The task of this program is to copy all contents of an existing file into new files that

are created. So we have 2 inputs for this code: first input which is the input file

name which reads the content of the file, second input is the output file which we

creating and writing into.

First step in this code: we prompt the user to enter input filename which calls a

system call printf to be shown on the screen and the another system call is scanf

which restore the user input.

Second step: we have to check if the files exist or not. In input file, we want it to be

exist so we turn the read permission ON, to open the file we use the fopen()

function with including the read permission “r”. Using the IF statement to if input

file returned from fopen is a NULL then there was an error, otherwise successful

file exists.

Third step: we do the same as the previous step except the output file shouldn’t

exist. If it exist we exit the program. If didn’t exist so we call the open system call

with write permission for the output file.

Forth step: there’s a loop to read from input file & write to output file until the read

operation fails (fread returns -1).while reading (200 characters )the input file we

save it to a buffer memory using the fgets() function .When fgets returns NULL it’s

an error or at the EOF(end of file).

Fifth step: both file descriptors are closed for reuse when the loop is exited.

The task of this program is to copy all contents of an existing file into new files that

are created. So we have 2 inputs for this code: first input which is the input file

name which reads the content of the file, second input is the output file which we

creating and writing into.

First step in this code: we prompt the user to enter input filename which calls a

system call printf to be shown on the screen and the another system call is scanf

which restore the user input.

Second step: we have to check if the files exist or not. In input file, we want it to be

exist so we turn the read permission ON, to open the file we use the fopen()

function with including the read permission “r”. Using the IF statement to if input

file returned from fopen is a NULL then there was an error, otherwise successful

file exists.

Third step: we do the same as the previous step except the output file shouldn’t

exist. If it exist we exit the program. If didn’t exist so we call the open system call

with write permission for the output file.

Forth step: there’s a loop to read from input file & write to output file until the read

operation fails (fread returns -1).while reading (200 characters )the input file we

save it to a buffer memory using the fgets() function .When fgets returns NULL it’s

an error or at the EOF(end of file).

Fifth step: both file descriptors are closed for reuse when the loop is exited.

The task of this program is to copy all contents of an existing file into new files that

are created. So we have 2 inputs for this code: first input which is the input file

name which reads the content of the file, second input is the output file which we

creating and writing into.

First step in this code: we prompt the user to enter input filename which calls a

system call printf to be shown on the screen and the another system call is scanf

which restore the user input.

Second step: we have to check if the files exist or not. In input file, we want it to be

exist so we turn the read permission ON, to open the file we use the fopen()

function with including the read permission “r”. Using the IF statement to if input

file returned from fopen is a NULL then there was an error, otherwise successful

file exists.

Third step: we do the same as the previous step except the output file shouldn’t

exist. If it exist we exit the program. If didn’t exist so we call the open system call

with write permission for the output file.

Forth step: there’s a loop to read from input file & write to output file until the read

operation fails (fread returns -1).while reading (200 characters )the input file we

save it to a buffer memory using the fgets() function .When fgets returns NULL it’s

an error or at the EOF(end of file).

Fifth step: both file descriptors are closed for reuse when the loop is exited.

The task of this program is to copy all contents of an existing file into new files that are created. So we have 2 inputs for this code: first input which is the input file name which reads the content of the file, second input is the output file which we creating and writing into.

Initial phase in this code: we brief the client to enter input filename which calls a system call printf to be appeared on the screen and the another system call is scanf which restore the client input.

Second step: we need to check if the documents exist or not. In info record, we need it to be exist so we turn the read authorization ON, to open the document we utilize the fopen() work with including the read consent "r". Utilizing the IF explanation to whenever input document got back from fopen is a NULL then there was a blunder, in any case fruitful record exists.

Third step: we do likewise as the past advance aside from the yield document shouldn't exist. On the off chance that it exist we leave the program. On the off chance that didn't exist so we call the open framework call with compose consent for the yield records.

Forth step: there’s a loop to read from input file & write to output file until the read operation fails (fread returns -1).while reading the input file we save it to a buffer memory using the fgets() function .When fgets returns NULL it’s an error or at the EOF(end of file).

Fifth step: both file descriptors are closed for reuse when the loop is exited.

🡺Also I have attached script file showing everything, also provided photos in photos folder in Zip file.

🡺I have also attached some screenshots of my executing and stracing of the file.

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated