

МИНОБРАЗОВАНИЯ РОССИИ

Федеральное государственное бюджетное образовательное  
учреждение высшего образования

«Вологодский государственный университет»  
(ВоГУ)

**Компьютерные технологии**

Методические указания к самостоятельной работе.  
(часть № 1)

**Направление подготовки:** 13.03.02 – ЭЛЕКТРОЭНЕРГЕТИКА И  
ЭЛЕКТРОТЕХНИКА

**Профиль подготовки:** электроснабжение

**Квалификация (степень) выпускника:** бакалавр

**Форма обучения:** заочная/ускоренная

**Факультет:** электроэнергетический

**Кафедра:** электроснабжение

Вологда

2018

## **О самостоятельной работе в рамках СРС.**

В 2011 году в России введены в действие федеральные государственные образовательные стандарты (ФГОС) высшего профессионального образования (ВПО). Эти стандарты призваны стать «проводниками» перспективных отечественных, международных и европейских тенденций реформирования и развития высшего образования, исходя из стратегических интересов и культурно-образовательных традиций России. В концептуальные основания стандартов ВПО нового поколения вошли важные отличительные признаки-идеи, отражающие связь проектируемых новых норм для отечественной высшей школы с ведущими общемировыми тенденциями в развитии высшего образования и придающие новым российским образовательным стандартам и программам «международное измерение». Одним из таких основных отличительных признаков является возрастание ответственности преподавателей и студентов за эффективность образовательного процесса и собственной деятельности.

Самостоятельная работа студентов — планируемая учебная, учебно-исследовательская или научно-исследовательская работа студентов, выполняемая во внеаудиторное (или аудиторное) время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия (при частичном непосредственном участии преподавателя, оставляющем ведущую роль и контроль за работой студентов). Самостоятельная работа студентов - это вид учебно-познавательной деятельности, состоящей в индивидуальном, распределенном во времени выполнении студентами комплекса усложняющихся заданий при консультационно-координирующей помощи преподавателя, ориентированной на самоорганизацию деятельности обучающихся в условиях содержательно-смыслового структурирования их личностного времени. Исходя из изложенного выше, можно сформулировать основные цели и задачи самостоятельной работы студентов. Основная цель самостоятельной работы студентов состоит в овладении фундаментальными знаниями, профессиональными умениями и навыками деятельности по профилю, опытом творческой, исследовательской деятельности

Задачами организации самостоятельной работы студентов являются:

- Развитие способности работать самостоятельно, формирование самостоятельности мышления и принятия решений.
- Развитие активности и познавательных способностей студентов, развитие исследовательских умений
- Стимулирование самообразования и самовоспитания
- Развитие способности планировать и распределять свое время Кроме того, эта самостоятельная работа неразрывно связана с формированием таких важных компетенций, как способность применять знания на

практике и способность находить, обрабатывать и анализировать информацию из разных источников. Среди функций самостоятельной работы студентов в общей системе обучения выделяют следующие:

- Развивающая (повышение культуры умственного труда, приобщение к творческим видам деятельности, формирование интеллектуальных способностей студентов);
- Информационно-обучающая
- Стимулирующая (формирование мотивов образования, самообразования)
- Воспитывающая (формирование личностно-профессиональных качеств специалиста)

В данном случае, реализация самостоятельной работы выполнена через метод заданий. Для реализации этого метода важно выбрать тему, взятую из реальной жизни, значимую для студента, для решения которой необходимо приложить имеющиеся у него знания и новые знания, которые еще предстоит получить. Выбор темы преподаватель и студент осуществляют совместно, раскрывают перспективы исследования, вырабатывают план действий, определяют источники информации, способы сбора и анализа информации. В процессе исследования преподаватель опосредованно наблюдает, дает рекомендации, консультирует. После завершения выполнения задания студент участвует в оценке своей деятельности. Форма проверки работы – письменная и устная.

## **ВВЕДЕНИЕ**

Принимая во внимание огромное влияние современных информационных технологий на процесс образования, многие педагоги все с большей готовностью включают их в свою методическую систему. Проникновение современных информационных технологий в сферу образования позволяет педагогам качественно изменить содержание, методы и организационные формы обучения. Целью внедрения этих технологий в образовании являются гуманизация, индивидуализация, интенсификация процесса обучения и повышение качества обучения на всех ступенях образовательной системы.

Под информационными и коммуникационными технологиями предлагается понимать комплекс объектов, действий и правил, связанных с подготовкой, переработкой и доставкой информации при персональной, массовой и производственной коммуникации, а также все технологии и отрасли, интегрально обеспечивающие перечисленные процессы.

В последние годы термин «информационные технологии» часто выступает синонимом термина «компьютерные технологии», так как все информационные технологии в настоящее время так или иначе связаны с применением компьютера. Однако, термин «информационные технологии» намного шире и включает в себя «компьютерные технологии» в качестве составляющей. При этом информационные технологии, основанные на использовании современных компьютерных и сетевых средств, образуют термин «Современные информационные технологии».

### **1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

Целями освоения дисциплины «Компьютерные технологии»:

1.– теоретическое и практическое освоение компьютерных и информационных технологий сбора, обработки и анализа фактического материала для научных исследований и повседневной работы;

2. Знание основ построения логических схем и работы логических элементов.

3. – закрепление представлений о легитимности и корректности использования ресурсов глобальной компьютерной сети в научной и творческой деятельности.

4. Получение основных навыков о современных компьютерных технологиях, сетевых технологиях, языках веб программирования.

## 2. ЯЗЫК HTML

HTML (HyperText Markup Language, язык гипертекстовой разметки) - специальные инструкции браузеру, с помощью которых создаются Веб-страницы.

Т.е. Web-страницы - это документы в формате HTML, содержащие текст и специальные тэги (дескрипторы) HTML. По большому счету тэги HTML необходимы для форматирования текста (т.е. придания ему нужного вида), который "понимает" браузер. Документы HTML хранятся в виде файлов с расширением .htm или .html.

Тэги HTML сообщают браузеру информацию о структуре и особенностях форматирования Веб-страницы. Каждый тэг содержит определенную инструкцию и заключается в угловые скобки  $\langle \rangle$ . Большинство тэгов состоят из открывающей и закрывающей частей и воздействуют на текст, заключенный внутри.

Для более быстрого знакомства с HTML-кодом советую просматривать понравившуюся вам страницу в режиме "В виде HTML". Для этого в браузере надо выбрать пункт меню "Вид" - "В виде HTML".

После этого откроется новое окно с исходным текстом HTML-кода. Просматривая готовый код HTML, вы узнаете, как тэги языка применяются более опытными программистами. Кроме того, это один из способов преодоления различных проблем, возникающих время от времени.

Что такое Веб-страница? По сути дела - это простой текстовый документ, содержащий тэги (которые в свою очередь являются обычным текстом, заключенным в скобки). Т.е., исходя из вышеизложенного, Веб-страницы можно набрать в любом текстовом редакторе (Блокнот, WordPad, Word и т.д.). В этом случае абсолютно все надо набирать своими ручками. Основной плюс такого метода - код получается абсолютно минимизированным, т.е. "вес" такой html-страницы будет минимальным. К

недостаткам следует отнести тот факт, что пользователь должен очень хорошо знать язык html-разметки, да и результаты своего труда нельзя будет сразу посмотреть. Поэтому таким методом создания Вэб-страниц практически никто не пользуется. Пореккомендовать его можно разве что для создания простеньких Вэб-страничек если нет возможности использования специализированных программ.

Практически все Веб-дизайнеры используют специальные программы, которые называются HTML-редакторы. Наиболее популярные - Macromedia HomeSite, Dreamweaver MX. Чем же они хороши? Прежде всего, тем, что они как раз и предназначены для создания Веб-страниц. Синтаксис в таких программах обычно выделяется разными цветами, так что очень хорошо видно где находится текст, а где дескрипторы, а где php-код. Многие сложные конструкции тэгов можно вводить одним щелчком мышки. Результат своего труда виден тут же, в соседнем окошке. Если вы серьезно занимаетесь вэб-дизайном, то без такой программы вам просто не обойтись.

Кроме вышеуказанных программ существует еще один тип специализированных программ - это, так называемые, средства визуального проектирования (Microsoft FrontPage, HoTMetal Pro). Они позволяют создавать Веб-страницы таким образом, чтобы полностью исключить необходимость набора тэгов HTML вручную. Подобные программы выполняют все необходимые операции автоматически, по мере "рисования" Вэб-страницы. К недостаткам таких программ следует отнести тот факт, что "вес" таких страниц оказывается большим из-за избыточного HTML-кода, да и изящную страничку без подправки кода "вручную" сделать весьма проблематично.

### **Основные теги:**

<HEAD> и </HEAD>. Между этими тегами располагается информация о документе.

<TITLE> и </TITLE>. В этих тегах заключается название странички, которое будет выведено в рамке окна программы просмотра.

<BODY> и </BODY>. "Тело" документа (текст, графика и т.д.) располагается между этими двумя тегами.

Параметры тега <BODY>:

BGCOLOR - цвет фона (<BODY BGCOLOR="#000000">)

BACKGROUND - бэкграунд

TEXT - цвет текста

LINK - цвет ссылки

VLINK - цвет ссылки, уже посещенной в прошлом

ALINK - цвет активной ссылки

<P> и </P> - теги, служащие для выделения абзацев. Новый абзац всегда отделяется от предыдущего пустой строкой.

<BR> - тег, служащий для переноса текста на другую строку. Может также служить для отделения графики от текста на интервал.

<HR> - тег, служащий для логического разделения текста горизонтальной линией.

<PRE> и </PRE> - Между этими тегами располагается предварительно отформатированный текст. На экран он выводится шрифтом типа "курьер".

### **Параметры выравнивания**

Используются в <P> и <H>

ALIGN=LEFT - выравнивание по левому полю

ALIGN=RIGHT - выравнивание по правому полю

ALIGN=CENTER - выравнивание по центру

## Теги выравнивания

<LEFT> и </LEFT> - выравнивание по левому полю

<RIGHT> и </RIGHT> - выравнивание по правому полю

<CENTER> и </CENTER> - выравнивание по центру

## Заголовки, служащие для выделения логических частей текста

<H1> и </H1> - Заголовок первого уровня.

<H2> и </H2> - Заголовок второго уровня.

<H3> и </H3> - Заголовок третьего уровня.

<H4> и </H4> - Заголовок четвертого уровня.

<H5> и </H5> - Заголовок пятого уровня.

<H6> и </H6> - Заголовок шестого уровня.

## Теги для выделения текста и шрифта

<B> и </B> - теги для выделения текста (слов, букв) жирным шрифтом.

<I> и </I> - теги для выделения текста (слов, букв) курсивным шрифтом, типа *Italic*.

