

BME-VIK
Linux programozás
Házi feladat

Vizi Gábor
GKPQHH

Specifikáció - Felhasználói útmutató - Fejlesztői útmutató

Házi feladattal kapcsolatos tudnivalók

- A házi feladat leadási határideje az utolsó gyakorlat, illetve az első vizsgaalkalom, ami a pótlási héten van. (Ha valaki csak házi feladatot ad le, akkor nem kell jelentkeznie a vizsgára.) Előzetes egyeztetés alapján korábbi leadásra van lehetőség.
 - A házi feladathoz két dokumentációt is készíteni kell. Egy felhasználóit, amely kliens program esetén a program kezelői felületét írja le, szerver program esetén a rendszer konfigurálását és futtatását. A másik, nagyobb terjedelmű a fejlesztői dokumentáció. Célja, hogy egy másik fejlesztő a dokumentum alapján megértse a program működését. Célszerű egy struktúrális felépítés ábrával kezdeni, majd az egyes részek funkcióit kifejteni. Egyéni protokollt használó hálózatos program esetén le kell írni a protokollt is. Minimális terjedelme 3 oldal. A dokumentációt elegendő elektronikusan leadni.
 - Leadáskor be kell mutatni a programot/rendszert működés közben. Ezt követően átnézzük a dokumentációt és megtekintjük a forráskódot. Várható továbbá néhány fejlesztéssel kapcsolatos kérdés is.
 - A házi feladat és a dokumentációk leadása az aláírás feltétele.
 - Házi feladatot személyesen lehet leadni. A leadás a tanszéki laborban történik. Lehetőség van saját notebookon, vagy távoli hozzáféréssel bemutatni.
 - (A tévhitek eloszlatása végett: nem csak hálózatos programot lehet házi feladatként készíteni.)
-

Specifikáció

Alkalmazás tulajdonságai

- Egy hálózati alkalmazás (TCP/IP)
 - gyakorlatilag: Chatprogram + SSH
 - egyik gépen szerver, másik gépen kliens
 - Lehet üzenetet küldeni és parancsokat kiadni
 - Jelenleg tesztelhető localhost-on
 - Pl. Raspberry Pi B-hez
 - Funkciók
 - menüből választható a funkció
 - chatprogram (üzenetek küldése)
 - színes üzenetek
 - logolás
 - parancskezelés / utasítások elküldése - fogadása és feldolgozása
 - !! bővítés !!
 - Példák parancsokra:
 - local: @
 - remote: #
 - Jelenleg BASH parancsokat lehet beírni
 - Bővítési lehetőségek
 - LED kapcsolgatás
 - billentyűzet LED-ek villogtatása
 - Leállítás - "sudo halt -h"
 - status lekérdezés
 - Működés:
 - menüből kiválasztani, hogy szerver vagy kliens legyen, ezután csatlakozás
 - "server" vagy "client"
 - kilépéshez "exit"
 - Ha chat funkciót választ a felhasználó, akkor
 - Lehet üzenetet küldeni
 - "üzenet"
 - parancsot
 - "@localparancs"
 - "#remotelparancs"
-

Felhasználói dokumentáció

A program elindítása a BASH-ból:

`./LinuxHF`

Ezután előjön a menü:

Menu:

[server] - Szerver

[client] - Kliens

[exit] - Kilépés

A szögletes zárójelben szereplő kifejezést pl. [server] kell begépelni az adott parancs futtatásához. Így "server" a szerver elindításához, "client" a kliens elindításához, "exit" a kilépéshez.

A szerver elindulásakor kiírja az IP címet és vár a csatlakozásra, erről szövegesen is tájékoztat.

A kliens elinduláskor bekéri az IP címet, vagy hosztnevet, amihez csatlakoznia kell, ezt gépeljük be, majd üssünk enter-t. Ekkor megkísérli feloldani a hosztnevet vagy megkeresni az IP címet, majd csatlakozik hozzá.

