ФЕДЕРАЛЬНОЕ АГЕНСТВО СВЯЗИ

[ОРДЕНА ТРУДОВОГО КРАСНОГО ЗНАМЕНИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ "МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ И ИНФОРМАТИКИ"](http://www.list-org.com/search.php?type=name&val=ОРДЕНА%20ТРУДОВОГО%20КРАСНОГО%20ЗНАМЕНИ%20ФЕДЕРАЛЬНОЕ%20ГОСУДАРСТВЕННОЕ%20БЮДЖЕТНОЕ%20ОБРАЗОВАТЕЛЬНОЕ%20УЧРЕЖДЕНИЕ%20ВЫСШЕГО%20ОБРАЗОВАНИЯ%20%20МОСКОВСКИЙ%20ТЕХНИЧЕСКИЙ%20УНИВЕРСИТЕТ%20СВЯЗИ%20И%20ИНФОРМАТИКИ)

Кафедра Информационных Технологий

Самостоятельная работа по дисциплине:

**Разработка приложения «Производственный календарь»**

Выполнил студент

группы БИН1707

Кустов Александр

# **Введение**

приложение программа электронный

Когда появился первый календарь, не знает никто. Но точно известно, что само слово «календарь» родом из Древнего Рима. Календами называли дни, когда должники платили проценты. С тех времен календарь оброс множеством других дат. В нем выделяют праздники и выходные, назначают события. Для современного делового человека календарь - это вещь первой необходимости. Без него невозможно спланировать ни одной сделки, встречи или собрания, он помогает скоординировать действия.

Многие, вероятно, слышали о таком понятии как «вечные календари». Современное поколение при этом скорее всего понимает множество программ, предлагаемых в интернете. Подобные программы называются «Вечный календарь» и представляют из себя алгоритм расчета любой даты на очень длительный период. Более старшее поколение, возможно, помнит и другие «вечные календари». Подобные календари выпускались во времена Советского Союза достаточно широко, изготавливались они сначала из картона, потом появились и варианты из алюминия. Картонные календари выпускались в больших объемах в 70-е годы XX века. Качество календарей было достаточно плохим, дизайн неинтересным.

Вечные календари выпускались двух основных видов: в виде таблиц и в виде диска. Календари в виде таблиц иногда печатали в газетах и журналах, некоторые записные книжки также включали в себя подобный календарь.

Достаточно интересен был дисковый календарь. Основным элементом подобного рода календарей являлся диск, имевший два радиуса: внутренний, на который наносились, как правило, дни недели и внешний - на него наносили номера годов. Чтобы получить календарь на месяц, нужно было просто повернуть диск таким образом, чтобы номер нужного года совпал с номером определяемого месяца.

Данные календари не выпускаются уже очень давно - с 2000 года. В связи с широким распространением компьютерных «вечных» календарей, необходимость в них совершенно пропала. Да и люди предпочитают пользоваться для определения времени и планирования дел более удобными календарями: настенными настольными, карманными. Современные календари, изготовленные в современных типографиях, необычайно многообразны и удобны, функциональность в них сочетается с привлекательным дизайном. Ну а «вечный» календарь Вы всегда можете найти в интернете.

Ранее, до появления компьютеров и сотовых телефонов, календари были печатными. Поиск нужных дат производился вручную, что очень неудобно.

Целью данной курсовой работы является реализация электронного приложения «Производственный календарь», которое позволяло бы наиболее быстро определять даты выходных, которые совпали с праздниками и даты, на которые они будут перенесены.

Исходя из цели курсовой работы, было создано электронное приложение «Производственный календарь» на языке программирования Си. Интерфейс программы представлен в виде меню, состоящего из нескольких пунктов, каждый из которых отвечает за определенные операции, которые пользователь может произвести с программой.

Разработанное электронное приложение может быть полезным для всех пользователей персонального компьютера, у которых есть необходимость знать даты совпадений основных праздников с выходными днями, а также даты на которые будут перенесены выходные.

# **1. Анализ предметной области**

При анализе выбранной предметной области было проанализировано несколько способов решения поставленной задачи.

Следует отметить, что существует большое количество способов реализации производственного календаря, которые ненамного отличаются друг от друга, общий принцип при этом не меняется.

Из существующих решений стоит отметить «Производственный календарь рис. 2.1». Программа предназначена для определения дат начала и конца отпуска с учетом праздников.

Позволяет определить даты всех праздников и выходных в выбранном году. Также программа имеет возможность выполнить подсчет даты окончания отпуска с учетом праздников. Вся информация предоставлена в наиболее удобном для пользователя виде.

В ходе анализа работы данной программы существенных недостатков не было выявлено.

В разрабатываемом приложении пользователю необходимо:

Ввести необходимый год с клавиатуры командой «Ввод данных», либо командой «Считать данные из файла» считать данные из файла в программу для дальнейшей обработки;

Выбрать в интерактивном меню необходимую операцию («Праздники, которые выпали на выходные», «Количество праздников, которое совпадает с выходными») для обработки производственного календаря;

Далее, с производственным календарем можно производить все предложенные операции, в том числе записывать и считывать данные из файла.

