

TCP通信プログラム

ソケットと呼ばれる通信用ファイル・ディスクリプタを用いる。ファイル・ディスクリプタであるから、手順を踏めば、ファイルへの入出力と同様に通信プログラムを書くことができる。

手順

サーバ側

1. サーバ・ソケットの作成(クライアントからの接続を待ち受けるソケット)
2. クライアントからの要求が来たら、その要求をacceptする。このときサーバ・ソケットとクライアント・ソケットが接続され、クライアント・ソケットを通して通信が可能となる
3. 通信: データ送信 = クライアント・ソケットへのwrite、データ受信 = クライアント・ソケットからのread
4. 通信終了時にクライアント・ソケットをクローズ
5. サーバプログラム終了時にサーバ・ソケットをクローズ

クライアント側

1. クライアント・ソケットの作成、オープン(通信相手のホスト名とポート番号を指定)
2. 通信: データ送信 = クライアント・ソケットへのwrite、データ受信 = クライアント・ソケットからのread
3. 通信終了時にクライアント・ソケットをクローズ

TCPサーバプログラムの例

```
require 'socket'
s = TCPServer.new(12345)
while true
  client = s.accept
  puts Time.now.to_s
  client.print Time.now.to_s
  client.close
end
s.close
```

TCPクライアントプログラムの例

```
require 'socket'

sock = TCPClient.open("localhost", 12345)
puts sock.read
```

sock.close