

ソケット通信プログラミング  
谷口 義樹

1 与えられた課題

ソケット通信プログラミングについて学習する．具体的には簡単なチャットを作成を行う．

ソケットとは，ネットワークを介して他のホストに接続する際に用いられる「接続の端点」である．2つのプログラム間でデータのやりとりをするデータの入出力口として用いられる．ソケットはその使い方にも特徴があり，ソケットを介してデータを送受信するときにはファイルの入出力と同じ要領で行うことができる．つまり，送信したいデータをソケットに書き込むと通信相手のコンピュータのソケットに届く．また，受信はソケットからデータを読み出せばよい．つまり，ファイルの入出力とまったく同じ方法で，異なる2つの端末間でデータのやりとりができる方法が，ソケット通信である．

2 進捗状況

2.1 1対1通信プログラム

ソケット通信がどのような仕組みで行われているかを確認するため，ごく簡単なプログラムを作成し，その動作手順を確認した．なお，データ通信には TCP を用いた．具体的に作成したプログラムは，クライアントがある文字列をサーバーに投げると，サーバーがその文字列を全て大文字にして，クライアントに返すというものである．TCP によるデータ通信手順は Fig. 1 になる．

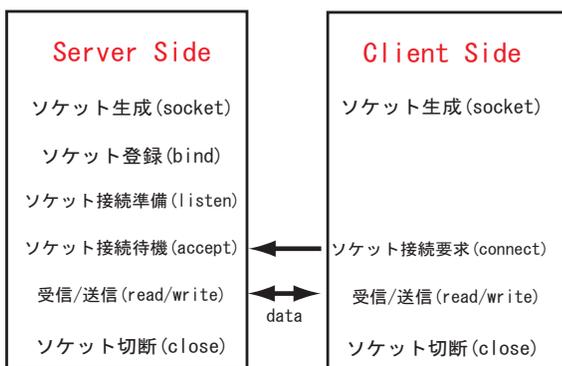


Fig. 1 TCP によるデータ通信手順

2.2 fork を用いたマルチクライアントプログラム

作成したプログラムは，接続を一つ確立すると，それ以降，別のクライアントからの接続要求に対して対応ができないので，このプログラムを複数のクライアントか

らの接続に対応させるための改良を行った．具体的な手法としては，fork を用いた．fork とは，新しいプロセスを作成するための UNIX システムコールである．fork の呼び出しにより，fork を呼び出した時点の親プロセスと等価な子プロセスが作成される．fork の仕組みを Fig. 2 に示す．

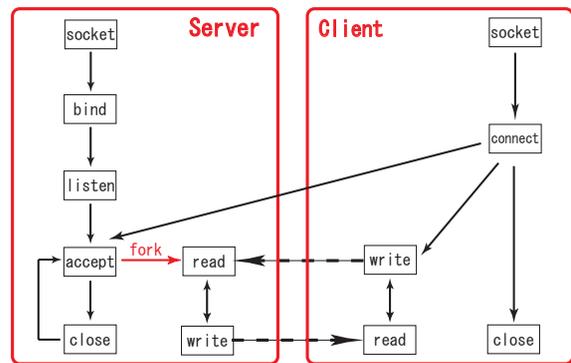


Fig. 2 fork の仕組み

図に示されるように，クライアントからの接続要求が来ると，親プロセスは fork によって子プロセスを作成し，実際の処理を子プロセスに任せて，自分は再び接続待ちの状態になる．この仕組みにより，複数のクライアントからの接続に対応できるプログラムに改良された．

2.3 fork から pthread への移行

しかし，目標であるチャットを作成するために fork を用いると，メモリが共有できないなどの問題からそれぞれの通信で共通の変数が使えないなどの問題が発生し，コードが複雑化することが分かった．そのため，fork の代わりとして POSIX スレッド (pthread) を用いてプログラミングを行うことにした．pthread は同じメモリスペースを共有するため，メモリ内の同じ変数にアクセスすることができ，宣言済みのグローバル変数を読み書きすることができる．

3 翌月への課題

- 現在の fork を用いたプログラムを pthread を用いたものに改良する．
- 目標である pthread を用いたチャットを完成させる．