Белорусский государственный технологический университет

Кафедра Информационных Систем и Технологий

**Курс «Операционные системы и системное программирование»**

**Лабораторная работа 3. Работа с консолью**

Выполнил: Сазановец Я.И.

ФИТ 3 курс 4 группа

Минск 2018

Задание 1. **Управление выводом в разные позиции курсора.**

#include "stdafx.h"

#include "windows.h"

int \_tmain(int argc, \_TCHAR\* argv[])

{

HANDLE consoleOutput;

COORD cursorPos;

// Получаем хэндл консоли

consoleOutput = GetStdHandle(STD\_OUTPUT\_HANDLE);

// Задаем координаты курсора и перемещаем курсор

cursorPos.X = 30;

cursorPos.Y = 3;

SetConsoleCursorPosition(consoleOutput, cursorPos);

// Выводим строку

printf("Test string at position (30, 3)");

// Повторяем с другими координатами...

cursorPos.X = 15;

cursorPos.Y = 8;

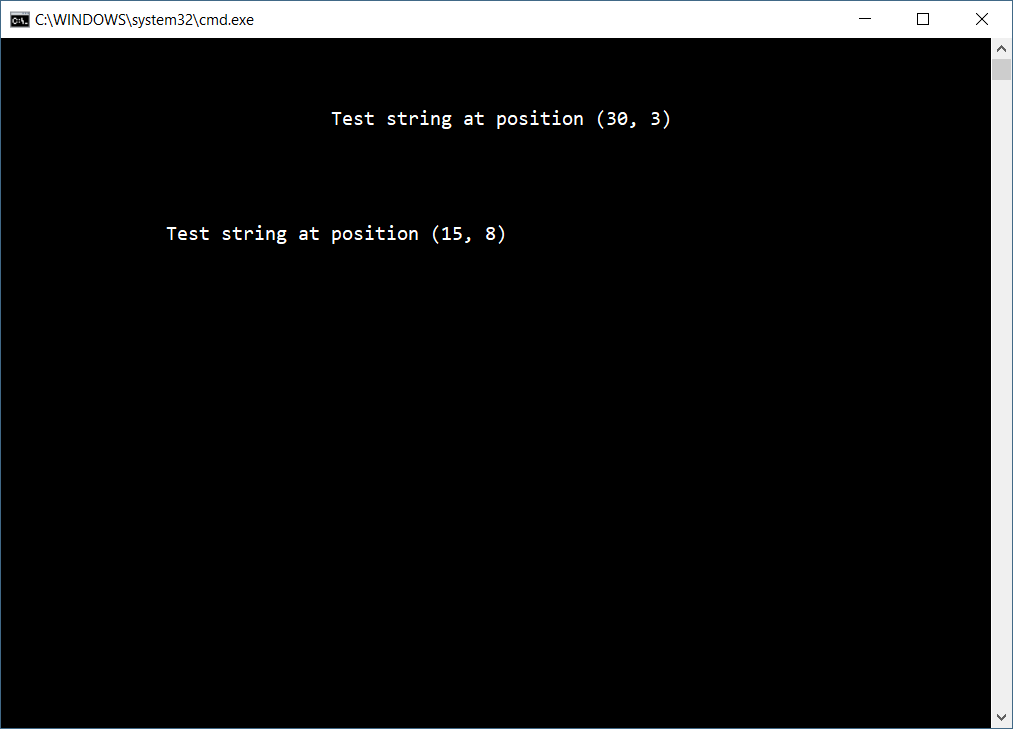
SetConsoleCursorPosition(consoleOutput, cursorPos);

printf("Test string at position (15, 8)");

getchar();

return 0;

}



Задание 2. **Вывод в цвете.**

#include "stdafx.h"

#include "windows.h"

#include <iostream>

// прототип недокументированый функции

typedef BOOL(WINAPI \*SETCONSOLEFONT)(HANDLE, DWORD);

SETCONSOLEFONT SetConsoleFont;

int \_tmain(int argc, \_TCHAR\* argv[])

{

//получить заголовок потока вывода

HANDLE myConsoleHandle = GetStdHandle(STD\_OUTPUT\_HANDLE);

