

**ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - СОФИЯ**  
Факултет Компютърни системи и технологии

**КУРСОВА РАБОТА**

**по**

**ПИК 1**

Изготвил: Борислав Бориславов Дата: 11.01.2021  
ФН. 121220089  
Гр. 47, КСИ

Курсова работа по дисциплина ПИК1 за студенти ФКСТ

Напишете програма, която симулира работата на принтер. Вашият принтер има 4 глави за печатане пълни с тонер, като всяка е с различен цвят и с капацитет 200 грама. При печатане на лист се взимат в предвид няколко критерия – гладкост на хартията, цветове на печат и наситеност на печата.

Хартията може да има 3 вида гладкости – гладка, обикновена и груба, като съответно гладката консумира 10% по – малко мастило от обикновената, а грубата 15% повече. Наситеността на печата се определя в процентно отношение между 70 и 120 процента и се въвежда от потреби теля.

Ако принтирате черно бяло то за обикновен лист е нужно 1 грам мастило, а ако има снимка количеството се повишава на 1.35 грама.

При цветното принтиране трябва да използвате комбинации от цветове, като потребителя въвежда колко е пълна страницата с процентно отношение за всеки цвят(общо 100 % за всички цветове) и кои цветове има за принтиране. За принтиране на 1 цвят на цяла страница отново се разходва по 1 грам мастило, като трябва да се вземат горните правила за крайното количество разходвано мастило.

Въвежда се команда Print последвана от типа на принтиране – цветно или черно бяло след което при цветно се записват броя цветове с които се принтира. Програмата подава запитвания за останалите елементи нужни за принтиране.

При недостиг на мастило на някоя глава програмата да изведе съобщение ERROR Cartrage need replacement.

Програмата продължава след команда REFILL с което се сменя главата с нова.

При успешно принтиране да се изведе PAGE PRINTED

При команда EXIT принтера се гаси и спира работата си.

Студент:……………………                                     асистент:…………………

Описание:

Програмата започва с деклариране на 2 променливи – programFlag и heads[4] . Следва while цикъл, който се върти през цялото време, докато потребителят не въведе команда EXIT. Тялото му започва с декларирането на променливи – input, blackAndWhiteFlag, inkCostColors[]. След тях се осъществява проверка за наличието на тонер в главите на принтeра, при недостик на конзолата се изписва съобщение CARTRIDGE NEEDS REPLACEMENT , след което потребителят въвежда команда REFILL и главата се сменя с нова, действието се развива във for цикъл, 4 пъти – 1 проверка за всяка глава. Следва втори while цикъл, който се върти докато не се избере тип на принтиране – черно-бял или цветен. При избор на черно-бял програмата поддава запитване за тип на листа - нормален или снимка. След което разхода за мастило се променя, в зависимост от типа на листа. При избор на цветно принтиране, потребителят въвежда процентно съотношение на цветовете червен, зелен и син, като за всеки цвят отново се променя разхода на мастило, в зависимост от данните, въведени от потребителя. След успешното задаване за цвят и тип, програмата подава запитване за гладкост на хартията – финна, нормална или груба, като всяка има различен разход мастило. При избор на финна, разхода намалява с 10%, а при груба се увеличава с 15%. След успешното задаване на гладкост, следва и последната заявка от програмата – наситеност. Тя се въвежда в проценти в диапазон от 70 до 120 процента. След като потребителят въведе процентите, програмата изчислява крайният разход на мастило, за зададените цветове и при успешно „принтиране“ се изписва на конзолата PAGE PRINTED. Следва кратка информация, траеща 5 секунди, която показва количеството на всяка една глава с мастило. Програмата започва отначало с проверката за наличие на мастило.

Функция TurnOff() – изпозва се, когато е въведена комадната EXIT, представлпва кратка „анимавия“ с текст Turning printer off, след което програмата спира.

Код:

#include <stdio.h>

#include <windows.h>

#include <conio.h>

int TurnOff()

{

system("cls");

printf("\n =============================\n | Turning printer off. |\n | |\n =============================\n");

Sleep(500);

system("cls");

printf("\n =============================\n | Turning printer off.. |\n | |\n =============================\n");

Sleep(500);

system("cls");

printf("\n =============================\n | Turning printer off... |\n | |\n =============================\n");

Sleep(500);

system("cls");

printf("\n =============================\n | |\n | |\n =============================\n");

return 0; //program ends

}

int main()

{

int programFlag = 0;

double heads[4] = { 200, 200, 200, 200}; //red, green, blue, black

while (programFlag != 1)

