

# 情報工学実験C ネットワーク実験課題レポート

## —クライアントサーバーモデルで動作する名簿管理プログラムの作成—

氏名: 寺岡久騎 (TERAOKA, Hisaki)

学生番号: 09B22433

出題日: 2024年12月13日

提出日: 2025年 1月21日

締切日: 2025年 1月21日

## 1 プログラムの処理の概要および作成方針

本実験では、ネットワーク通信により動作するクライアントサーバーモデルの TCPによる相互のデータ送受信を行う名簿管理プログラムの作成を行った。本章ではサーバ・クライアントプログラムのそれぞれの機能と、プログラム全体としての処理の概要について解説する。

### 1.1 プログラム全体の処理の流れと概要

プログラム全体の処理の流れとして、まずサーバはクライアントからのTCPの接続要求を待ち受け、クライアントはサーバに接続する。コネクションを確立することで名簿管理の処理が開始し、以降はシステムコール関数を用いたソケット通信により以下の一連の処理を繰り返す。

1. クライアントが名簿管理に関する標準入力からの入力をサーバにメッセージとして送信する
2. サーバがクライアントからのメッセージを受信する
3. サーバがメッセージに対応した名簿データに関する処理を行い、その結果をクライアントにメッセージとして送信する
4. クライアントがサーバから受信したメッセージを標準出力へ出力する

処理の終了時にどちらも接続を切断し、クライアントはプログラムを終了し、サーバは次のクライアントの接続を待ち受ける状態となる。

プログラムで扱う名簿データは「ID, 氏名, 誕生日, 住所, 備考」の項目からなるCSV形式であり、サーバはこの形式のデータを受信するとこれを新たな名簿データとしてメモリに保存する。また、 '%' から始まるメッセージは名簿データに対する様々な処理を実行するためのコマンド入力であり、サーバからクライアントへのメッセージは主にこのコマンド処理の結果である。以下、実装したコマンドに対応した処理や機能、及び処理結果の出力について示す。

■%Q コマンド クライアントプログラムを終了し、サーバプログラムはクライアントとの接続を終了して次のクライアントプログラムの接続を待ち受ける状態となる。

■%C コマンド 登録した名簿データ数の表示を行う。サーバからクライアントへ登録されている名簿データの数と、登録可能な残りのデータ数に関するメッセージを送信する。

■%P コマンド 引数に入力された整数値に応じて登録されている名簿データを表示する。サーバはクライアントへ該当の名簿データ全ての各項目を表示した文字列をメッセージとして送信する。引数の仕様を以下に示す。

- 0の場合：登録順に全件表示
- 負の場合：登録データの後ろから順に絶対値の数だけ表示
- 絶対値が総データ数以上の場合：登録順に全件表示

■%R コマンド サーバ側で引数に指定された名前のファイルを読み込み、CSV形式で記された文字列を名簿データとして登録を行う。

■%W コマンド サーバ側で引数に指定された名前のファイルへ、登録されている全ての名簿データをCSV形式で出力する。

■%S コマンド 引数で指定された整数値に対応する項目に関して、サーバプログラムのメモリに登録されている名簿データのソートを行う。

■%F コマンド 引数に入力された文字列と完全に一致する項目を持つ名簿データ全ての表示を行う。%Pコマンドと同様に、サーバは該当の名簿データの情報をメッセージとしてクライアントへ送信する。

■%D コマンド 引数に入力された整数値に応じて、サーバがメモリに保存している名簿データを削除する機能を持つ。引数の仕様を以下に示す。

- 0の場合：全データ削除
- 正の場合：先頭から引数の数だけ削除
- 負の場合：登録データの後ろから順に引数の絶対値の数だけ表示
- 絶対値が総データ数以上の場合：全データ削除

## 1.2 クライアントプログラムの機能と作成方針

クライアントは処理の始めにサーバとのTCP接続の確立のために、サーバが動作する計算機のホスト名と、サーバプログラムのポート番号が必要である。そのため、これらの情報はプログラム実行時の引数として入力し、これをそれぞれIPアドレス、ポート番号として接続の処理に利用するという設計にした。

サーバとの接続が確立した後は、標準入力からの入力された文字列をサーバへメッセージとして送信し、その後サーバの処理結果を受信して標準出力へ出力するという処理を繰り返す。この繰り返しの実現には、TCPの接続処理の後にループの構文を設け、その中でメッセージの入力と送信、受信と出力の処理を行うという方針を取った。

%Qコマンドによるプログラムの終了は、処理の複雑化を避けるため、上記のサーバとのメッセージの送受信・同期の処理から分岐せず、サーバからのメッセージを受信した後にループから抜けると

いう設計にした。

### 1.3 サーバプログラムの機能と作成方針

サーバプログラムは、まず通信相手となるクライアントからの接続要求を待ち受ける。TCPの  
コネクション確立のためには、

1.3.1 多重通信受付機能 (発展課題2)

1.3.2 サーバログ機能 (発展課題3)

1.3.3 Ctrl-C入力時の安全なプログラムの終了処理 (発展課題3)

1.3.4 プログラム停止時の一時データ記録と復帰処理 (発展課題3)

## 2 プログラムの処理の説明

2.1 サーバプログラム

2.2 クライアントプログラム

## 3 プログラムの使用法と使用例

3.1 他学生のサーバプログラムとの通信

3.2 多重通信受付機能の使用

## 4 プログラムの作成過程に関する考察

## 5 得られた結果に関する考察