Для окончания работы приложения необходимо выбрать соответствующий пункт меню «Выход».

# **2. Проектирование**

Интерфейс пользователя должен обеспечивать интерактивное взаимодействие при помощи меню, предоставляющего выбор любой из реализованных операций в любой последовательности. Список реализуемых операций таков:

Ввод данных;

Праздники, которые выпали на выходные;

Количество праздников, которое совпадает с выходными;

Очистить экран;

Вывести все данные;

Записать данные в файл;

Считать данные из файла;

О программе;

При реализации необходимо предусмотреть проверку на адекватность всех вводимых данных - недопустимые значения при вводе года, недопустимые значения при считывании данных из файла.

Также для удобства необходимо реализовать операцию очистки экрана, с последующим выводом меню и последних данных, веденных пользователем.

Выполнение операций должно сопровождаться подсказками и предупреждениями. Как, например, при выборе операций «Ввод данных», «Праздники, которые выпали на выходные», «Количество праздников, которое совпадает с выходными», «Вывести все данные», «Записать данные в файл», «Считать данные из файла», то при отсутствии данных в программе выводить соответствующие сообщения об ошибке.

При реализации необходимо предусмотреть проверку на адекватность всех вводимых данных - значение не соответствующее допустимому диапазону при вводе года и выборе пункта меню.

Реализуемые в программе операции для удобства должны выполняться в отдельном окне, поэтому необходимо предусмотреть очистку экрана, при выборе некоторых операций с последующим выводом меню на экран.

Будет полезным добавить дополнительный пункт в меню - «О программе», прочитав который, пользователь сможет ознакомиться с возможностями программы.

Выход из программы осуществляется с помощью выбора соответствующего пункта меню «Выход».

# **3. Кодирование и отладка**

Для выполнения курсового проекта была выбрана среда разработки приложений для языков программирования C/C++ Visual Studio 2015, поскольку она содержит очень удобный редактор, адаптированный под:8.187 с пакетом обновления 1 (SP1)Server 2012 R2Server 2012Server 2008 R2 с пакетом обновления 1 (SP1)

и имеет стильный, удобный интерфейс на русском языке..h - служит для генерации файла предкомпилированных заголовков; в него включено большинство стандартных и используемых в каждом приложении включаемых файлов.

Для реализации поставленных задач были использованы следующие стандартные

библиотеки языка Си:.h - содержит константы и объявления функций и типов, используемых для различных операций стандартного ввода и вывода, таких как:- открытие файла для определённых типов чтения и записи;- закрытие файла;- используется для вывода в стандартный поток вывода;- используется для ввода из стандартного потока ввода;- используется для вывода в файл;- используется для ввода из файла;- заголовочный файл с классами, функциями и переменными для организации ввода-вывода в языке программирования C++.

Для достижения поставленной цели был разработан необходимый набор функций:

Функция null предназначена для ввода изначальных значений массива. Описание функции:

void null (int x[])

Тип возвращаемого значения - void, поскольку функция ничего не возвращает. Формальные параметры: int x[] - передается одномерный массив, состоящий из элементов каждый из которых определяет день недели определенного праздника. Полная реализация функции представлена в приложении (листинг Б., строки 7-16).

) Функция dayWeek определяет день недели для каждого из праздников в году, веденным пользователем. Описание функции:dayWeek (int g, int x[], int r)

Тип возвращаемого значения - void, поскольку функция ничего не возвращает. Формальные параметры: int g - год, веденный пользователем, int x[] - передается одномерный массив, каждый из элементов которого задает день недели определенного праздника, int r - переменная, которая считает количество високосных лет, от 1970 до введенного пользователем.

) Функция weekend предназначена для вывода праздников, которые выпали на выходные, и вывода дат, на которые они были перенесены. Описание функции:weekend (int y[], char c[7] [33], char p[7] [9], int x[])

Тип возвращаемого значения - void, поскольку функция ничего не возвращает. Формальные параметры: int y[] - передается одномерный массив, состоящий из элементов, каждый из которых задает дату определенного праздника, int x[] - передается одномерный массив, каждый из элементов которого задает день недели определенного праздника,

char c[7] [33] - передается двумерный массив, каждый из элементов которого имеет свое текстовое значение, состоящее из названия праздника и месяца, в который он празднуется, char p[7] [9] - передается двумерный массив, каждый из элементов которого имеет свое текстовое значение, состоящее из месяца определенного праздника.

) Функция numberHolidays предназначена для подсчета количества праздников которые совпадают с выходными. Описание функции:

void numberHolidays (int x[])

Тип возвращаемого значения - void, поскольку функция ничего не возвращает. Формальные параметры: int x[] - передается одномерный массив, каждый из элементов которого задает день недели определенного праздника.

