

【QUESTION】

1. 解除キーはどこで使用するのでしょうか。

2. バッチのサンプルプログラムを起動させ、クライアントのサンプルプログラムをCALLし、画面より IP:127.0.0.1, PORT:30031で実行を押すサーバー側が以下のメッセージでエラー終了してしまいます。なぜかお分かりになるでしょうか。よろしく願います。

<経緯>

1) 初回は、正常に起動していたのですが、PCの不備のため一度強制終了しています。
2) その後、"アドレスは使用中です"と表示するので、コマンド入力より CALL SKCLOSをしております。それ以降以下のメッセージが出るようになりました。

メッセージID: SOK1001 重大度: 30
メッセージ・タイプ...: 完了
送信日付: 00/04/14 送信時刻: 10:47:15

メッセージ...: 引数に指定された値が正しくない。
原因 - -ソケット関数呼び出しが失敗しました。ERRNO は 3021 です。ERRNO の意味は 引数に指定された値が正しくない。 です。使用した SOKRPG プログラムは SKGIVE です。

3. ソケットディスクリプター(int)は、SKOPEN時には、HEXで見ると、"00000000"を受取ります。(デバックにて確認) SKCLOSもそのままこれを使うかと思います。SKACPTで、ND(パラメータ)で新たなソケットディスクリプターを取得しています。このソケットディスクリプター(int)は、どこから持ってきて、またこの内容と使い方を教えて頂きたいのですが。

4. ポートを2つ使用しての方法です。
RPG版では、"SKBIND"にてポートをパラメータに送っています。

【ANSWER】

プログラムは、SOKRPG開始の時に使用する
SKOPEN
SKUDPO

がユーザーキーの確認を行っています。
ユーザー登録していただいているのは、バグや仕様変更のご連絡を直接メールで案内させていただく場合に備えてと、SOKRPGを使用して開発される場合に、当然いろいろなお問い合わせがあるかと予想し、ユーザー登録させていただいております。
いまのところ、当初、予想したほどお問い合わせや要望はいただいていません。

"アドレスは使用中です"の場合は、
netstatの3番で port 30031を探し、オプション4で削除してください。
PGMをキャンセルなどの場合は、SKCLOSではリセットできないでしょう

"引数に指定された値が正しくない。"は、
SKGIVE/SKTAKEは、データキューを使用してソケットのディスクリプターを渡しています。
恐らく異常終了したときの情報がデータキューに残ってしまっているのだと思います。
SOKRPGライブラリーのDTAQオブジェクトを削除してみてください。

ソケットディスクリプターは、OS/400がJOBごとに割り振ります。
ソケットを作成するたびに、unsigned int型で0から順番に番号が割り当てられます。
ソケットが作成される関数は、SKOPEN(socket関数) SKACPT(accept関数)です。
SKOPENとSKCLOSを連続して発行すると、順番にカウントアップされて行くでしょう。
SKACPTで新たに割りふられるソケットディスクリプターは、接続要求元との通信で使用します。SKSEDN/SKRCVxでその値を引数に与えて下さい。
新たな接続要求を受付けるのは、元々のディスクリプターに成りますので、SKACPTは、元々のディスクリプターを使用します。

【QUESTION】

7-2:また、ノンブロックをするとセレクトをするとのはお決まりでしょうか？

8. サンプルプログラムのSOKRPGBCHD (バッチ1) と SOKRPGBCHP (バッチ2) の終了方法ですが、私は、クライアント (SOKRPGCA) より、

1. "END" を入力し、バッチ2を終了 (クライアントも終了するので再度立ち上げ直し)
2. "EXIT" を入力し、バッチ1を終了させています。正しくは、どのように終了させるべきでしょうか？

9. クライアントの設定の説明を頂きましたが、サーバーサイドでの設定は、ノンブロックの考慮は不要なのでしょうか。

【ANSWER】

ノンブロックにするのは、データを受信していなければ、すぐにプログラムに制御を戻すためです。

たぶんそうなるでしょう。何故かと言うと、セレクトは指定した時間内に複数のportへのデータ受信を待機できるからです。もし、使わない場合は以下になるでしょう。

```
DO
recv port_1
recv port_2
wait ロジック
ENDDO
```

この場合の問題は、

1. 無駄なloopでCPUを浪費すること
 2. waitロジックは自分で実装する必要があること
 3. 順番に処理+waitとなるので、クライアントへの応答が遅くなる (selectの場合は、データ受信と同時にPGMに戻ります。)
- と言う訳で、ノンブロックにした場合は普通の人にはセレクトを使用します。

正しくはと言われると少々困ってしまいますが、取りあえず上記の方法でかまわないと思います。通信中でなければ、ENDJOBしてもかまいません。

非常に難しい質問ですね。

サーバーサイドのプログラム作成ではいくつかのモデルパターンがあります。

たとえば、

1. 一般にデーモンと呼ばれるパターン
接続要求のみを処理し、新たなプロセス (JOB) or スレッドにクライアント通信を任せる。
2. サーバープロセス内でI/Oを行う。
クライアントが一台、または少数の場合
2.のパターンだとタイムアウトの処理が必要になると思われます。

また、ノンブロックが必要かどうかは、サーバー/クライアントかというよりもタイムアウトが必要かどうかの観点で判断したほうがよいと思います。