

COMP 7005

Assignment 1

Design

Winston Nguyen
A01297231
Jan 26th, 2024

Pseudocode	5
check_args	5
Parameters	5
Return	5
Pseudo Code	5
create_socket	5
Parameters	5
Return	5
Pseudo Code	5
connect_client	6
Parameters	6
Return	6
Code	6
send_message	6
Parameters	6
Return	6
Pseudo Code	6
receieve_response	6
Parameters	6
Return	7
Pseudo Code	7
close_socket_client	7
Parameters	7
Return	7
Code	7
read_file	7
Parameters	7
Return	7
Pseudo Code	7
replace_new_lines	8
Parameters	8
Return	8
Pseudo Code	8
bind_socket	8
Parameters	8
Return	8
Code	8
remove_existing_socket	8
Parameters	8
Return	9

Pseudo Code	9
listen_connections	9
Parameters	9
Return	9
Pseudo Code	9
accept_connection	9
Parameters	9
Return	9
Pseudo Code	9
check_data	10
Parameters	10
Return	10
Pseudo Code	10
send_response	10
Parameters	10
Return	10
Pseudo Code	10
close_socket_server	11
Parameters	11
Return	11
Pseudo Code	11
get_words	11
Parameters	11
Return	11
Pseudo Code	11
remove_whitespace	12
Parameters	12
Return	12
Pseudo Code	12
get_word_count	12
Parameters	12
Return	12
Pseudo Code	12
get_char_count	13
Parameters	13
Return	13
Pseudo Code	13
get_char_freq	13
Parameters	13
Return	13
Pseudo Code	13

sort_dict	14
Parameters	14
Return	14
Pseudo Code	14
format_response	14
Parameters	14
Return	14
Pseudo Code	14

Pseudocode

check_args

Parameters

Parameter	Description
args	The command line arguments passed to main

Return

- Nothing

Pseudo Code

```
If length of args is not a valid length (2):  
    Throw error  
If passed in argument is not a valid file ext (.txt):  
    Throw error
```

create_socket

Parameters

Parameter	Description

Return

- Nothing

Pseudo Code

```
Create domain socket  
If failure to create socket:  
    Throw error
```

connect_client

Parameters

Parameter	Description
path	The path the client will connect to

Return

- Nothing

Code

```
Connect client to socket using socket_path
If failure to connect to path:
    Throw error
```

send_message

Parameters

Parameter	Description
words	A string of data to send to the server

Return

- Nothing

Pseudo Code

```
Encode words to send
Send encoded words via the domain socket
If failure to send words:
    Throw error
```

receieve_response

Parameters

Parameter	Description

Return

- Nothing

Pseudo Code

```
Wait for response from server
Print response
If failure to receive response:
    Throw error
```

close_socket_client

Parameters

Parameter	Description
client	The client connection to be closed.

Return

- Nothing

Code

```
Close the client
If failure to close the client
    Throw error
```

read_file

Parameters

Parameter	Description
file_name	The name of the file to read.

Return

- String: words read inside the given file_name

Pseudo Code

```
Open the file using file_name as a read
Read the content
Call replace_new_lines
```

replace_new_lines

Parameters

Parameter	Description
text_data	The string contains words from a text file.

Return

- String: text_data with the new lines replaced.

Pseudo Code

Replace all newlines inside of text_data with spaces

If failure to replace lines:

Throw error

bind_socket

Parameters

Parameter	Description
path	The path to bind the socket to.

Return

- Nothing

Code

Bind the server to the socket path name

If failure to bind:

Throw error

remove_existing_socket

Parameters

Parameter	Description
path	The path of the socket to remove.

Return

- Nothing

Pseudo Code

```
Remove any existing connections to the socket path
If failure to remove path:
    Throw error
```

listen_connections

Parameters

Parameter	Description

Return

- Nothing

Pseudo Code

```
Listen for incoming connections
Wait until a connection is found
If failure to connect to a connection:
    Throw error
```

accept_connection

Parameters

Parameter	Description

Return

- Bytes: encoded data from connection

Pseudo Code

```
Accept a found connection
```

```
Receive the connections data
If failure to receive data:
    Throw error
```

check_data

Parameters

Parameter	Description
data	The encoded data received from the connection.

Return

- String: decoded data.

Pseudo Code

```
If no data exists:
    Throw error
Decode the data
If failure to decode:
    Throw error
Else:
    Return decoded data
```

send_response

Parameters

Parameter	Description
response	The formatted response to send to the client/accepted connection.

Return

- Nothing

Pseudo Code

```
Encode the response
Send the encoded response through the connection.
```

```
If failure to send response:  
    Throw error
```

close_socket_server

Parameters

Parameter	Description
connection	The connection to close
socket_path	The path connection to remove.

Return

- Nothing

Pseudo Code

```
Close the connection  
If failure to close:  
    Throw error
```

```
Remove the connection to the socket_path  
If failure to remove:  
    Throw error
```

get_words

Parameters

Parameter	Description
word_string	The string of words received from the client/connection.

Return

- Array: list of words

Pseudo Code

```
Call remove_whitespace  
Separate words inside of word_string by spaces
```

```
Remove empty strings in the list  
Return list
```

remove_whitespace

Parameters

Parameter	Description
word_string	A string of words.

Return

- String: with extra whitespaces removed

Pseudo Code

```
Replace consecutive whitespaces with a single space.  
If failure to remove whitespaces:  
    Throw error
```

get_word_count

Parameters

Parameter	Description
words	An array of strings.

Return

- Int: the number of words in the list

Pseudo Code

```
Get the length of words  
Return
```

get_char_count

Parameters

Parameter	Description
words	An array of strings.

Return

- Int: the number of characters total.

Pseudo Code

```
Set count to 0
For each string in words:
    Increase count by the length of the string
Return count
```

get_char_freq

Parameters

Parameter	Description
words	An array of strings.

Return

- Dictionary: holding the number of times a character is found

Pseudo Code

```
Create a dictionary

// List comprehension in python
Set each string to lowercase in words

Join the words to a single string
For each char in words:
    If the char does not exists in the dictionary:
        Set dictionary[char] = 0
    Else:
        Increase dictionary[char] by one
```

Return dictionary

sort_dict

Parameters

Parameter	Description
char_freq	A dictionary of character frequencies.

Return

- Dictionary: sorted by keys

Pseudo Code

```
Get the keys inside the dictionary
Sort dictionary in ascending order using keys
Return sorted dictionary
```

format_response

Parameters

Parameter	Description
word_count	The number of words in the connection data string.
char_count	The number of characters in the connection's data string.
char_freq	Each character's frequencies found in the connection's data string.

Return

- String: a formatted response.

Pseudo Code

```
Format word_count, char_count, char_freq into a single string
Return formatted string
```