TCHAR str[] = TEXT("HELLO TO EVERYBODY\n");

//получить заголовок программы (модуля)

HMODULE hmod = GetModuleHandleA("KERNEL32.DLL");

SetConsoleFont = (SETCONSOLEFONT)GetProcAddress(hmod, "SetConsoleFont");

if (!SetConsoleFont)

{

// если ошибка

std::cout << "error\n"; exit(1);

}

// устанавливаем 3 шрифт

SetConsoleFont(myConsoleHandle, 3);

SetConsoleTextAttribute(myConsoleHandle, FOREGROUND\_GREEN | FOREGROUND\_INTENSITY);

short x = 5; short y = 6;

COORD pos = { x, y };

HANDLE hConsole\_c = CreateConsoleScreenBuffer(GENERIC\_READ | GENERIC\_WRITE, 0, NULL, CONSOLE\_TEXTMODE\_BUFFER, NULL);

SetConsoleActiveScreenBuffer(hConsole\_c);

DWORD len = 18;

DWORD dwBytesWritten = 0;

WriteConsoleOutputCharacter(hConsole\_c, str, len, pos, &dwBytesWritten);

pos.X = 10;

pos.Y = 8;

WriteConsoleOutputCharacter(hConsole\_c, str, len, pos, &dwBytesWritten);

pos.X = 15;

pos.Y = 10;

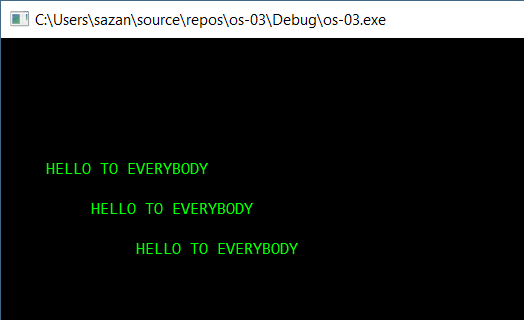
WriteConsoleOutputCharacter(hConsole\_c, str, len, pos, &dwBytesWritten);

getchar();

CloseHandle(hConsole\_c);

return 0;

}



Задание 3. **Управление цветами в выводе.**

#include "stdafx.h"

#include "windows.h"

const WORD colors[] =

{

0x1A, 0x2B, 0x3C, 0x4D, 0x5E, 0x6F,

0xA1, 0xB2, 0xC3, 0xD4, 0xE5, 0xF6

};

int \_tmain(int argc, \_TCHAR\* argv[])

{

HANDLE hstdout = GetStdHandle(STD\_OUTPUT\_HANDLE);

WORD index = 0;

// Remember how things were when we started

CONSOLE\_SCREEN\_BUFFER\_INFO csbi;

GetConsoleScreenBufferInfo(hstdout, &csbi);

// Tell the user how to stop

SetConsoleTextAttribute(hstdout, 0xEC);

//std::cout << "Press any key to quit.\n";

printf("press any key \n");

while (index<13)

{

SetConsoleTextAttribute(hstdout, colors[index]);

// std::cout << "\t\t\t\t Hello World \t\t\t\t" << std::endl;

printf("\t\t\t\t Hello World \t\t\t\t\n");

if (++index > sizeof(colors) / sizeof(colors[0]))

break;

}

SetConsoleTextAttribute(hstdout, csbi.wAttributes);

getchar();

return 0;

}



Задание 4. **Управление цветами и выводом в разные позиции курсора.**

#include "stdafx.h"

#include "windows.h"

const WORD colors[] =

{

0x1A, 0x2B, 0x3C, 0x4D, 0x5E, 0x6F,

0xA1, 0xB2, 0xC3, 0xD4, 0xE5, 0xF6

};

int \_tmain(int argc, \_TCHAR\* argv[])

{

HANDLE consoleOutput;

COORD cursorPos;

// Получаем хэндл консоли

consoleOutput = GetStdHandle(STD\_OUTPUT\_HANDLE);

WORD index = 0;

// Remember how things were when we started

CONSOLE\_SCREEN\_BUFFER\_INFO csbi;

GetConsoleScreenBufferInfo(consoleOutput, &csbi);

// Задаем координаты курсора и перемещаем курсор

cursorPos.X = 30;

cursorPos.Y = 3;

