**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра Вычислительной техники**

отчет

**по лабораторной работе № 2**

**по дисциплине «Организация ЭВМ и систем»**

Тема: Исследование Видеосистемы(Текстовый режим)

|  |  |  |
| --- | --- | --- |
| Студент гр. 3311 |  | Пасечный Л.В. |
| Преподаватель |  | Гречухин М. Н. |

Санкт-Петербург

2024

**Цель работы.**

Знакомство с внутренним представлением различных типов данных, используемых компьютером при их обработке.

**Задание (вариант 16)**

1. Написать программу, в которой в окно с координатами (x1,у1,х2,у2) с шагами Т (секунд) и S (строк) выводится строка при всех возможных комбинациях цвета фона и цвета символов. Строка содержит обозначение цвета фона и символа. Для каждой комбинации цветов в окне должны выводиться номера или символьные обозначения цветов фона и символов (варианты приведены в табл. 2.4). Цвет окна должен соответствовать цвету фона.
2. Дополнить программу из п.1 скроллингом окна с направлением вниз, используя функции прерывания 10h BIOS.

3. Две отлаженные программы предъявить преподавателю.

**Постановка задачи и описание решения**

Для работы с видеосистемой используется TurboC++, в качестве виртуальной машины используется DosBox. Для работы с видеосистемой используются библиотеки dos.h и conio.h.

Создаётся окно размером 25, 12, 60, 23

Функция set\_color():

Отчищается консоль командой clrscr() и функцией clearScr() .

Заданы два цикла for, в которых проходим по списку цветов.

Далее мы задаём цвета с помощью функции set\_bg\_color().

С помощью функций sprintf(Добавляет значения, которые мы хотим вывести в массив буфер) и cprintf(выводит записанное значение в буфере в заданном окне). В зависимости от кода происходит либо скролинг либо стирание написанного значения.

Функция set\_bg\_color():

Содержит textattr, это функция где первые 4 бита отвечают за цвет текста, следующие 3 бита отвечают за цвет фона, а последний (левый крайний бит) отвечает за мерцание текста.

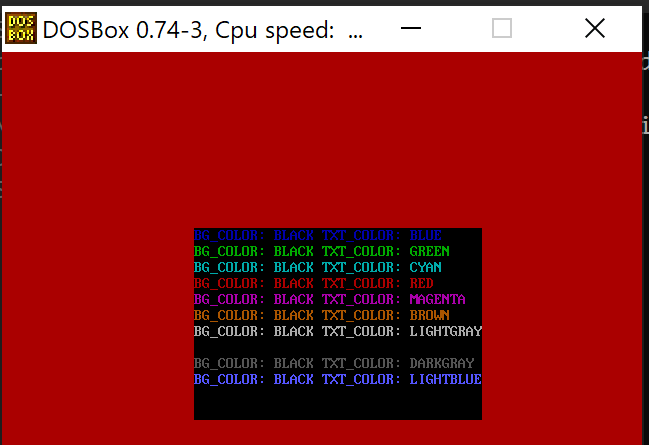
Функция scroll:

с помощью регистров задаются координаты окна, и создаётся скролинг с помощью атрибута.

**Контрольные примеры**

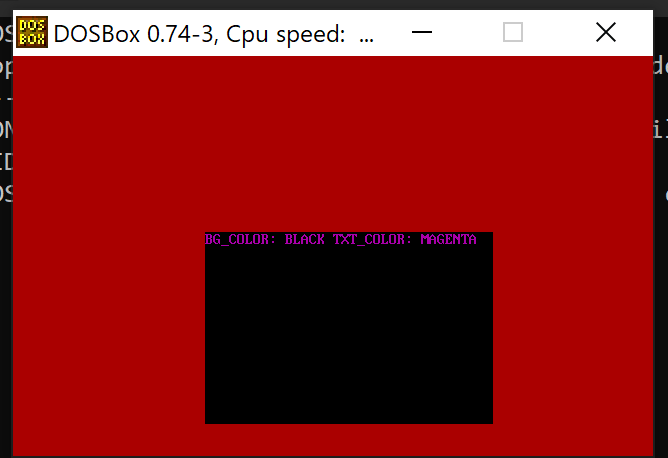
**Пример 1:**

Со скролингом



**Пример 2:**

**Без скролинга**



**Текст программы**

**Без сколлинга:**

#include <stdio.h>

#include <conio.h>

#include <stdlib.h>

#include <dos.h>

const char\* COLORS[] = {"BLACK","BLUE","GREEN","CYAN","RED","MAGENTA",

"BROWN", "LIGHTGRAY", "DARKGRAY", "LIGHTBLUE", "LIGHTGREEN"

