MultiChat

A Multicast based chat application that works in a LAN environment. Any user can join multiple courses. A user can send a message to any of his/her registered course and will also receive messages sent to those courses.

Getting Started

Compilation

Use `make` to compile the code. It will generate an executable named `mchat`.

Running the MultiChat

```
$ ./mchat <path to course-data file>
```

For example, if the course-data file is named `course-info.txt` and is stored in the current directory, to run the application,

```
$ ./mchat course-info.txt
```

Optional loopback argument

The `mchat` application also accepts an optional argument to set loopback. (Loopback is *disabled* by default).

For example, if the course-data file named `course-info.txt` is stored in the current directory and you want to allow loopback, run application as,

```
$ ./mchat course-info.txt 1
```

Format for the course-data file

This file contains a list of all courses that can be currently taken along with their ComCode (a 4 digit unique code). This file helps the client to generate group IP address based on the course code. The course-data file should have the following format,

```
<k = Number of Courses>
<C1 COMCODE>,<C1 COURSE CODE>,<C1 COURSE TITLE>
<C2 COMCODE>,<C2 COURSE CODE>,<C2 COURSE TITLE>
..
..
..
<Ck COMCODE>,<Ck COURSE CODE>,<Ck COURSE TITLE>
```

For example,

```
6
1092,CS F213,OBJECT ORIENTED PROG
1090,CS F214,LOGIC IN COMPUTER SC
1093,CS F215,DIGITAL DESIGN
1091,CS F222,DISCR STRUC FOR COMP SCI
1316,CS F301,PRINCIPLES OF PROGG LANG
2266,CS F320,FOUNDATIONS OF DATA SCIENCE
```

(A sample file containing this information for all courses offered in First Semester 2019-20 is included.)

Commands

The program gives these options to a user,

```
1. Send Message
2. Join new course
3. Leave a course
4. Exit
```

Enter the corresponding option number to select an option.

Default settings

These default settings are defined in the first few lines of the code and can be changed from there.

- Maximum message length is set to `2047 bytes`.

- Port for the Multicast socket is set to `6000`.
- BaseIP for multicast groups is set to `238.101`.
- Loopback of Messages is `disabled`.

Working

Generation of multicast group IP Address

All the group IP Addresses are generated based on the ComCode of a course. All IPs have a format of `238.101.1xx.1xx`.

The procedure used to generate an IP address from a course ComCode is,

```
char *genIpFromComCode(int comCode){
    char *multicastIP = allocString(16);
    sprintf(multicastIP,"%s.1%d.1%d",baseIP,comCode/100,comCode%100);
    return multicastIP;
}
```

For example, if a course has ComCode equal to `1092`, then the corresponding group IP will be `238.101.110.192`.

Generation of a Message

A message entered by a user is padded with some extra characters to improve the message readability. An example message is shown below,

```
- - - - - - - - - - - - - - - - - - - -
MESSAGE FROM IS F462
- - - - - - - - - - - - - - - - - - - -
This is a test message.
- - - - - - - - - - - - - - - - - - - -
```

Receiving Messages

A separate child process is run to accept messages from the subscribed groups and print them to console. This process is killed when `4. Exit` option is selected or `SIGINT` signal is given by the user.