) Функция input предназначена для ввода пользователем года в пределах от 2005 до 2099. Описание функции:

int input()

Тип возвращаемого значения - int, поскольку с помощью этой функции в главную программу возвращается номер года, введенный пользователем.

Функция возвращает значение g, которое в дальнейшем используется для функций: dayWeek, print, fpr, leapYear.

) Функция menu предназначена для вывода меню программы пользователю. Описание функции:

void menu()

Тип возвращаемого значения - void, поскольку функция ничего не возвращает.

) Функция print предназначена для вывода, введенного пользователем года. Описание функции:

void print (int g)

Тип возвращаемого значения - void, поскольку функция ничего не возвращает. Формальные параметры: int g - передается год введенный пользователем.

) Функция fpr предназначена для записи данных в файл.

Описание функции:

void fpr (int x[], int g)

Тип возвращаемого значения - void, поскольку функция ничего не возвращает. Формальные параметры: int g - передается год введенный пользователем, int x[] - передается одномерный массив, каждый из элементов которого задает день недели определенного праздника.

) Функция fsc предназначена для считывания, ранее использованных программой, данных из файла. Описание функции:

int fsc (int x[])

Тип возвращаемого значения - void, поскольку функция ничего не возвращает. Формальные параметры: int x[] - передается одномерный массив, каждый из элементов которого задает день недели определенного праздника.

Функция возвращает значение g, которое в дальнейшем используется для функций: dayWeek, print, fpr.

) Функция leapYear предназначена подсчета количества високосных лет от 1970 до введенного пользователем. Описание функции:

int leapYear (int g)

Тип возвращаемого значения - int, поскольку с помощью этой функции в главную программу возвращается количество високосных лет от 1970 до введенного пользователем. Формальные параметры: int g - передается год введенный пользователем

) Функция about предназначена для вывода информации о программе.

Описание функции:

void about()

Тип возвращаемого значения - void, поскольку функция ничего не возвращает

Реализованные функции в дальнейшем используются в основном теле программы - в главной функции, глобальные переменные которой следующие:

int pu - номер операции из меню программы, изначально принимает значение 1.

int r - количество високосных лет от 1970 до введенного пользователем.

int g - год введенный пользователем, изначально принимает значение 0.

int v - год от которого начинается отсчет, изначально принимает значение 0.

int x[7] - дни недели каждого из праздников, изначально принимает значения 3,1,7,5,6,5,3.

int y[7] - дни дат каждого из праздников, принимает значения 7,23,8,1,9,12,4.c[7] [33] - содержит текстовые значения названия праздников и месяцев их дат, принимает значения «января рождество», «февраля день защитника отечества», «марта женский день», «мая праздник весны и труда», «мая день победы», «июня день России», «ноября день народного единства».p[7] [9] - содержит текстовые значения месяцев каждого из праздников, принимает значения «января», «февраля», «марта», «мая», «мая», «июня», «ноября».\* file - указатель на переменную файлового типа;

Выход из программы осуществляется выбором пункта меню «0».

Подробное описание работы некоторых функций:

dayWeek

void dayWeek (int g, int x[], int r)

{i;= g - 1970;(i = 0; i < 7; i++)[i] = x[i] + g + r;(i = 0; i < 7; i++)

if (x[i] > 6)[i] = x[i]% 7;

}

К каждому из значений дней недели праздников 1970 года прибавляются разность введенного и 1970 года, и количество високосных лет за промежуток от 1970 г. до введенного пользователем. Далее для ранее найденных значений (больших 7) находится остаток от деления на 7, т.к. значения больше 6 не соответствуют значениям дней недели (1 - пн., 2 - вт., 3 - ср, 4 - чт., 5 - пт., 6 - сб., 0 - вс.).

leapYear

int leapYear (int g)

{

int r = 0, v = 1972;

r = ((g - v) / 4)+1;

return r;

}

Для того чтобы найти кол-во високосных лет, разность между введенным пользователем и 1972 годами делится на 4, и к ней же добавляется 1, т.к. високосный 1972 год не входит в эту разность.

fprfpr (int x[], int g)

{i;\* fp;

fopen\_s (&fp, «file.txt», «wb»);

if (fp)

{

for (i = 0; i<7; i++)(&x[i], sizeof (x[i]), 1, fp);(&g, sizeof(g), 1, fp);

}(fp);

}

Запись данных в файл.\* fp - указатель на переменную файлового типа;

Функция fopen\_s предназначена для открытия файла, где (ссылка на указатель, на переменную файлового типа, имя файла, создает двоичный файл для записи).

Функция fclose - закрывает файл.- функция, позволяющая записывать данные в файл, где (ссылка на указатель, на данные, которые будут записаны в файл; размер каждого элемента в байтах; счетчик, определяет, сколько считывается или записывается элементов данных; указатель, на переменную файлового типа).

fscfsc (int x[])

{i = 0, g = 0;\* fp;\_s (&fp, «file.txt», «rb»);(fp)

{(i!= 7)

{((fread (&x[i], sizeof (x[i]), 1, fp)) == 1)++;

{(«Ошибка, файл не имеет данных для считывания, либо ониодержат недопустимые значения\n»);

return 0;

}((i == 7) && ((fread (&g, sizeof(g), 1, fp))!= 1))

{(«Ошибка, файл не имеет данных для считывания, либо ониодержат недопустимые значения\n»);

return 0;

}

}

}

fclose(fp);

printf («Считывание данных из файла произошло успешно.\n»);

return g;

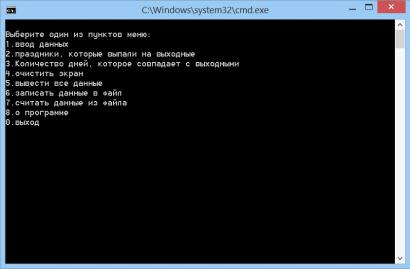
}

Считывание данных из файла

# **4. Тестирование**

Для проверки корректности работы программы проведем несколько тестов.

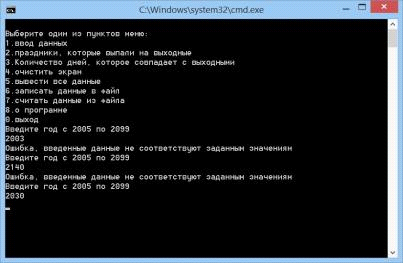
Сначала компилируем и запускаем программу. Пользователь увидит диалоговое окно (см. рисунок), следующего содержания:



Окно программы в начальный момент запуска.

Тест №1: Проверка ввода года.

После запуска программы, выбираем пункт меню «1». Далее осуществляем ввод данных, в формате: «Введите год с 2005 по 2099» - ввод осуществляется без пробелов, дефисов и других знаков препинания. При некорректном вводе данных выводится сообщение об ошибке (см. строки 13, 16 рисунка):

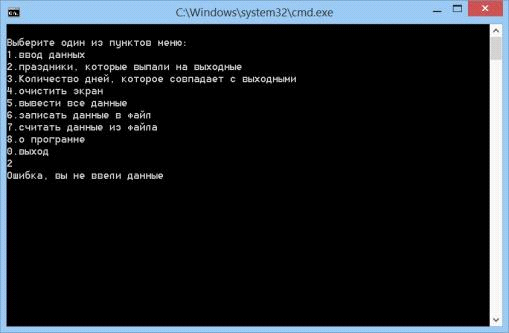


Диалоговое окно проверки теста №1

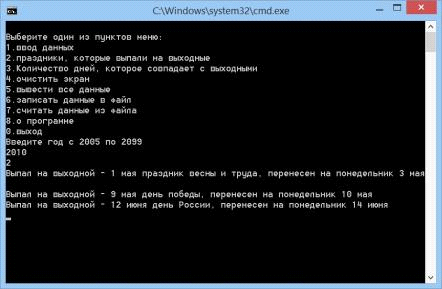
В результате проведенного теста можно сделать вывод, что ввод данных прошел успешно. Программа работает корректно.

Тест №2: Проверка совпадения выходных и дат праздников, введенного пользователем года.

Выбираем пункт меню «2». Далее программа выводит праздники, которые совпали с выходными и даты, на которые они будут перенесены (см. рисунок 4.2.1). При отсутствии данных из «1» пункта меню, выводится сообщение об ошибке (см. строка 12 рисунка 4.2):



Диалоговое окно проверки теста №2

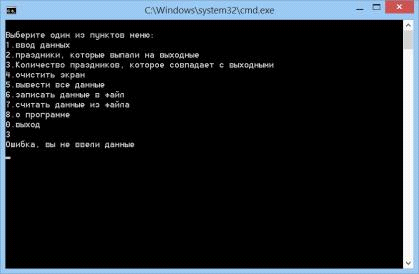


Диалоговое окно проверки теста №2

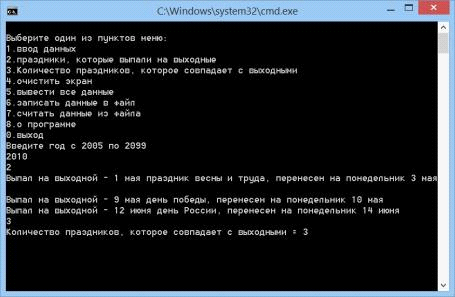
В результате проведенного теста можно сделать вывод, что вывод данных прошел успешно. Программа работает корректно.

Тест №3: Подсчет количества праздников, которое совпадает с выходными.

Выбираем пункт меню «3». Далее программа выводит количество праздников, которое совпадает с выходными (см. рисунок 4.3.1. При отсутствии данных из «1» пункта меню, выводится сообщение об ошибке (см. строка 12 рисунка 4.3):



Диалоговое окно проверки теста №3

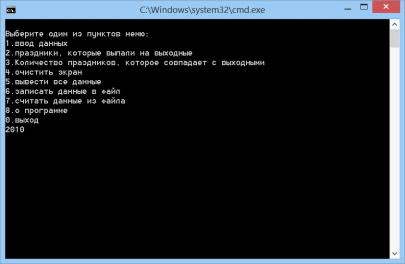


Диалоговое окно проверки теста №3

В результате проведенного теста можно сделать вывод, что вывод данных прошел успешно. Программа работает корректно.

Тест №4: Очистка экрана.

Выбираем пункт меню «4», предварительно выбрав «1», «2», «3» пункты. Далее программа очищает экран и выводит меню, с введенным пользователем годом. (см. рисунок 4.4).

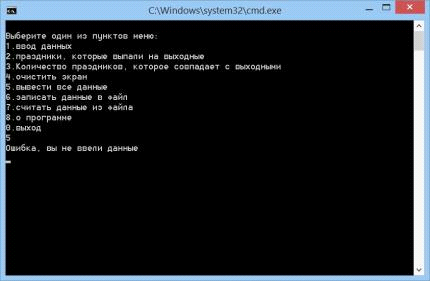


Диалоговое окно проверки теста №4

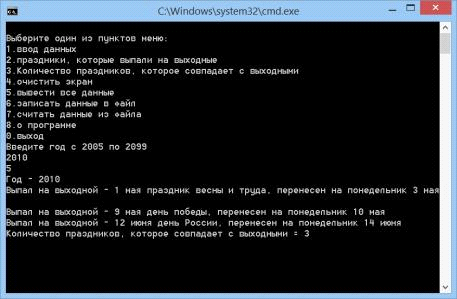
В результате проведенного теста можно сделать вывод, что очистка экрана прошла успешно. Программа работает корректно.

Тест №5: Вывод всех данных.

Выбираем пункт меню «5», предварительно выбрав «1» пункт. Далее программа выводит данные из «1», «2», «3» пунктов меню (см. рисунок 4.5.1). При отсутствии данных из «1» пункта меню, выводится сообщение об ошибке (см. строка 12 рисунка 4.5):



Диалоговое окно проверки теста №5

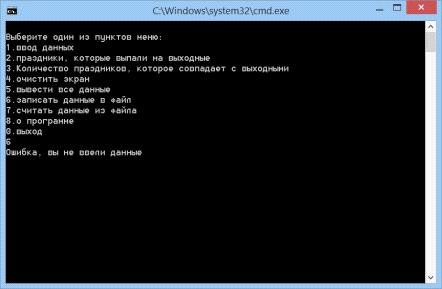


Диалоговое окно проверки теста №5

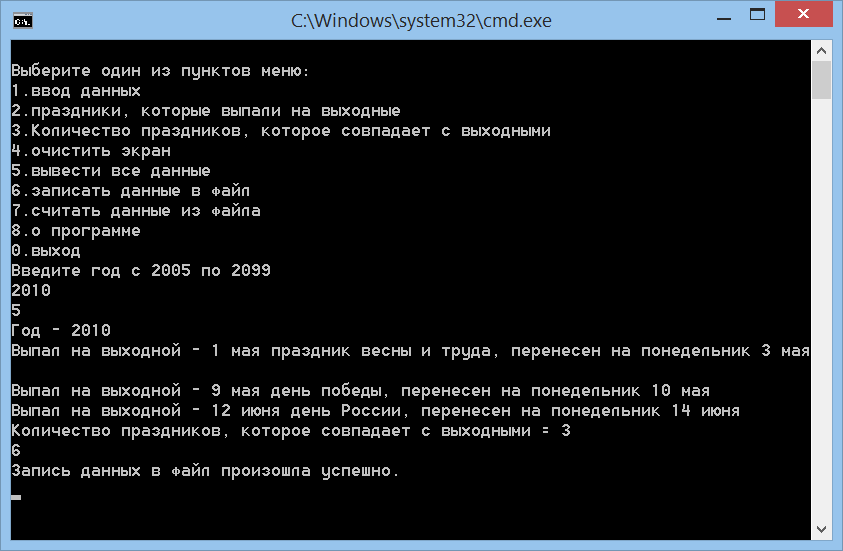
В результате проведенного теста можно сделать вывод, что вывод всех данных прошел успешно. Программа работает корректно.

Тест №6: Запись и считывание данных из файла.

Выбираем пункт меню «6», предварительно выбрав «1» пункт. Далее программа записывает данные в файл (см. рисунок 4.6.1). При отсутствии данных из «1» пункта меню, выводится сообщение об ошибке (см. строка 12 рисунка 4.6):

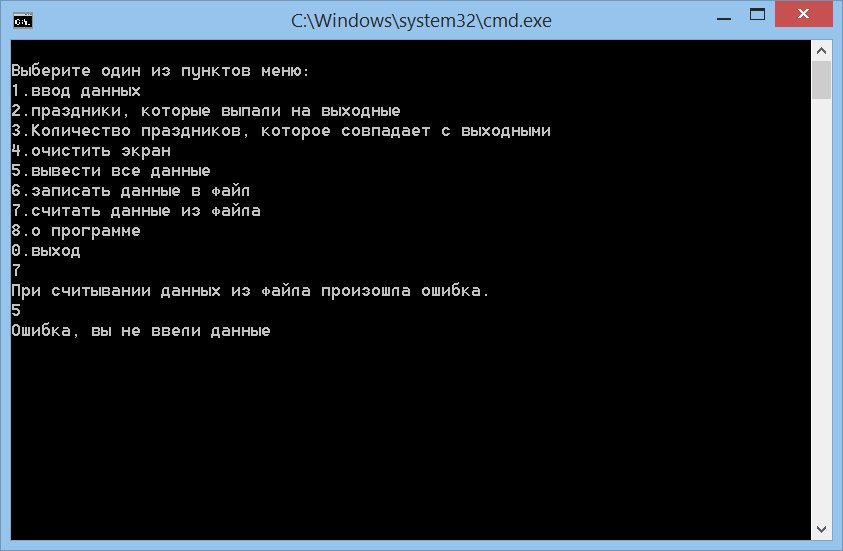


Диалоговое окно проверки теста №6

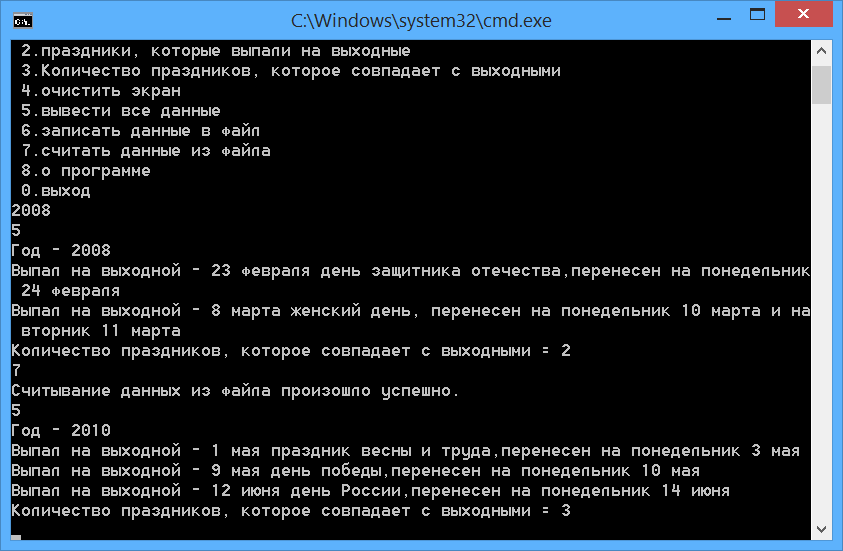


Диалоговое окно проверки теста №6

Выбираем пункт меню «7». Далее программа считывает данные из файла (см. рисунок 4.7.1). При отсутствии данных в файле, либо при недопустимых значениях считанных данных из файла, выводится сообщение об ошибке (см. строка 12 рисунка 4.7):



Диалоговое окно проверки теста №6



Диалоговое окно проверки теста №6

В результате проведенного теста можно сделать вывод, что запись и считывание данных из файла работают правильно. Программа работает корректно.

Проведенные тесты полностью подтверждают корректность работы программы.

**5. Сопровождение**

Разработанный электронный продукт, выполненный на языке программирования Си, поддерживается на большинстве операционных систем, среди которых Windows 7, windows 8, windows 8.1, windows 10.

Помимо этого, программа корректно работает как на 32-разрядных системах, так и на 64-разрядных системах. Обратная совместимость 64-битных версий Windows с 32-битными приложениями осуществляется за счет механизмов, реализованных в WoW64.(Windows-on-Windows 64-bit) - подсистема операционной системы Windows, позволяющая запускать 32-битные приложения на всех 64-битных версиях Windows.

Подсистема WoW64 не поддерживает следующие программы:

программы, скомпилированные для 16-разрядных операционных систем;

программы режима ядра, скомпилированные для 32-разрядных операционных систем.

Для положительной работы программы необходимо всего 37 Кбайт свободной памяти на жестком диске компьютера и 718 Кбайт оперативной памяти. Исходя из расчетных данных, каждый пользователь современного ПК может позволить себе использование разработанного программного продукта «Производственный календарь».

Приблизительный размер выходного текстового файла, непосредственно содержащего необходимые для работы программы данные, вычисляется по формуле

V = 32\*N [байт] Так как максимальное количество символов производственного календаря = 32, а информационный вес одного символа = 1 байт, то максимальный размер выходного файла составляет 32 байт.

# **6. Руководство пользователя**

При запуске программы появляется меню, предоставляющее на выбор пользователю все возможные операции с программой производственный календарь;

Для внесения данных в производственный календарь необходимо воспользоваться опцией «Ввод данных». Затем ввести необходимый год (минимум 2005, максимум 2099);

Для того чтобы определить даты праздников, которые выпали на выходные необходимо воспользоваться опцией «Праздники, которые выпали на выходные». При вызове этой опции программа выведет даты всех праздников, которые выпали на выходные, а также даты, на которые они будут перенесены;

Для подсчета и вывода количества праздников, которые совпали с выходными необходимо воспользоваться опцией «Количество праздников, которое совпадает с выходными»;

Очистка экрана производится при помощи операции «Очистить экран». После очистки экрана выводится меню и год, введенный пользователем, либо считанный из файла;

Вывод всех данных производится при помощи операции «Вывести все данные». В результате выполнения данной операции программа выводит год, введенный пользователем, либо считанный из файла, «Праздники, которые выпали на выходные», «Количество праздников, которое совпадает с выходными»;

Запись данных в файл производится при помощи операции «Записать данные в файл». В случае успешной записи данных файл программа выведет «Запись данных в файл произошла успешно», иначе «Ошибка вы не ввели данные»;

Чтение данных из файла производится при помощи операции «Считать данные из файла». В случае успешного считывания данных из файла программа выведет «Считывание данных из файла произошло успешно», иначе «Ошибка, файл не имеет данных для считывания, либо они содержат не допустимые значения»;

Для выхода из программы необходимо воспользоваться операцией «Выход».

# **7. Замечания**

При попытке выполнения операций «Праздники, которые выпали на выходные», «Количество праздников, которое совпадает с выходными», «Вывести все данные», «Записать данные в файл», без предварительного введения или считывания из файла данных выдаются соответствующие предупреждения о невозможности выполнения данных операций;

При повторном выборе операции «Ввод данных» все несохраненные до этого данные будут удалены;

При повторном считывании данных из файла все несохраненные до этого данные будут удалены;

Считывание данных возможно только из файлов, созданных при помощи данной программы;

Команда «Считать данные из файла» при отсутствии данных в файле или наличия недопустимых значений этих данных, выведет ошибку;

При записи данных в файл, все ранее записанные данные будут удалены.

Операция главного меню «О программе» дает возможность пользователю ознакомиться с настоящим программным продуктом.

# **Заключение**

В ходе выполнения курсового проекта была изучена и проанализирована учебно-методическая документация, учебная литература и интернет по выбранной теме.

Были найдены всевозможные способы достижения поставленной цели, анализ которых позволил выбрать наиболее оптимальный способ, для достижения которого был разработан необходимый набор универсальных функций.

Проектирование позволило сформировать структуру ПО и разработать необходимые программные алгоритмы, разработать интерфейс пользователя, форматы ввода, вывода и визуализации информации.

Исходя из результатов анализа и проектирования, было произведено кодирование и отладка приложения на языке программирования Си.

Итогом данного курсового проектирования является приложение «Производственный календарь». Приложение позволяет по указанному году вычислить количество совпадений основных праздников с выходными днями и вывести эти даты, для каждого совпадения указать дату, на которую должен быть перенесен выходной, записывать данные в файл, а так же считывать ранее записанные, с помощью этого приложения, данные для дальнейшей обработки.

Для проверки корректности работы программы было произведено несколько тестов, в результате которых не было выявлено явных ошибок или неточностей в работе программы.

Были разработаны методические рекомендации для пользователя, в которых были отмечены некоторые особенности работы программы.

Исходя из вышеперечисленного, следует считать, что задачи работы полностью выполнены и цель курсового проектирования достигнута.

# **Использованная литература**

Электронный каталог КиберФорум [Электронный ресурс]: форум начинающих и профессиональных программистов, системных администраторов, администраторов баз данных. Раздел C/C++. Режим доступа: http://www.cyberforum.ru/cpp/.

Гришмановский П.В., Даниленко И.Н. Курсовое проектирование. Разработка программного обеспечения. Методические указания к выполнению курсовых проектов для студентов специальностей 210100 «Управление и информатика в технических системах» и 220400 «Программное обеспечение вычислительной техники и автоматизированных систем». Сургут: Издательство СурГУ, 2004. - 23 с.

Кучин Н.В. Основы программирования на языке Си: учебное пособие / Н.В. Кучин, М.М. Павлова. - СПб: СПбГУАП, 2001. - 86 с.

Царев, Р.Ю. Программирование на языке Си: учебное пособие / Р.Ю. Царев. - Красноярск: Сиб. федер. ун-т, 2014. - 104 с.

