* Pseudocode

START

Declare Headers

Declare Function Prototypes such as void copier(), void scanner()

Main Function

char choice [30]

Print “\n\n===============================================\n\n”

Print “\n\TCOPIER\n”  
Print “\n\tSCANNER\n”  
Print “\n\tEXIT\n”

Print “\n\tEnter your choice : “

Get choice

if (choice == “COPIER”) then

Using copier() to go to function copier

else if (choice == “SCANNER”) then

Using scanner() to go to function scanner

else if (choice == “EXIT”) then

display “\tThanks for using this program. \n\n”

else

Display “\tYou’ve made a mistake, Try again… \n\n\n

Using main () to go back main function.

Endif, endif, endif, end

Copier Function

int choice

Print “=================COPIER=================”

Print \n\t#1 TO BE ROUTE OUT / FILE\n.

Print \t#2 NO NEED TO PHOTOCOPY.

get choice.

switch(choice)

case 1:

int pagequantity, starter

char size [10], type [15], option [20]

Print "\n\tPage Quantity: "

Get and read pagequantity FROM stdin

Print "\n\t====TYPE====\n"

Print "\tBlack & White\n"

Print "\tColored\n"

Print "\tType : "

Using getchar () to read and discard the newline character left by the previous input

Get and read type from stdin

Print "\t====SIZE====\n"

Print "\tA4\n"

Print "\t8 x 13\n"

Print "\tSize : "

Using getchar () to read and discard the newline character left by the previous input

Get and read papersize FROM stdin

Print “\t====PRINTING SETTING====\n"

Print “\t1 sided-2 sided: T to T”

Print “\n\t2 sided-2 sided”

Print “\n\t1 sided-comb 4 orig”

Print “\n\t1 sided-com 4 orig”

Print “\n\tCreate Margin”

Print “\n\tNone”

Print “\n\tPrinting Setting :”

Using getchar() to read and discard the newline character left by the previous input

Get and read option from stdin

starter = 1

copystarter = 1

If (copyquantity > 1) then

While(copystarter <= copyquantity)then

Print \t====Document Sorted====\n

WHILE (starter <= pagequantity) DO

Print “\tDone Photocopied Page # \n, starter”

Increment starter

ENDLOOP

Increment copystarter

ENDLOOP

Print “\n\n \t====ALL DOCUMENT PAGES ARE PHOTOCOPIED! ====\n\n”

Using main() to go back to main function.

else if (copyquantity == 2) then

WHILE (starter <= pagequantity) DO

Print \t Done Photocopying Page # \n, starter.

Increment starter

ENDLOOP

Else

Print ("\tYou've made a mistake , Try again..\n\n\n");

Using main () to go back main function.

Endif, endif

END

Case 2:

Display “\t GIVE DIRECTLY TO RELEASING OFFICER OR PUT IT TO FILLING FOLDER\n\n\n”

Default:

Print “\t You’ve made a mistake, Try again...\n\n\n”

Using main () to go back to main function

END FUNCTION

Scanner Function

Print”\n\tInsert the USB DRIVE”

Declare fp as FILE

Open "scannedfile.txt" for appending and set fp as the file pointer

Int unscan, pagequantity

Char file\_name[40], type[40], papersize[40]

Print "=================SCANNER================="

Print "\n\tFile Name : "

Using getchar() to read and discard the newline character left by the previous input

Get file\_name and read file\_name FROM stdin

Print to fp "File Name : " + file\_name

Print "\n\tPage Quantity: "

Get and read pagequantity FROM stdin

Print to fp "\tPage Quantity : " + pagequantity

Print "\n\t====TYPE====\n"

Print "\tBlack & White\n"

Print "\tColored\n"

Print "\tType : "

Using getchar () to read and discard the newline character left by the previous input

Get type and read type from stdin

Print to fp "\n\tType : " + type

Print "\t====SIZE====\n"

Print "\tA4\n"

Print "\t8 x 13\n"

Print "\tSize : "

Using getchar () to read and discard the newline character left by the previous input

Get papersize and read papersize FROM stdin

Print to fp "\tSize : " + papersize

unscan = 1

WHILE (unscan <= pagequantity) DO

Print "\tDone Scanning Page # " + unscan + "\n"

Increment unscan BY 1

ENDLOOP

Print "\n\n \t====Scanning is Done! ====\n\n"

Close fp

Using main () to go back to main menu

END FUNCTION