SetConsoleCursorPosition(consoleOutput, cursorPos);

SetConsoleTextAttribute(consoleOutput, 0xEC);

// Выводим строку

printf("Test string at position (30, 3)");

// Повторяем с другими координатами...

cursorPos.X = 15;

cursorPos.Y = 8;

SetConsoleCursorPosition(consoleOutput, cursorPos);

index++;

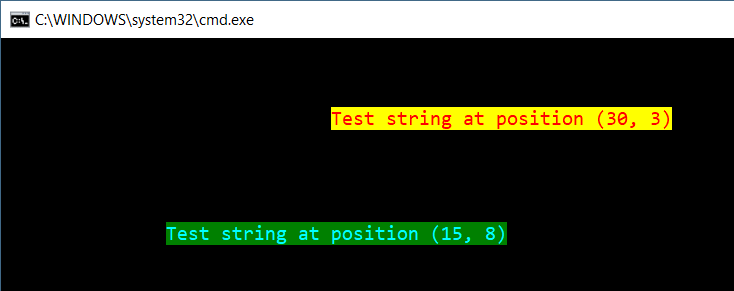
SetConsoleTextAttribute(consoleOutput, colors[index]);

printf("Test string at position (15, 8)");

getchar();

return 0;

}



Общие задания:

1. Нарисовать цветной столбик.

#include "stdafx.h"

#include "windows.h"

#include <iostream>

int \_tmain(int argc, \_TCHAR\* argv[])

{

HANDLE consoleOutput = GetStdHandle(STD\_OUTPUT\_HANDLE);

COORD cursor;

CONSOLE\_SCREEN\_BUFFER\_INFO csbi;

GetConsoleScreenBufferInfo(consoleOutput, &csbi);

for (int i = 0; i < 5; i++)

{

cursor.X = 15;

cursor.Y = 2 + i;

SetConsoleCursorPosition(consoleOutput, cursor);

SetConsoleTextAttribute(consoleOutput, 0x2B);

std::cout << " " << std::endl;

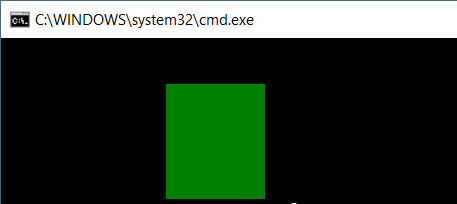
}

SetConsoleTextAttribute(consoleOutput, 0x0F);

system("pause");

return 0;

}



2. Нарисовать таблицу.

#include "stdafx.h"

#include <Windows.h>

#include <stdlib.h>

#include <iostream>

using namespace std;

void Print(int x, int y, char\* str, int color = 0x1A)

{

HANDLE consoleOutput;

COORD cursorPos;

consoleOutput = GetStdHandle(STD\_OUTPUT\_HANDLE);

cursorPos.X = x;

cursorPos.Y = y;

SetConsoleCursorPosition(consoleOutput, cursorPos);

SetConsoleTextAttribute(consoleOutput, color);

std::cout << str << std::endl;

}

void PrintTable(char columns[][10], int count\_columns, char lines[][10][10], int count\_lines)

{

char\* line = new char[64]{ "\t\t\t\t" };

char\* column = new char[20]{ " " };

int column\_width = 15;

int table\_width = column\_width \* count\_columns;

Print(0, 0, line);

for (int i = 0, l = 1; i < count\_lines + 1; i++, l+=2)

{

Print(0, l + 1, line);

Print(table\_width, l, column);

for (int k = 0, width = 5; k < count\_columns; k++, width += column\_width)

{

Print(width - 5, l, column);

if ((l -1) == 0)

Print(width, 1, columns[k], 0x0F);

else

Print(width, l, lines[i - 1][k], 0x0F);

}

}

std::cout << std::endl;

delete[] line;

delete[] column;

}

int main()

{

setlocale(NULL, "");

char columns[][10] = { "Фамилия", "Группа" };

char lines[][10][10] = { { "Сазановец", "4" } /\*, { "Калишок", "4" } \*/ };

PrintTable(columns, 2, lines, 1 /\*2\*/);

return 0;

}