# **Приложение**

Исходный код программного продукта «Производственный календарь»:

#include «stdafx.h»

#include <iostream>

void null (int x[])

{[0] = 3;[1] = 1;[2] = 7;[3] = 5;[4] = 6;[5] = 5;[6] = 3;

}

dayWeek (int g, int x[], int r)

{i;= g - 1970;(i = 0; i < 7; i++)[i] = x[i] + g + r;(i = 0; i < 7; i++)

if (x[i] > 6)[i] = x[i]% 7;

}

void weekend (int y[], char c[7] [33], char p[7] [9], int x[])

{

int i;(i = 0; i < 7; i++)

{(i == 2)

{(x[i] == 6)(«Выпал на выходной -%d % s, перенесен на понедельник % d % s и

на вторник % d % s\n», y[i], c[i], y[i] + 2, p[i], y[i] + 3, p[i]);

if (x[i] == 0)(«Выпал на выходной -%d % s, перенесен на понедельник % d % s и

на вторник % d % s\n», y[i], c[i], y[i] + 1, p[i], y[i] + 2, p[i]);

}

{(x[i] == 6)(«Выпал на выходной -%d % s, перенесен на понедельник % d

%s\n», y[i], c[i], y[i] + 2, p[i]);

if (x[i] == 0)(«Выпал на выходной -%d % s, перенесен на понедельник % d

%s\n», y[i], c[i], y[i] + 1, p[i]);

}

}

}

numberHolidays (int x[])

{

int kol, i;(kol = 0, i = 0; i<7; i++)

if ((x[i] == 6) || (x[i] == 0))++;(«Количество праздников, которое совпадает с выходными =%d\n»,);

}

menu()

{(«\nВыберите один из пунктов меню:\n1.ввод данных\n2.праздники,

которые

выпали на выходные\n3. Количество праздников, которое совпадает с

выходными\n4.очистить экран\n5.вывести все данные\n6.записать данные в

файл\n7.считать данные из файла\n8.о программе\n0.выход\n»);

}

print (int g)

{

system («cls»);();(g!= 0)

printf («%d\n», g);

}

input()

{g, k = 1;

while(k)

{(«Введите год с 2005 по 2099\n»);

scanf\_s («%d», &g);((g > 2004) && (g < 2100))

{= 0;(g);

}(«Ошибка, введенные данные не соответствуют заданным

значениям\n»);

}g;

}

void fpr (int x[], int g)

{i;\* fp;

fopen\_s (&fp, «file.txt», «wb»);

if (fp)

{

for (i = 0; i<7; i++) (&x[i], sizeof (x[i]), 1, fp);(&g, sizeof(g), 1, fp);

}(fp);

}fsc (int x[])

{i = 0, g = 0;\* fp;\_s (&fp, «file.txt», «rb»);(fp)

{(i!= 7)

{((fread (&x[i], sizeof (x[i]), 1, fp)) == 1)++;

{(«Ошибка, файл не имеет данных для считывания, либо ониодержат недопустимые значения\n»);

return 0;

}((i == 7) && ((fread (&g, sizeof(g), 1, fp))!= 1))

{(«Ошибка, файл не имеет данных для считывания, либо ониодержат недопустимые значения\n»);

return 0;

}

}

}

fclose(fp);

printf («Считывание данных из файла произошло успешно.\n»);

return g;

}

int leapYear (int g)

{

int r = 0, v = 1972;

r = ((g - v) / 4)+1;

return r;

}

about()

{(«Электронное приложение производственный календарь, которое по

указанному году определяет количество совпадений основных праздников»

«с выходными днями, выводит даты этих дней, а также даты на которые будут

перенесены выходные.\nРабота была выполнена: Бормисов Д.В., группа 606-41, 2016

год.\n»);

}

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

{

setlocale (LC\_ALL, «rus»);pu=1, r, g=0, v=0, x[7] = {3,1,7,5,6,5,3}, y[7] = {7,23,8,1,9,12,4};

char c[7] [33] = {«января рождество», «февраля день защитника

отечества», «марта женский день», «мая праздник весны и труда», «мая день

победы», «июня день России», «ноября день народного единства»};p[7] [9] = {«января», «февраля», «марта», «мая», «мая», «июня», «ноября»};();(pu)

{

scanf\_s («%d», &pu);(pu)

{

case 1:(g);(x);= input();= leapYear(g);(g, x, r); break;2:((g > 2004) && (g < 2100))(y, c, p, x);

else(«Ошибка, вы не ввели данные\n»); break;

case 3: ((g > 2004) && (g < 2100))(x);

else(«Ошибка, вы не ввели данные\n»); break;

case 4:(g); break;5:

if ((g > 2004) && (g < 2100))

{

printf («Год -%d\n», g);(y, c, p, x);

numberHolidays(x);

}(«Ошибка, вы не ввели данные\n»); break;6:((g > 2004) && (g < 2100))

{(x, g);(«Запись данных в файл произошла успешно.\n»);

}(«Ошибка, вы не ввели данные\n»); break;

case 7:= fsc(x);= leapYear(g);;

case 8:();(«pause»);

print(g);;

}

}0;

}