requests for files will be served from under that directory.

2 Design

This program first set up sockets for basic client and server communication.
Then, convert values to and from the given big-endian input, wire format, to a data that can be stored in five different types of buffer, uint8_t, uint16_t, uint32_t, uint64_t, and uint8_t *.
Do several unit tests on this part.

For uint:

```
n = 0 //counter
While(not the end of input){
        uint8_t = Argv[] >> (sizeof(Argv[]) -  8*n) & 0xff
        uint16_t = Argv[] >> (sizeof(Argv[]) -  16*n) & 0xffff
        uint32_t = Argv[] >> (sizeof(Argv[]) -  32*n) & 0xffffffff
        uint64_t = Argv[] >> (sizeof(Argv[]) -  64*n) & 0xffffffffffffffff
        Argv = argv << 8 (16/32/64)
        N += 1
```

For words:

```
Get the first 16 bytes
Count = sizeof(string)
for(int i=0; i<count; i++){
        Uint8_t *t[i] = (char) argv[] >>(sizeof(argv[]) - 8*i) & 0xff
```

To do the math functions:

To do the file functions: