HACETTEPE UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING BBM203

Name : Mehmet Taha

Surname : Usta

Number : 21527472

E~mail : b21527472@cs.hacettepe.edu.tr

Subject : Stack, Queue, and Dynamic Memory Allocation

Programming Language : C

1. Problem Definition

Design a simple version of network communication between peers within a network.

2. Methods and Solution

Firstly, the command line arguments are stored with variables. Then opens the input files. According to the user input, Clients are created using the malloc function. Then the clients.dat file starts to read with the fscanf function. The elements of the clients array are edited when reading the lines. Then the routing.dat file starts to read with the fscanf function. The elements of the clients routing table array are edited when reading the lines. Finally commands.dat entry is read by the fgets function. NewLine characters are deleted when reading input file. Command lines are splited by space. Then directed by the first word. İf the word never matches, the result is "Invalid command".İf the word matches “MESSAGE”, MESSAGE command will accomplish the preparation of the message for transmission. İf the word matches “SHOW_FRAME_INFO”, SHOW_FRAME_INFO command will display stated Frame info. İf the word matches “SHOW_Q_INFO”, SHOW_Q_INFO command will display current total number of frames. İf the word matches “SEND”, SEND command will cause the message that is currently placed on the sender’s outgoing queue to be sent to it’s intended receiver. İf the word matches “PRINT_LOG”, PRINT_LOG prints the log of stated Client upon the successful completion of communicating the message between sender client and receiver client. After the problem is solved, elements in the clients array are released using the free () function in the for loop. Finally, the fclose () function closes the inputs, and the program ends.

3. Functions Implemented and not Implemented

Strtok() = Breaks string str into a series of tokens using the delimiter delim.

Fopen() = Function is used to open a file to perform operations such as reading, writing etc.

Atoi() = Converts the string argument str to an integer (type int).

Malloc()=Allocates the requested memory and returns a pointer to it.

While() = loop in C programming repeatedly executes a target statement as long as a given condition is true.

Fgets() = Function is used to read a file line by line.

Strlen() = Computes the length of the string str up to, but not including the terminating null character.

Strcmp() = The strcmp() function takes two strings and return an integer.

Strcpy() = Copies the string pointed to, by src to dest.

For() = A for loop is a repetition control structure that allows you to efficiently write a loop that needs to execute a specific number of times.

printf() = Function is used to print the “character, string, float, integer, octal and hexadecimal values” onto the output screen.

Free() = Deallocates the memory previously allocated by a call to calloc, malloc, or realloc.

Fclose() = Function closes the file that is being pointed by file pointer fp

Typedef struct{} = Defines a physically grouped list of variables to be placed under one name in a block of memory, allowing the different

variables to be accessed via a single pointer, or the struct declared name which returns the same address

Time_stamp() = Contains time and date function declarations to provide standardized access to time/date manipulation and formatting.

createStack()=Function to create a stack. It initializes size of stack as 0

createQueue()=Function to create a queue of given capacity. It initializes size of queue as 0

createLogs() = Function to create a Logs of given capacity.

createClients() = Function to reads files, creates clients, allocates memory. Then fills clientID, clientIpAddress, ClientMacAddress, routing table.

isEmptyQueue () = Queue is empty when size is 0.

isFullQueue () = Queue is full when rear becomes equal to the capacity

enqueue () = Function to add an frame* item to the queue.It changes rear

enqueue2 () = Function to add an Frame item to the queue.It changes rear

dequeue () = Function to remove an item from queue.It changes front

Allqueue() = Function to get all queue

isFullStack() = Stack is full when top is equal to the last index

isEmptyStack () = Stack is empty when top is equal to -1

push() = Function to add an item to stack. It increases top by 1

findIdNumber () = function to finds client id

substring () = C substring function definition

findneighbor() = function to finds neighbor id