

# Dokumentáció

A “rendszerkövetelmény” című tárgyhöz készült dokumentáció.

## A használandó fordítóprogram és annak szükséges kapcsolói

- GCC (GNU Compiler Collection) fordító használata
- -o kapcsoló használata a futtatható állomány alias-nak megadásához: (fordítás) gcc chart.c -o test → (futtatás) ./chart
- -fopenmp kapcsoló használata szálkezekés és párhuzamos programozás esetén
- Fontos, hogy a futtatható állomány neve **chart** legyen, enélkül a program nem tud megfelelően működni

## Rendszerkövetelmény

- A program Linux rendszerre (ill. Ubuntu, Debian, stb.... disztr.) lett fejlesztve, más operációs rendszerek alatt hibásan működhet
- Különösebb hardver specifikációt nem igényel

## Felhasználói útmutatás a programhoz

- Lehetséges parancssori argumentumok száma: 1 || 2
- --version: program verziószáma, elkészültének dátuma, fejlesztő neve
- --help: segédlet
- Kétféle üzemmód: -send(küldő), -receive(fogadó) (előbbi az alapértelmezett)
- Kétféle kommunikációs mód: -file, -socket (előbbi az alapértelmezett)
- Kiválasztjuk a kommunikációs módot, majd futtatjuk a programot egyszerre küldőként és fogadóként. Ezt követően az adott kommunikációs csatornán megvalósul az adatátvitel, és legenerálódik egy BMP fájl (chart.bmp)

## A program által visszaadott értékek magyarázata

- Hibakód **0**: ha a program hiba nélkül lefutott, illetve, ha CTRL+C billentyűkombinációval leállásra kényszerítettük a programot (ez esetben egy hibaüzenetet is dob).
- Hibakód **1**: szerver nem válaszolt a megadott időkereten belül
- Hibakód **2**: aljzat (socket) létrehozása sikertelen
- Hibakód **3**: hálózati kommunikációnál küldési hiba
- Hibakód **4**: hálózati kommunikációnál fogadási hiba
- Hibakód **6**: hálózati kommunikációnál a kliens küldött és fogadott értéke különbözik
- Hibakód **7**: hálózati kommunikációnál szerveroldalon bindelési hiba
- Hibakód **8**: fájl kommunikációnál nem található fogadó üzemmódban működő folyamat

## Az elkészített alprogramok rövid leírása

- **int Measurement(int \*\*Values):** random értékek generálása, ahol az első érték 0, majd rákövetkező értékek 1-gyel kisebbek, vagy pedig 1-gyel nagyobbak. Paramétere egy int\* típusú pointer, ami majd az értékekkel feltöltött tömbre helyére mutat a függvény befejeztekor. A függvény visszatér az egész értékek számával.
- **void BMPcreator(int \*Values, int NumValues):** a BMP képfájl létrehozására szolgál. Bemeneti paramétere az ábrázolandó értékeket tartalmazó **tömb kezdőcíme, ábrázolandó egész értékek száma**. Ez az eljárás meghívja még a bin\_to\_dec függvényt, illetve a dec\_to\_bin eljárást. Ezek a nevükben is megtalálható átváltást hivatottak elvégezni. Meghívódik még a fill\_array eljárás is, ami 4 darab tömböt tölt fel egy már létező tömb különböző értékeivel. Mindegyik tömb egy bájtot reprezentál binárisan, amit majd átadva a bin\_to\_decnek, egy decimális értéket kapunk.
- **int FindPID():** a program folyamatainak (chart) pid-jeit keresi (/proc/..), ha talál saját magán kívül másik ilyen néven szereplő folyamatot, akkor annak az azonosítójával tér vissza, egyébként meg 1-gyel.
- **void SendViaFile(int \*Values, int NumValues):** felhasználó alapértelmezett mappájában (/home/username) létrehoz egy Measurment.txt nevű szöveges fájlt, amit soronként feltölt a tömb értékeivel. Values mutató egy egészeket tartalmazó tömb kezdőcímét, míg a NumValues változó fogja tárolni a tömbben lévő egészek darabszámát. Itt meghívódik a FindPID() függvény ami, ha egy valid pid-del tér vissza, akkor küld egy szignált erre a PID-re, egyébként meg terminál a program.
- **void ReceiveViaFile(int sig):** egy szignálkezelő eljárás, olvassa a Measurement.txt-t, majd az olvasott értékekkel feltölt egy tömböt, amely tömb memóriában elfoglalt területe folyamatosan változik. A tömb címével és méretével meghívja a BMPcreator eljárást.
- **void SendViaSocket(int \*Values, int NumValues):** hálózati kommunikáció kliens oldala. Elküldi a NumValues értékét a szervernek, majd aztán a Values pointer által mutatott tömb címén található értékeket. Itt az első paraméterként kapott memóriacím egy tömb kezdőcíme, a második pedig a tömbben lévő egész típusú értékek száma. Folyamatosan ellenőrzi, hogy jó értékeket küld-e vissza a szerver.
- **void ReceiveViaSocket():** hálózati kommunikáció szerver oldala. Folyamatosan fogadja kliens által küldött értékeket, majd nyugtázza azokat. A kapott tömb értékeit eltárolja egy általa létrehozott (ugyanakkor méretű) tömbbe, majd a tömb kezdőcímével és méretével meghívja a BMPcreator eljárást.
- **void SignalHandler(int sig):** egy szignálkezelő eljárás ami SIGINT szignál esetén bezárja a programot, SIGUSR1 esetén hibaüzenetet dob, és SIGALRM esetén a hibaüzenetet követően szintén bezárja a programot.