{

int input = 0; //used in every scanf\_s statement

int blackAndWhiteFlag = 0;

double inkCostColors[4] = { 1, 1, 1, 1 }; // temp array for the colors, 0Red, 1Green, 2Blue, 3Black

for (int i = 0; i < 4; i++)

{

if (heads[i] <= 0)

{

system("cls");

printf("\n =============================\n | ERROR |\n | Cartrage need replacement |\n =============================\n");

Sleep(2000);

system("cls");

printf("\n =============================\n | Choose an option |\n | 1.Refill 2.EXIT |\n =============================\n");

int value = 0;

scanf\_s("%d", &value);

if (value == 1)

{

heads[i] = 200;

system("cls");

printf("\n =============================\n | Cartrage replaced! |\n | |\n =============================\n");

Sleep(1500);

}

else if (value == 2)

{

TurnOff();

return 0;

}

else

{

system("cls");

printf("\n =============================\n | INVALID VALUE |\n | 1.Refill 2.EXIT |\n =============================\n");

}

}

}

system("cls");

printf("\n =============================\n | Choose an option |\n | 1.Print 2.EXIT |\n =============================\n");

int loopFlag = 0;

while (loopFlag != 1)

{

printf(" ----> ");

scanf\_s("%d", &input); // choose between print or exit

if (input == 1) // print

{

loopFlag = 1; // if choose print - flag is 1

system("cls");

printf("\n =============================\n | Choose print type |\n | 1.Black&White 2.Colored |\n =============================\n ----> ");

int flagColor = 0;

while (flagColor != 1)

{

scanf\_s("%d", &input); //choose blackWhite or colored

if (input == 1) //black and white

{

blackAndWhiteFlag = 1;

flagColor = 1;

system("cls");

printf("\n =============================\n | Choose print type |\n | 1.Normal 2.Image |\n =============================\n");

int flag = 0;

while (flag != 1)

{

printf(" ----> ");

scanf\_s("%d", &input); // type normal or image

if (input == 1) //normal

{

//ink value here is the default value (1)

system("cls");

printf("\n =============================\n | TYPE CHOSEN |\n | SUCCESSFULLY |\n =============================\n ----> \n");

Sleep(1500);

flag = 1;

}

else if (input == 2) //image

{

system("cls");

printf("\n =============================\n | TYPE CHOSEN |\n | SUCCESSFULLY |\n =============================\n ----> \n");

Sleep(1500);

inkCostColors[3] = 1.35; //set the black color ink cost to 1.35

flag = 1;

}

else

{

system("cls");

printf("\n =============================\n | INVALID VALUE |\n | 1.Normal 2.Image |\n =============================\n");

Sleep(1500);

system("cls");

printf("\n =============================\n | Choose print type |\n | 1.Normal 2.Image |\n =============================\n");

continue;

}

}

}

else if (input == 2) //colored

{

//programFlag = 1;

flagColor = 1;

system("cls");

printf("\n =============================\n | Enter colors percentage |\n | 1.Red 2.Green 3.Blue |\n =============================\n");

int percentage = 100; //percent that indicates the list/paper size, 100 percent means that is full

for (int i = 0; i < 3; i++) // loops 3 times for each color

{

if (percentage <= 0) // check if the percent has reached 0 or less, if yes - breaks the loop

{

break;

}

printf(" ----> %d. ", i + 1);

scanf\_s("%d", &input); // enter current color percent

if (input > percentage || input < 0) // check percent value if correct

{

system("cls");

printf("\n =============================\n | INVALID AMOUNT |\n | 1.Red 2.Green 3.Blue |\n =============================\n");

Sleep(1500);

system("cls");

printf("\n =============================\n | Enter colors percentage |\n | 1.Red 2.Green 3.Blue |\n =============================\n");

i--;

continue;

}

inkCostColors[i] = input \* 0.01; // adds the value to the specific color in the temp array

percentage -= input; // update the percentage left

system("cls"); // . .

printf("\n =============================\n | Percentage left - %d |\n | 1.Red 2.Green 3.Blue |\n =============================\n", percentage);

}

system("cls");

printf("\n =============================\n | COLORS CHOSEN |\n | SUCCESSFULLY |\n =============================\n ----> \n");

Sleep(1500);

}

else

{

system("cls");

printf("\n =============================\n | INVALID VALUE |\n | 1.Black&White 2.Colored |\n =============================\n");

Sleep(1500);

system("cls");

printf("\n =============================\n | Choose print type |\n | 1.Black&White 2.Colored |\n =============================\n ----> ");

continue;

}

}

}

else if (input == 2) // exit

{

TurnOff();

return 0;