<U> и </U> - текст, расположенный между двумя этими тегами, будет подчеркнут.

<BLINK> и </BLINK> - текст, расположенный между двумя этими тегами, будет мигать.

<FONT SIZE=n>и</FONT> - теги для изменения размера шрифта (где n - размер шрифта в пикселях px).

<FONT COLOR="#FFFFFF"> и </FONT> - теги для изменения цвета шрифта.

Теги для формирования списков

<OL> и </OL> - теги, показывающие начало и конец нумерованного списка

<UL> и </UL> - теги, показывающие начало и конец маркированного



списка.

<LI> - Элемент списка

<DL> и </DL> - теги, показывающие начало и конец глоссария.

<DT> - Термин глоссария, располагается без отступа от левого поля страницы.

<DD> - Описание термина, располагается с отступом от левого поля страницы.

### **Теги-команды для вставки в текст объектов не текстовой информации**

<IMG SRC="file.gif"> или <IMG SRC="file.jpg"> - команда для вставки графического изображения

<IMG SRC="file.wav"> - команда для вставки звукового фрагмента

<IMG SRC="file.avi"> - команда для вставки видео фрагмента

### **Параметры графического изображения**

WIDTH - ширина картинка в пикселях

HEIGHT - высота картинка в пикселях

ALIGN - выравнивание (ALIGN=LEFT - выравнивание по левому полю, ALIGN=RIGHT - по правому полю, ALIGN=TOP - по верхней границе, ALIGN=BOTTOM - по нижней границе, ALIGN=MIDDLE или CENTER - по центру)

HSPACE - горизонтальный отступ от графического изображения

VSPACE - вертикальный отступ

ALT - альтернативный текст, служит для обозначения изображения

### **Команды, служащие для гиперсвязи с другими HTML-документами и ресурсами Интернет**

<A HREF="fail.htm">и</A>или<A HREF="http://www.ru">и</A> - гиперсвязи

<ADDRESS><A

HREF="mailto:nick@mail.ru">nick@mail.ru</ADDRESS> - гиперсвязь с адресом электронной почты

## **Таблицы**

Таблица - сетка для показа данных в строках и столбцах, а также средство для форматирования текста

<TABLE> и </TABLE> - теги для вставки таблицы в HTML документ

Параметры тега <TABLE>

BGCOLOR - цвет фона

BORDER - ширина бордюра

WIDHT - ширина таблицы

## **Теги разметки таблицы**

<CAPTION> и </CAPTION> - название таблицы, имеет параметр ALIGN=TOP - выравнивание над таблицей и ALIGN=BOTTOM - под таблицей.

<TR> и </TR> - Строчка таблицы. Может иметь параметры BGCOLOR - цвет фона внутри строки;

ALIGN=LEFT, RIGHT, CENTER - выравнивание внутри строки;

VALIGN=TOP, BOTTOM, MIDDLE - вертикальное выравнивание внутри строки таблицы.

<TD> и </TD>- Столбец таблицы. Может иметь параметры BGCOLOR - цвет фона под столбцом;

ALIGN=LEFT, RIGHT, CENTER - выравнивание внутри столбца;

VALIGN=TOP, BOTTOM, MIDDLE - вертикальное выравнивание;

COLSPAN растягивание клетки на несколько столбцов, ROWSPAN - растягивание клетки на несколько строк.

<TH> и </TH> - Заголовок столбца. Может иметь параметры BGCOLOR - цвет фона под названием;

ALIGN=LEFT, RIGHT, CENTER - выравнивание;  
VALIGN=TOP, BOTTOM, MIDDLE - вертикальное выравнивание;  
COLSPAN, ROWSPAN - растягивание клетки на несколько столбцов  
или строк; WIDTH - ширина названия.

### **Рамки-фреймы**

Рамки-фреймы - средство для деления экрана на несколько областей, в каждой из которых отображается содержимое отдельной Web-страницы или Web-сайта.

<FRAMESET> и </FRAMESET> теги для создания рамки

Параметры тега <FRAMESET>

COLS - подразделяют экран на определенное количество колонок  
(вертикальных)

ROWS - подразделяют экран на определенное количество колонок  
(горизонтальных)

BORDCOLOR - цвет рамки

BORDER - ширина бордюра

FRAMEBORDER - граница рамки (FRAMEBORDER=YES - есть граница, FRAMEBORDER=NO - нет границы, FRAMESPACING=n - ширина границы)

<FRAME> Тег для описания рамки (<FRAME SRC="file.htm">).

### **Параметры тега <FRAME>**

SCROLLING - параметр для регулировки полосы прокрутки:

SCROLLING=YES - полоса прокрутки будет всегда

SCROLLING=NO - полосы прокрутки не будет

SCROLLING=AUTO - полоса прокрутки появляется только в случае  
необходимости

MARGINWIDTH и MARGINHEIGHT - параметры, которые  
управляют отступом внутри рамок, служат для выравнивания графического

изображения внутри рамки

NORESIZE - параметр, указывающий на то, что размер рамки-фрейма никогда не будет меняться.

A link to <A HREF="file.htm" TARGET="frame2"> file.htm</A> - Связь между фреймами

TARGET - атрибут связи между фреймами. Имеет несколько значений:

\_BLANK загружает содержимое страницы, заданной ссылкой, в новое пустое окно.

\_SELF содержимое страницы, заданной ссылкой, в окно, которое содержит ссылку.

\_PARENT загружает содержимое страницы, заданной ссылкой, в окно, являющееся непосредственным владельцем набора фреймов.

\_TOP содержимое страницы, заданной ссылкой, в окно, игнорируя используемые фреймы.

Обработка браузеров, не поддерживающих фреймы:

<FRAMESET>

...Здесь располагаются фреймы

</FRAMESET>

<NOFRAMES>

<BODY>

...Здесь располагается текст без фреймов

<BODY>

</NOFRAMES>

### 3. ЗАДАНИЕ № 1

#### Вариант 0

Написать текст простой веб страницы на языке HTML, в которой будет размещен рисунок и ссылка на файл.

#### Вариант 1

Написать текст простой веб страницы на языке HTML выводящую на экран таблицу 5x5, каждая ячейка которой заполнена картинками.

#### Вариант 2

Написать текст простой веб страницы на языке HTML содержащую ссылки на 5 файлов разных форматов.

#### Вариант 3

Написать текст простой веб страницы на языке HTML которая бы содержала таблицу 2x2, в ячейке 1.1 находилась бы картинка, в ячейке 1:2 ссылка на файлы.

#### Вариант 4

Написать текст простой веб страницы на языке HTML которая бы выводила на экран слово «Электроснабжение» в трех разных размерах шрифта, являющееся ссылкой на файл.

#### Вариант 5

Написать текст простой веб страницы на языке HTML в которой содержались бы ссылки на поисковые системы yandex (ya.ru), google (google.com), mail.ru (mail.ru)

#### Вариант 6

Написать текст простой веб страницы на языке HTML в которой будет размещен рисунок и ссылки на поисковые системы yandex (ya.ru), google (google.com), mail.ru (mail.ru)

#### Вариант 7

Написать текст простой веб страницы на языке HTML которая

бы последовательно выводила на экран 2 таблицы размером 3x3

#### Вариант 8

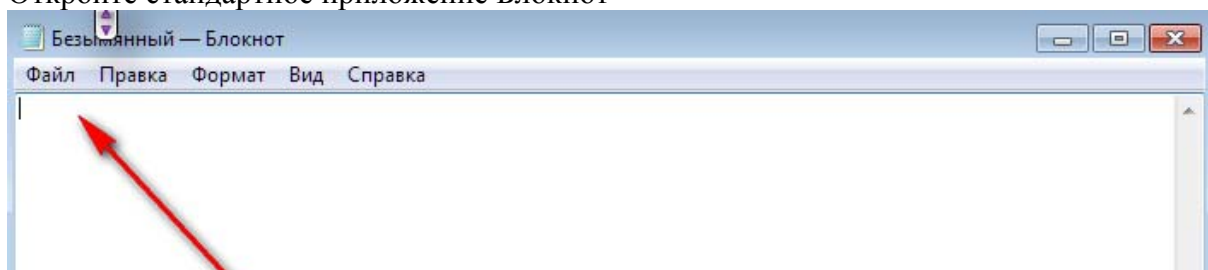
Написать текст простой веб страницы на языке HTML, которая выведет на экран три картинки размером 200x200 пикселей.

#### Вариант 9

Написать текст простой веб страницы на языке HTML выводящей на экран 2 рисунка размером 300x300, одновременно являющимися ссылками на файлы.

### Порядок выполнения работы

Откройте стандартное приложение Блокнот

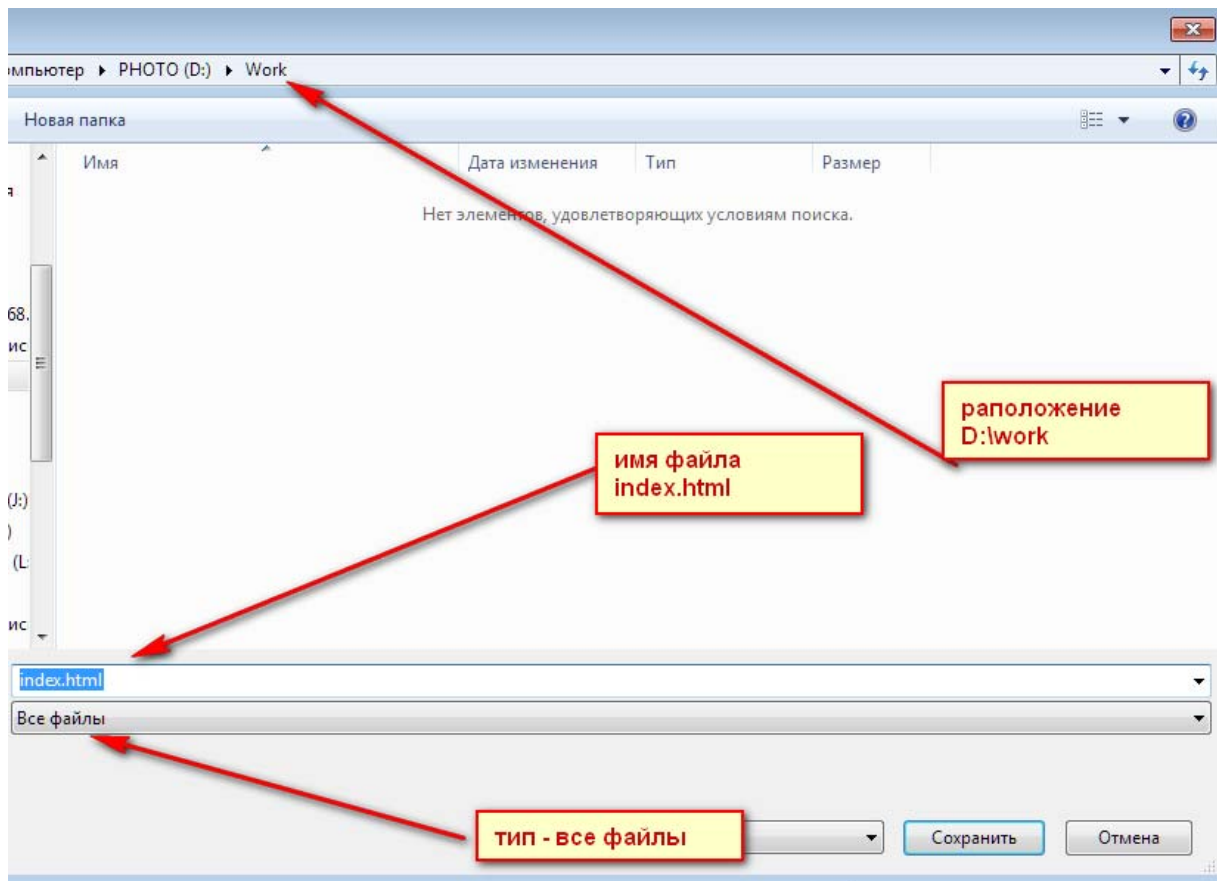


Выберите Файл → Сохранить как

Тип файла – все файлы.

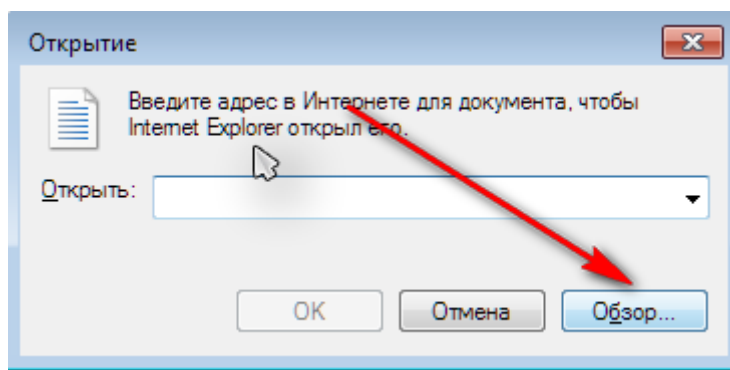
Место – D:\work

Имя файла : index.html



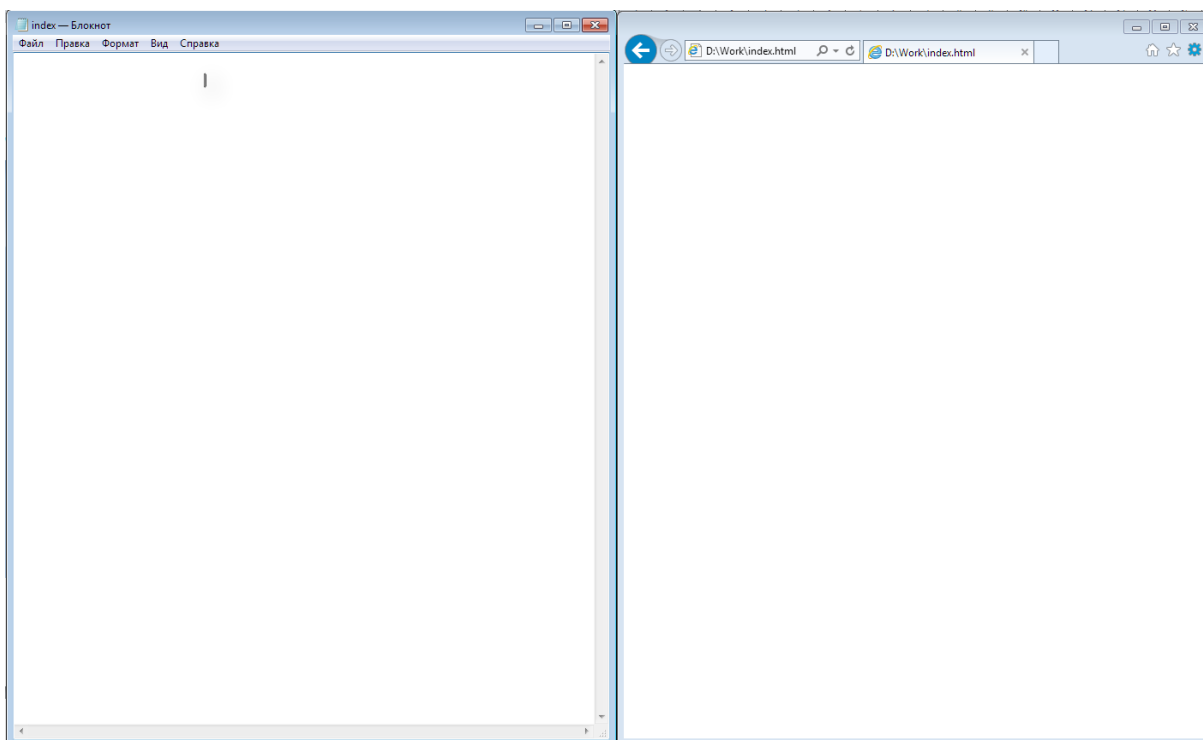
Сохраните страницу с этим именем

Откройте браузер. Выберите “открыть страницу” или нажмите Ctrl+O



Укажите файл D:\work\index.html

Расположите окна на экране так, что бы было видно два окна



Наберите код простой страницы и отобразите его в браузере.