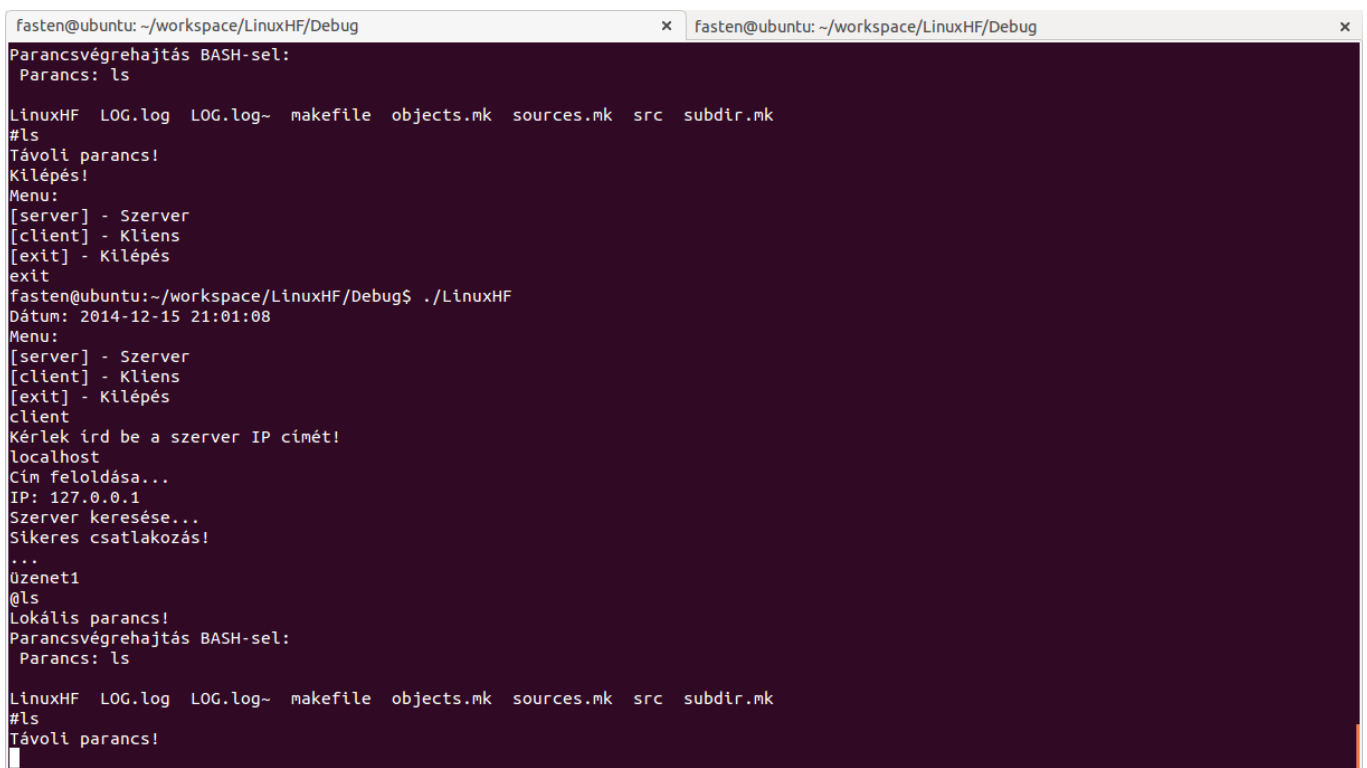
Csatlakozás után mindkét oldalon szöveges üzenet tájékoztat a kapcsolat létrejöttéről.

Ezután üzeneteket lehet küldeni mindkét oldalra. A túloldalról érkező üzenetek pirossal jelennek meg.

Parancs végrehajtás lokális oldalon: `@<command>` szintaktikával történik,

pl. `@ls`

Távoli végrehajtás: `#<command>` szintaktikával.



```
fasten@ubuntu: ~/workspace/LinuxHF/Debug x fasten@ubuntu: ~/workspace/LinuxHF/Debug x
Parancsvégrehajtás BASH-sel:
Parancs: ls

LinuxHF LOG.log LOG.log~ makefile objects.mk sources.mk src subdir.mk
#ls
Távoli parancs!
Kilépés!
Menu:
[server] - Szerver
[client] - Kliens
[exit] - Kilépés
exit
fasten@ubuntu:~/workspace/LinuxHF/Debug$ ./LinuxHF
Dátum: 2014-12-15 21:01:08
Menu:
[server] - Szerver
[client] - Kliens
[exit] - Kilépés
client
Kérlek írd be a szerver IP címét!
localhost
Cím feloldása...
IP: 127.0.0.1
Szerver keresése...
Sikeres csatlakozás!
...
Üzenet1
@ls
Lokális parancs!
Parancsvégrehajtás BASH-sel:
Parancs: ls

LinuxHF LOG.log LOG.log~ makefile objects.mk sources.mk src subdir.mk
#ls
Távoli parancs!
```

Ha a meglévő kapcsolatból ki akarunk lépni, az

`exit`

paranccsal tehetjük.

A program logol is, ezt a `LinuxHF` fájl mellett teszi, a `LOG.log` fájlba.

Tartalmára és logolás mintájára példa:

[...]

2014-12-15 19:58:34 \$program start

2014-12-15 19:58:45 \$chat started

2014-12-15 19:58:55 \$command: ls

2014-12-15 19:59:19 \$chat end

2014-12-15 19:59:38 \$program end

Fejlesztői dokumentáció:

Fordításkor - linkeléskor:

[lpthread](#)

vagyis a pthread library-t linkelni kell!

A fordítás történhet Eclipse-ben is, jelenleg ez történik, de bármilyen gcc fordítóval is lehet.

Jelenleg meglévő fájlok:

[chat.c](#)

[chat.h](#)

[command.c](#)

[command.h](#)

[datetime.c](#)

[datetime.h](#)

[getipaddress.c](#)

[getipaddress.h](#)

[global.c](#)

[global.h](#)

[linprog-hf-vg.c](#)

[log.c](#)

[log.h](#)

[tcpclient.c](#)

[tcpclient.h](#)

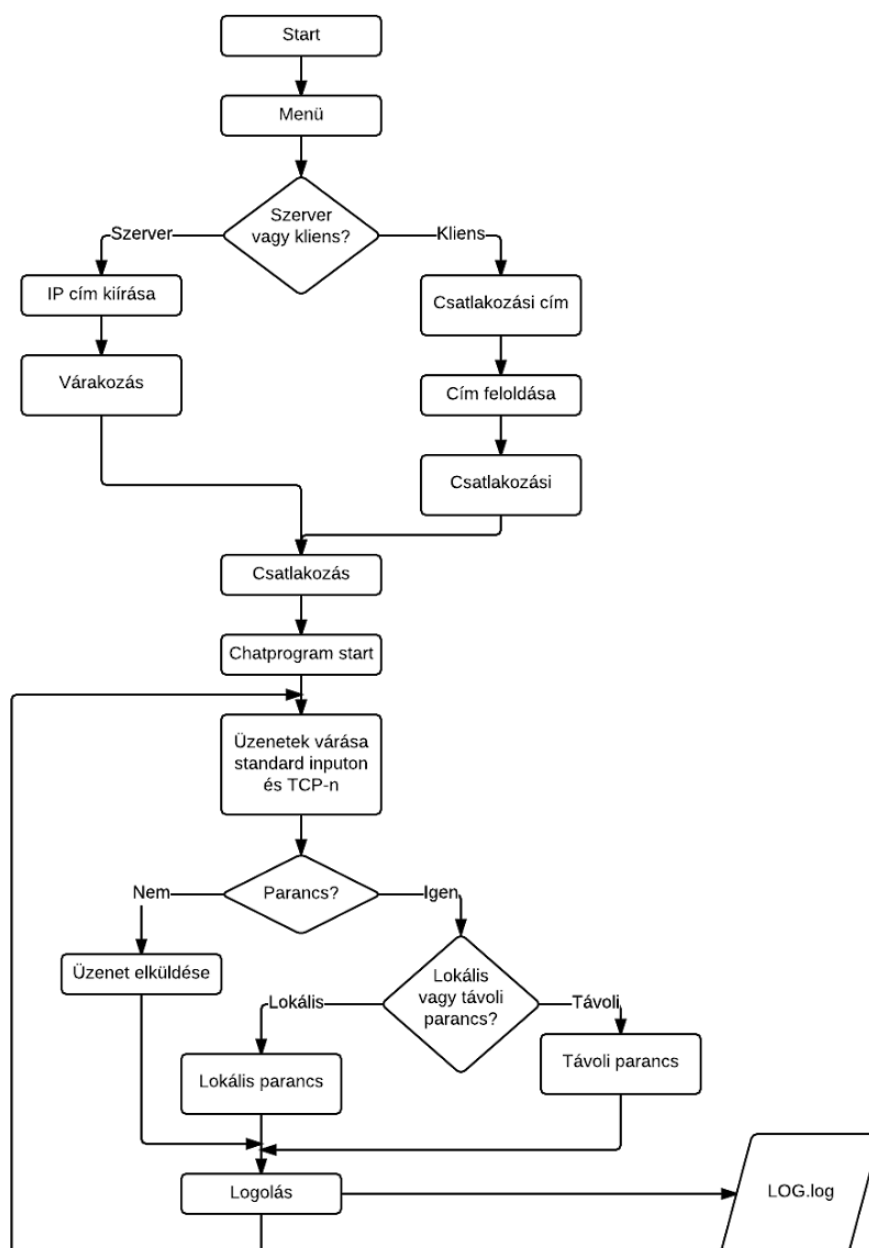
[tcpserver.c](#)

