

1. Egyszerű fájlmegosztás (15 pont)

A standard socket könyvtár felhasználásával készítse el a következő egyszerű fájlservert és a hozzá kapcsolódó kliens alkalmazást.

1.1. Szerver

A szerver a futtató könyvtárban elhelyezett fájlok letöltését teszi lehetővé. Az egyszerűség kedvéért hozzunk létre egy `a.txt` illetve egy `b.txt` nevű állományt, majd a továbbiakban égezzük be a kódba, hogy ez a két fájl létezik és letölthető. A szerver a csatlakozott kliensektől a következő üzeneteket várja:

- **ls** - Elérhető fájlok listájának lekérdezése
- **<fájlnév>** - Adott fájl letöltése (feltehető, hogy a fájl neve 100 karakternél rövidebb)
- **ex** - Kilépés a fájlserverről.
- Tegyük fel, hogy minden üzenet helyes, azaz semmilyen ellenőrzést nem kell végezni.

Az **ls** üzenet hatására a szerver küldjön vissza egy fájlnevek listát. Az üzenet formátuma szabadon megválasztható. A **<fájlnév>** üzenet hatására:

- Ha a fájl létezik, akkor küldje át a fájl méretét bájtokban, majd kezdje el átküldeni a megadott fájl bájtjait. (Részpontoszám kapható, ha beégetett üzenetet használ.)
- Ha a fájl nem létezik, akkor `-1` értéket küldjön vissza, ezzel jelezve a kliensnek, hogy a keresett állomány nem létezik.

Végezetül az **ex** üzenet hatására a szerver bontsa a kapcsolatot a klienssel.

FONTOS, hogy a szerverhez egyszerre több kliens is csatlakozhat, és a kérések kiszolgálása is párhuzamosan történik (egyszerűsítés: a fájl küldése során nem kell másokat kiszolgálni, feltehető, hogy most elég rövid ideig tart ez a lépés!) Ennek a megvalósításához a socket könyvtár `select` utasítását **KELL** használni.

1.2. Kliens

A kliens működése legyen a következő:

- Csatlakozik a szerverhez.
- Küld egy **ls** üzenetet a szervernek.
- Letölti a szerverről az egyik fájlt. (Azt ténylegesen írja a munkakönyvtár megfelelő nevű állományába!)
- Ezt követően 3-szor ismétlje meg az előző két lépést. Minden küldés között várjon 2 mp-et (`sleep(2)`).
- Majd egy **ex** üzenettel lépjen ki a szerverről és bontsa a kapcsolatot.

A program minden lépésnél írja ki az elküldött és a fogadott üzeneteket.

2. Segítség

- Karakterlánc összehasonlítása (fejállomány - `string.h`) `strcmp(első,masodik)-rel`.
- C sztringek esetén, a karakter tömbben szereplő `\0` karakter jelzi a sztring végét! Erre érdemes odafigyelni!
- Fájlkezelés C-ben: <http://www.cprogramming.com/tutorial/cfileio.html> (Bináris fájlkezelés rész)