NAME:
Bankclient & Bankserver

Usage:
Input Parameters:
Client: <Address> <Port>
Server: <Port>
Command Syntax:
create <Account Name>: Can only be used outside of service mode, used to create an account
serve <Account Name>: Puts the session into service mode
deposit <Amount>: adds money to an account
withdraw <Amount>: removes money from an account
end: Can only be used inside a service session ends service mode
quit: Can only be used outside a service session disconnects the client from the server

Output:
Stdout: Correspondence between client and server and server diagnostic data.

To compile:
Make using make file "make server" and "make client" for compiling individuals.

Design Information
Session:
Linked list Struct containing data for a given session. Fields: current account, IP, socketfd, in service, etc. Session contains a reference to the current account and that is used to do all withdraw deposit and query requests.

Account:
Linked list containing data for every account created on the system. Fields: Balance, in service, name. Account is synchronized on all accesses with a universal Account semaphore which locks and unlocks all accounts to prevent printing issues.

Function Documentation:

Client
NAME:
 runCommand – runs a command based on a given command and argument.

SYNOPSIS:

char runCommand(char * command, char * argument, int sockfd)

DESCRIPTION:

Verifies command correctness, transforms it for sending to the server, then sends it.

RETURN VALUE: char command sent to the server.

NAME:

sockReadLoop – constantly reads from the server.

SYNOPSIS

void * sockReadLoop(void * sockfd)

DESCRIPTION:

Constantly reads + prints on given socket. Used for threading. Terminates the program if connection to the server is lost.

RETURN VALUE:

void * NULL

NAME:

isOnlyDouble– checks if string is a double

SYNOPSIS

int * isOnlyDouble(char * str)

DESCRIPTION:

Makes sure the input is a double.

RETURN VALUE:

Int isDouble

Server

NAME:

runCommand– Takes the raw string input from the client and processes it, routing it to the correct command

SYNOPSIS

int runCommand(char* input, Session * session)

DESCRIPTION:

Routes string command to the proper command function.

RETURN VALUE:

Int errorCode

NAME:

createCommand, serveCommand, depositCommand, withdrawCommand, endCommand, quitCommand– Takes the raw string input from the client and the current session data and manipulates current session/account to manage the bank

SYNOPSIS

int ___Command(char* input, Session * session)

DESCRIPTION:

Does the operations involved to do a given command.

RETURN VALUE:

Int errorCode

NAME:

printError– Sends an error corresponding to a code to the client for printing.

SYNOPSIS

void printError(int code, Session * session)

DESCRIPTION:

Given specific command codes, write() is used to send an error to the client for an illegal operation.

RETURN VALUE:

Int errorCode