[tcpserver.h](#)

A modulok - fájlok tartalma triviális, a név adja. A main a linprog-hf-vg.c fájlban található, ez a kiinduló pontja a programnak.

linprog-hf-vg	main(), menü
getipaddress	Lekérdezi a hosztnevet és IP címet, majd visszatér a kapcsolódási címmel (IP címmel)
tcpserver	TCP szervert hoz létre, csatlakozásra vár
tcpclient	TCP kliens, csatlakozást kísérel meg
chat	Itt hozunk létre két szálát, egyikben várjuk a TCP-n keresztül érkező üzeneteket, másikban várjuk az STDIN-en (standard input-on) érkező üzeneteket
command	Itt futtatjuk le a parancsokat, ha az alakjuk megfelelő
log	Lementi egy fájlba az eseményeket
datetime	Dátum-időpont páros lementéséhez szükséges
global	Itt lennének a globális változók, de jelenleg nincs használat alatt

Blokkvázlat



A lokális - távoli parancsok futtatása a `system()`; függvénnyel történik, ez átadja a BASH-nek a sztringet, majd megpróbálja lefuttatni. Jelenleg a távoli oldalon lefutott végeredményt nem látni lokális oldalon.

Tesztelt utasítások:

- @clear
- #clear
- @ls
- #ls
- @sudo halt -h
- exit

Fejlesztés menete

getc-vel várja a parancsot

// most már scanf-vel

- szerver
 - írja ki az IP címet
- kliens
- szerver kiírja az IP címet
- kliens bekéri az IP címet

- ipconfig
- kapott üzenetek feldolgozása:
 - @-vel kezdődő parancs (saját)
 - #-cal kezdődő parancs a túloldalt egy linux parancs (pl. ls, ll)
 - ezt define-olni
 - parancsok
 - shutdown
 - status
 - akármilyen
- LOG - file-ba log mindkét oldal
- színes üzenetek, parancsok

- github beüzemelés, vagy csak git

- Eclipse letöltés

<http://eclipse.org/downloads/?osType=linux&release=undefined>

- header fájlba rendezni a modulokat, függvényeket

- Utánnézni, hogy lehet a linuxnak utasítást küldeni ".bat"
 - BASH command
 - system() függvénnyel

- Dokumentáció, felhasználói útmutató, fejlesztői dokumentáció