

# Лабораторное занятие 11

## Разработка проектов с обработкой событий

### 1 Цель работы

1.1 Приобрести навыки по созданию программ с обработкой пользовательских событий.

### 2 Литература

2.1 Фленов, М.Е. Библия C#. – 3 изд.– Санкт-Петербург: БХВ-Петербург, 2016.  
– Режим доступа: <https://ibooks.ru/reading.php?productid=353561>, только для зарегистрированных пользователей. – Загл. с экрана. – п.5.7.

### 3 Подготовка к работе

3.1 Повторить теоретический материал (см. п.2).

3.2 Изучить описание практической работы.

### 4 Основное оборудование

4.1 Персональный компьютер.

### 5 Задание

5.1 Создать бесконечный цикл, выводящий на консоль произвольное сообщение. Если пользователь нажимает на клавишу “С” (Cancel), завершать работу программы с соответствующим сообщением. Учитывайте, что у пользователя может быть нажат caps lock или русская раскладка. Для считывания клавиши можно использовать getch() из conio.h

5.2 С помощью таймера менять цвет консоли и текста на случайный каждые три секунды.

Для этого необходимо подключить библиотеку “Windows.h”.

Затем добавить функцию void SetConsoleColor(int color), листинг которой представлен ниже:

```
void SetConsoleColor(int color) {  
    HANDLE hConsole = GetStdHandle(STD_OUTPUT_HANDLE);  
    SetConsoleTextAttribute(hConsole, color);  
}
```

Добавить массив, содержащий доступные цвета:

```
int colors[] = { FOREGROUND_RED, FOREGROUND_GREEN, FOREGROUND_BLUE, FOREGROUND_RED |  
FOREGROUND_GREEN, FOREGROUND_RED | FOREGROUND_BLUE, FOREGROUND_GREEN |  
FOREGROUND_BLUE };
```

Добавить в код бесконечный цикл:

```
while (true) {  
    for (int color : colors) {  
        SetConsoleColor(color);  
        cout << "Цвет изменен!" << endl;  
        Sleep(3000); // Пауза в 3 секунды  
    }  
}
```

```
}  
}
```

Протестировать написанный код, запустив его.

5.3 Написать программу на с++, запускающую браузер chrome и открывающую сайт колледжа (arcotel.ru)

Для этого необходимо подключить библиотеку “Windows.h”.

Использовать функцию ShellExecute()

Подробнее ознакомиться с ней можно **в приложении**.

## 6 Порядок выполнения работы

6.1 Запустить Visual Studio и выполнить все задания из п.5.

6.2 Ответить на контрольные вопросы.

6.3 Составить электронный отчет и сохранить C:\Temp\KSK\

## 7 Содержание отчета

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

## 8 Контрольные вопросы

8.1 Что такое ООП?

8.2 Назовите основные парадигмы ООП.

8.3 Для чего создаются конструкторы?

## 9. Приложение

**ShellExecute** — это функция Windows API, предназначенная для запуска внешних программ и открытия документов. Она предоставляет простой и удобный способ взаимодействовать с операционной системой для выполнения различных действий, связанных с файлами и приложениями.

```
BOOL ShellExecute(  
    HWND hwnd,  
    LPCTSTR lpOperation,  
    LPCTSTR lpFile,  
    LPCTSTR lpParameter,  
    LPCTSTR lpDirectory,  
    INT nShowCmd  
);
```

- **hwnd**: Дескриптор родительского окна (обычно NULL).
- **lpOperation**: Операция, которую необходимо выполнить (например, "open", "print").
- **lpFile**: Путь к файлу или программе.
- **lpParameters**: Дополнительные параметры для программы.
- **lpDirectory**: Рабочий каталог для программы.

- **nShowCmd:** Как отобразить окно программы (например, SW\_SHOWNORMAL, SW\_HIDE).

Например:

```
ShellExecute(NULL, L"find", L"c:\\MyPrograms", NULL, NULL, SW_SHOWNORMAL);
```

```
ShellExecute(NULL, L"open", L"drome.exe", L"https://www.example.com", NULL, SW_SHOWNORMAL);
```