Скопируйте текст и скриншот в отчет.

#### 4. ЗАДАНИЕ № 2

##### Создание страницы – резюме

Использование формата веб-резюме позволяет людям разных специальностей найти работу, используя современные информационные технологии, которые наглядно позволят показать все ваши достоинства как работника и оперативно донести эту информацию до максимального количества работодателей. При этом основной целью ставится индивидуальный подход к каждому соискателю, путем донесения информации до работодателя через личный веб-сайт с размещением информации в текстовом, графическом и видео- формате.

Предлагаемая форма поиска работы более эффективна, в отличии от использования шаблонных форм-резюме на job-сайтах, так как работодатель в данный момент рассматривает только вашу кандидатуру, а не поглядывает на длинный список резюме на порталах о трудоустройстве, в результате чего может остановиться не на самой оптимальной кандидатуре. Среди преимуществ веб-резюме перед стандартным резюме, можно выделить следующие:



- У Вас появиться больше шансов быть замеченными.
- Вы используете новые технологии поиска работы, что говорит о Вас как о современном человеке.
- Ваше резюме сможет увидеть больше потенциальных работодателей, например выпускающих предизолированные трубы.
- Информация о Вас, будет доступна в сети Интернет неограниченный период времени, в любое время суток.
- Вы не зажаты рамками стандартного резюме и можете предоставить больше информации в оригинальной форме (краткая характеристика, автобиография, квалификационные документы, достижения в трудовой деятельности, Ваши проекты, разработки и т.д.).
- Вы значительно экономите время на поиск работы, не публикуя свое резюме на десятках сайтов по трудоустройству.

Персональное web-resume позволяет значительно расширить объем информации о Вас. Например, только использование вашего фото увеличивает шанс найти работу в четыре раза, а если разместить фото или видео выполненных вами объектов, изделий, приложить сканы документов или чертежей, с которыми вы работали, то работодатель обязательно свяжется с Вами и пригласит на работу. Кроме этого о Вас узнает значительно больше потенциальных работодателей, так как ваша персональная веб-страница проиндексируется большинством поисковых систем и интернет-каталогов и не потеряет свою актуальность в течении длительного времени, в отличии от стандартного резюме на порталах по поиску работы, которое могут удалить или переместить в течении нескольких дней на более низкие позиции. При этом, использование сайта-резюме вовсе не исключает и более принятый способ размещения своего резюме на любом job-сайте, вы можете там оставить ссылку на свою персональную страничку.

Создайте собственную страницу – резюме на языке HTML и скопируйте её в отчет в исходном коде HTML. Страница должна содержать как минимум 5 блоков (информация о соискателе, опыт работы, обучение, личные качества, другое)

## 5. ГРАФИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ АЛГОРИТМОВ

Графический способ представления алгоритмов является более компактным и наглядным по сравнению со словесным. При графическом представлении алгоритм изображается в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению одного или нескольких действий.

Такое графическое представление называется схемой алгоритма или блок-схемой. В блок-схеме каждому типу действий (вводу исходных данных, вычислению значений выражений, проверке условий, управлению повторением действий, окончанию обработки и т.п.) соответствует геометрическая фигура, представленная в виде блочного символа. Блочные символы соединяются линиями переходов, определяющими очередность выполнения действий.

Название символа	Обозначение и пример заполнения	Пояснение
Процесс		Вычислительное действие или последовательность действий
Решение		Проверка условий
Модификация		Начало цикла
Предопределенный процесс		Вычисления по подпрограмме, стандартной подпрограмме
Ввод-вывод		Ввод-вывод в общем виде
Пуск-останов		Начало, конец алгоритма, вход и выход в подпрограмму
Документ		Вывод результатов на печать

Блок "процесс" применяется для обозначения действия или последовательности действий, изменяющих значение, форму представления или размещения данных. Для улучшения наглядности схемы несколько отдельных блоков обработки можно объединять в один блок. Представление отдельных операций достаточно свободно.

Блок "решение" используется для обозначения переходов управления по условию. В каждом блоке "решение" должны быть указаны вопрос, условие или сравнение, которые он определяет.

Блок "модификация" используется для организации циклических конструкций. (Слово модификация означает видоизменение, преобразование). Внутри блока записывается параметр цикла, для которого указываются его начальное значение, граничное условие и шаг изменения значения параметра для каждого повторения.

Блок "предопределенный процесс" используется для указания обращений к вспомогательным алгоритмам, существующим автономно в виде некоторых самостоятельных модулей, и для обращений к библиотечным подпрограммам.

## 6. ЗАДАНИЕ № 3

Нарисуйте линейный алгоритм и составьте текст программы на любом, известном вам языке программирования. Прокомментируйте текст.

### Вариант №1

Сформировать строку из 10 символов. На четных позициях должны находиться четные цифры, на нечетных позициях - буквы.

### Вариант № 2

Дана строка. Определите, какой символ в ней встречается раньше: 'x' или 'w'. Если какого-то из символов нет, вывести сообщение об этом.

### Вариант № 3

Удалите в строке все буквы 'x'. за которыми следует 'abc'.

### Вариант № 4

Дан текст. Найти слова, состоящие из цифр, и сумму чисел, которые образуют эти слова.

#### Вариант № 5

Даны две строки. Определите, содержится ли меньшая по длине строка в большей.