}

else //invalid input

{

system("cls");

printf("\n =============================\n | INVALID VALUE |\n | 1.Print 2.EXIT |\n =============================\n");

Sleep(1500);

system("cls");

printf("\n =============================\n | Choose an option |\n | 1.Print 2.EXIT |\n =============================\n");

continue;

}

}

system("cls");

printf("\n =============================\n | CHOOSE SMOOTHNESS |\n | 1.Smooth 2.Normal 3.Rough |\n =============================\n");

int flag = 0; //flag that indicates when smoothness is chosen

while (flag != 1)

{

printf(" ----> ");

scanf\_s("%d", &input); // choose between smooth, normal or rough

if (input == 1)

{

// decreases ink costs with 10 percent, because its smooth

for (int i = 0; i < 4; i++)

{

if (blackAndWhiteFlag == 0 && i < 3)

{

inkCostColors[i] \*= 0.9;

heads[i] -= inkCostColors[i];

}

else if (blackAndWhiteFlag == 1 && i == 3)

{

inkCostColors[i] \*= 0.9;

heads[i] -= inkCostColors[i];

}

}

flag = 1; //indicator that a smoothness is now chosen

system("cls");

printf("\n =============================\n | CHOSEN |\n | SMOOTH |\n =============================\n ----> ");

Sleep(1500);

}

else if (input == 2)

{

flag = 1; //indicator that a smoothness is now chosen

system("cls");

printf("\n =============================\n | CHOSEN |\n | NORMAL |\n =============================\n ----> ");

Sleep(1500);

}

else if (input == 3)

{

//ink costs increase with 15 percent, because its rough

for (int i = 0; i < 4; i++)

{

if (blackAndWhiteFlag == 0 && i < 3)

{

inkCostColors[i] \*= 1.15;

heads[i] -= inkCostColors[i];

}

else if (blackAndWhiteFlag == 1 && i == 3)

{

inkCostColors[i] \*= 1.15;

heads[i] -= inkCostColors[i];

}

}

flag = 1; //indicator that a smoothness is now chosen

system("cls");

printf("\n =============================\n | CHOSEN |\n | ROUGH |\n =============================\n ----> ");

Sleep(1500);

}

else

{

system("cls"); // . .

printf("\n =============================\n | INVALID VALUE |\n | 1.Smooth 2.Normal 3.Rough |\n =============================\n");

Sleep(1500);

system("cls");

printf("\n =============================\n | CHOOSE SMOOTHNESS |\n | 1.Smooth 2.Normal 3.Rough |\n =============================\n");

continue;

}

}

system("cls");

printf("\n =============================\n | CHOOSE SATURATION |\n | IN RANGE 70 - 120 |\n =============================\n");

flag = 0; //indicator for the following loop, becomes 1 if value is correct, otherwise stays at 0

while (flag != 1)

{

printf(" ----> ");

scanf\_s("%d", &input); // choose saturation percent between 70 and 120 %

if (input < 70 || input > 120) //check for invalid input

{

system("cls");

printf("\n =============================\n | INVALID VALUE |\n | IN RANGE 70 - 120 |\n =============================\n");

Sleep(1500);

system("cls");

printf("\n =============================\n | CHOOSE SATURATION |\n | IN RANGE 70 - 120 |\n =============================\n");

continue;

}

flag = 1; // set flag to 1 when value is correct and chosen

double saturationPercent = 0;

if (input == 120)

{

saturationPercent = input \* 0.01;

}

else

{

saturationPercent = (120. - input) \* 0.01; // calculates the saturation percent, according to the user given % (70-120)

}

//if 120 todo

if (saturationPercent != 0) //check if saturation percent is larger than zero, to prevent errors in next line

{

//decreases ink costs with the saturation percent

for (int i = 0; i < 4; i++)

{

if (blackAndWhiteFlag == 0 && i < 3)

{

inkCostColors[i] -= inkCostColors[i] \* saturationPercent;

heads[i] -= inkCostColors[i];

}

else if (blackAndWhiteFlag == 1 && i == 3)

{

inkCostColors[i] -= inkCostColors[i] \* saturationPercent;

heads[i] -= inkCostColors[i];

}

}

}

}

for (int i = 0; i < 4; i++)

{

if (heads[i] < 0)

{

heads[i] = 0;

}

}

system("cls");

printf("\n =============================\n | PAGE PRINTED |\n | |\n =============================\n");

Sleep(1500);

system("cls");

printf("\n =============================\n |Red: %.2f, Green: %.2f |\n |Blue: %.2f, Black: %.2f |\n =============================\n", heads[0], heads[1], heads[2], heads[3]);

Sleep(5000);Фун

}

return 0;

}

Блокова схема:







