Introduction:
A system for whole file caching.
- Features a caching protocol based on “check on use” using Java RMI
- Support Java concurrency management
- An LRU cache with fixed size containing variable sized files
- Emulate C file operations on locally-cached files
- Robustly handle multiple concurrent clients and proxies
- Ensure open-close session semantics on concurrent file access

User Instructions(shell):

1. Set class path:
   ```
   export CLASSPATH=$PWD:$PWD/../lib
   ```

2. Set consistent environment variables(pin15440, proxyport15440) at server, proxy and client:
   e.g.
   ```
   pin15440=xxx
     export pin15440
   ```

3. To compile, run:
   ```
   make
   ```

4. To start the server, run:
   ```
   java Server <port> <rootdir>
   e.g. java Server 1099 ../server
   ```
   * <rootdir> points to a directory tree populated with the initial files and subdirectories to serve.

5. To start proxy, run:
   ```
   java Proxy <serverip> <port> <cachedir> <cachesize>
   e.g. java Proxy 127.0.0.1 1099 ../cache 500000
   ```
   * <cachedir> is a directory the proxy uses to store its cache data; <cachesize> is the size of the cache in bytes

6. To start client, run:
   ```
   LD_PRELOAD=../lib/lib440lib.so ../tools/440read foo
   ```
   * Replace 440read with any library functions :)