#### Вариант № 6

Вывести слова, в которых заменить каждую большую букву одно-именной малой; удалить все символы, не являющиеся буквами или цифрами; вывести в алфавитном порядке все гласные буквы, входящие в каждое слово строки.

#### Вариант № 7

Вывести в алфавитном порядке все слова, содержащие наибольшее количество гласных букв; найти все слова, в которые буква «а» входит не менее двух раз.

#### Вариант № 8

Строка состоит из слов, разделенных одним или несколькими пробелами. Поменяйте местами наибольшее по длине слово и наименьшее.

#### Вариант № 9

Дана строка, в которой нет начальных и конечных пробелов. Необходимо изменить ее так, чтобы длина строки стала равна заданной длине, больше чем текущая длина строки. Это следует сделать путем вставки между словами дополнительных пробелов. Количество пробелов между отдельными словами не должно отличаться более чем на один пробел (то есть пробелы добавляются равномерно).

### Вариант № 10

Дана строка, содержащая буквы и скобки '(', ')', '[, ]', '{, }'. Если скобки расставлены правильно (то есть каждой открывающей скобки соответствует закрывающая того же вида), то вывести 'yes', иначе вывести номер первой ошибочной скобки или -1, если закрывающая скобка отсутствует

### Вариант № 11

Дан текст. Некоторые его фрагменты выделены группами символов ##. Заменить выделение группами символов '<' и '>'. Пример: 'Это ##тестовый пример## для задачи ##на## строки' преобразуется в 'Это <тестовый> пример для задачи <на> строки'

### Вариант № 12

В сообщении, состоящем из одних русских букв и пробелов, каждую букву заменили ее порядковым номером в русском алфавите (А — 1, Б — 2, ..., Я — 33), а пробел — нулем. Требуется по заданной последовательности цифр найти количество исходных сообщений, из которых она могла получиться.

### Вариант № 13

Проверить, соблюдается ли в заданном тексте баланс открывающих и закрывающих круглых скобок, то есть можно ли установить взаимно однозначное соответствие открывающих и закрывающих скобок, причем открывающая скобка всегда предшествует соответствующей закрывающей.

### Вариант № 14

Написать программу получения строки, в которой удалены все «лишние» пробелы, то есть из нескольких подряд идущих пробелов оставить только один.

### Вариант № 15

Написать программу, которая преобразует введенное число (целое или вещественное) в соответствующую строку символов.

### Вариант № 16

Дан массив. Найдите два соседних элемента, сумма которых минимальна.

### Вариант № 17

Найдите количество чисел, каждое из которых равно сумме квадратов своих соседей и при этом не является наибольшим в массиве.

### Вариант № 18

Напишите программу, которая вводит с клавиатуры непустой массив целых чисел, и выводит число локальных максимумов (элемент является локальным максимумом, если он не имеет соседей, больших, чем он сам).

### Вариант № 19

Найти наибольший четный элемент массива и поменять его местами с наименьшим нечетным элементом. Если одного из таких элементов нет, то всем элементам массива присвоить значение, равное нулю.

### Вариант № 20

Найти в массиве все серии подряд идущих одинаковых элементов и удалить из них все элементы кроме одного.

### Вариант № 21

Преобразовать массив таким образом, чтобы сначала располагались все элементы, модуль которых не превышает единицу, а потом – все остальные.

#### Вариант № 22

В данном массиве найти серию подряд идущих элементов наибольшей длины, в которой первое число равно последнему, второе - предпоследнему и так далее.

#### Вариант № 23

Пользователь вводит цены 1 кг конфет и 1 кг печенья. Найдите стоимость: а) одной покупки из 300 г конфет и 400 г печенья; б) трех покупок, каждая из 2 кг печенья и 1 кг 800 г конфет.

#### Вариант № 24

Пользователь вводит количество дней, указывает процент скидки и вводит сумму. Рассчитать прибыль, если за каждый день сумма увеличивается на 3 \$ и затем применяется скидка, то есть итоговая сумма еще увеличивается на данное число процентов.

#### Вариант № 25

Из трехзначного числа  $x$  вычли его последнюю цифру. Когда результат разделили на 10, а к частному слева приписали последнюю цифру числа  $x$ , то получилось число 237. Найти число  $x$ .

#### Вариант № 26

Дано число. Если оно от 2 до 5 включительно, то увеличить его на 10. Если оно от 7 до 40, то уменьшить на 100. Если оно не более 0 или более 3000, то увеличить в 3 раза (то есть умножить на 3). Иначе занулить это число.

#### Вариант № 27

Дано две даты, каждая из которых состоит из трех чисел (день, месяц и год). Вывести yes, если первая дата раньше второй, иначе вывести no.

### Вариант № 28

Дано четырехзначное число. Если оно читается слева направо и справа налево одинаково, то вывести yes, иначе no.

### Вариант № 29

Вывести на экран числа от 1000 до 9999 такие, что все цифры различны.

### Вариант № 30

Реализуйте серию из  $n$  игр "Камень, ножницы, бумага" с компьютером. В результате выведите статистику: сколько игр выиграл пользователь, сколько раз каждого вида ходов было выбрано. Дополните игру анализом компьютера ваших ходов и выбор наиболее подходящего против вас хода.

### Вариант № 31

Сгенерируйте серию из 10 случайных чисел от 1 до 3 и найдите: а) на сколько количество двоек больше/меньше количества троек, б) количество троек, стоящих на четных местах, в) количество двоек среди первых пяти чисел серии.

### Вариант № 32

Компьютер загадывает число от 1 до 100. У пользователя три попытки отгадать. После каждой неудачной попытки компьютер сообщает меньше или больше загаданное число.



## Вариант № 33

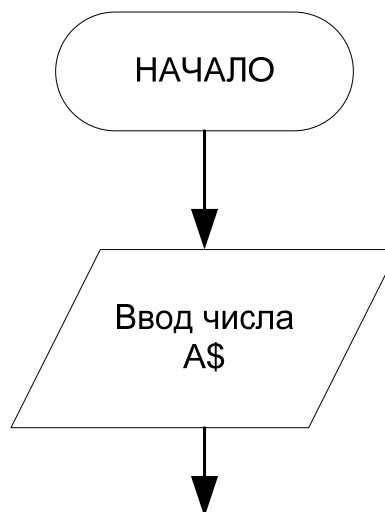
Пользователь вводит положительное целое число. Зашифровать каждую цифру серией из букв (конкретный принцип составления серии букв разработать самостоятельно).

### 7. ПОЯСНЕНИЯ К ВЫПОЛНЕНИЮ КОНТРОЛЬНОЙ РАБОТЫ

Контрольная работа по дисциплине выполняется со всеми требованиями к оформлению курсовых и выпускных работ на листах формата А4 рукописным или печатным способом. Чертежи допускается размещать на листах формата А3.

Выполнение каждого задания состоит из пояснительной части, в которой описывается методика решения задачи и графической части, где предложено схемотехническое решение.

Графический алгоритм выполняется с использованием стандартных элементов построения графических алгоритмов и диаграмм:



которые приведены ранее.

## 8. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

№ п/п	Библиографическое описание по ГОСТ	Кол-во экземпляров в библиотеке ВоГУ	Наличие литературы на кафедре и других библиотеках
1	<b><u>Основная</u></b> Цифровая и импульсная электроника // Лачи, В. И. Электроника / В. И. Лачин, Н. С. Савелов. - Ростов н/Д, 2000. - С. 289-424.	4	
2	Теоретические основы электротехники. Общая электротехника и электроника : расчет и анализ линейных электрических цепей в установившемся режиме: метод. пособие по выполнению контрол. заданий: ФЗДО: специальности 140201, 220201/ сост.: В. В. Реутов, Г. Л. Ганичев. - Вологда: ВоГТУ, 2011. - 51, [1] с.	14	
3	Лачин, В. И. Электроника: учеб. пособие для вузов по направлению "Автоматизация и упр."/ В. И. Лачин, Н. С. Савелов. - Изд. 8-е. - Ростов н/Д: Феникс, 2010. - 703 с.:	1	
4	Красников, Г. Я. Нанoeлектроника: состояние, проблемы и перспективы развития / Г. Я. Красников, Н. А. Зайцев // Нано- и микросистемная техника. - 2009. - № 1. - С. 2-5.		
5	Эмиссионная электроника / Н. Н. Коваль, Е. М. Окс, Ю. С. Протасов, Н. Н. Семашко; под ред. Ю. С. Протасова. - М.: МГТУ, 2009. - 595 с.	1	
6	Гальперин, М. В. Электротехника и электроника: учебник для СПО / М. В. Гальперин. - М.: ФОРУМ: ИНФРА-М, 2009. - 479 с.	1	
7	Элементы электронных схем. Полупроводниковые диоды. Транзисторы. Тиристоры. Усилители // Лачин, В. И. Электроника / В. И. Лачин, Н. С. Савелов. - Ростов н/Д, 2000. - С.8-157.	4	

8	Старкова, Л. Е. Основы электроники: учеб. пособие / Л. Е. Старкова. - Вологда: ВоГТУ, 2010. - 87 с.	31	
9	Майер, А. А. Оптические транзисторы для сверхбыстрой обработки и передачи информации / А. А. Майер // Электротехника. - 2010. - № 12. - С. 43-46.		
10	Волович, Г. И. Схемотехника аналоговых и аналого-цифровых электронных устройств/ Г. И. Волович. - 2-е изд. - М.: Додэка-XXI, 2007. - 527, [1] с.	1	
11	Евстифеев, А. В. Микроконтроллеры AVR семейств Tiny и Mega фирмы Atmel / А. В. Евстифеев. - 4-е изд. - М.: Додэка-XXI, 2007. - 557 с.	1	
12	Голубков, А. Программатор МК Atmel серии AT89 / А. Голубков // Радио. - 2003. - № 9. - С. 24-26.		
13	Гребнев, В. В. Микроконтроллеры семейства AVR фирмы Atmel / В. В. Гребнев. - М.: РадиоСофт, 2002. - 172 с.	1	
14	Алексеев, К. Б. Микроконтроллерное управление электроприводом: учеб. пособие для вузов / К. Б. Алексеев, К. А. Палагута. - М.: МГИУ, 2008. - 296 с.	5	
15	<b>Мельников, П. П.</b> Технология разработки HTML-документов : учеб. пособие по специальностям "Финансы и кредит", "Бухгалтер. учет, анализ и аудит" / П. П. Мельников . - М. : Финансы и статистика , 2005 . - 106, [3] с. : ил. . - (*)	2	
16	<b>Хоумер, А.</b> Dynamic HTML : справочник / Алекс Хоумер, Крис Улмен . - СПб. [и др.] : Питер , 2000 . - 510 с. : ил.	1	