,"LIGHTCYAN", "LIGHTRED", "LIGHTMAGENTA", "YELLOW", "WHITE"

};

void set\_bg\_color(int color, int txt\_color){

textattr((color << 4) | txt\_color);

}

void clearScr(int color){

set\_bg\_color(color, 0);

clrscr();

}

void set\_color(int x1, int x2, int y1, int y2, float T, int S){

char bfr1[40];

char bfr2[40];

int line = 0;

int back\_ground, text, attr;

clrscr();

clearScr(4);

window(25, 12, 60, 23);

for (back\_ground = 0; back\_ground < 16; back\_ground++){

clearScr(back\_ground);

for(text = 0; text < 16; text++){

set\_bg\_color(back\_ground, text);

sprintf(bfr1, "BG\_COLOR: %s TXT\_COLOR: %s", COLORS[back\_ground], COLORS[text]);

cprintf("%s\r\n", bfr1);

delay((int)(T\*1000));

line += 2;

if (line >= S){

clearScr(back\_ground);

line = 0;

}

}

}

clrscr();

clearScr(4);

}

int main(){

int x1,y1,x2,y2, S;

float T;

x1 = 25;

y1 = 12;

x2 = 60;

y2 = 23;

T = 0.6;

S = 1;

set\_color(x1, y1, x2, y2, T, S);

return 0;

}

**Со скролингом:**

#include <stdio.h>

#include <conio.h>

#include <stdlib.h>

#include <dos.h>

const char\* COLORS[] = {"BLACK","BLUE","GREEN","CYAN","RED","MAGENTA",

"BROWN", "LIGHTGRAY", "DARKGRAY", "LIGHTBLUE", "LIGHTGREEN"

,"LIGHTCYAN", "LIGHTRED", "LIGHTMAGENTA", "YELLOW", "WHITE"

};

void set\_bg\_color(int color, int txt\_color){

textattr((color << 4) | txt\_color);

}

void clearScr(int color){

set\_bg\_color(color, 0);

clrscr();

}

enum {ENTIRE, UP, DOWN};

void scroll(int direction, char l\_row, char l\_col, char r\_row, char r\_col, char attr, int line){

union REGS r;

if (direction == DOWN) r.h.ah = 7;

else r.h.ah = 6;

r.h.al = line;

r.h.ch = l\_row; r.h.cl = l\_col; r.h.dh = r\_row; r.h.dl = r\_col;

r.h.bh = attr;

int86(0x10, &r, &r);

}

void set\_color(int x1, int x2, int y1, int y2, float T, int S){

char bfr1[40];

int line = 0, flag = 0;

int back\_ground, text, attr;

clrscr();

clearScr(4);

window(25, 12, 60, 23);

clearScr(4);

for (back\_ground = 0; back\_ground < 16; back\_ground++){

clearScr(back\_ground);

for(text = 0; text < 16; text++){

set\_bg\_color(back\_ground, text);

sprintf(bfr1, "BG\_COLOR: %s TXT\_COLOR: %s", COLORS[back\_ground], COLORS[text]);

cprintf("%s\r\n", bfr1);

delay((int)(T\*1000));

line += 2;

if (line >= S){

scroll(DOWN, x1, y1, x2, y2, 0x07, 1);

}

}

if (flag && COLORS[back\_ground] == "BLACK"){

clrscr();

clearScr(4);

break;

}else if(COLORS[back\_ground] == "BLACK"){

flag++;

}

}

}

int main(){

int x1,y1,x2,y2, S;

float T;

x1 = 25;

y1 = 12;

x2 = 60;

y2 = 23;

T = 0.6;

S = 1;

set\_color(x1, y1, x2, y2, T, S);

return 0;

}

**Выводы.**

В ходе работы мы научились работать с видеосистемами. Познакомились с библиотеками conio.h dos.h. Работали в виртуальной машине .