



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.01 Информатика и вычислительная техника

О Т Ч Е Т

по лабораторной работе № 1

Дисциплина: Языки интернет-программирования

Студент

ИУ6-33Б

(Группа)


(Подпись, дата)

Д. И. Мироненко

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

В. Д. Шульман

(И.О. Фамилия)

Москва, 2023

Задание

1. Выполните разметку HTML-страницы с использованием элементов семантической разметки текста. Напишите не менее 10 строк произвольного текста. Выделите в каждой строке несколько слов как более важные, значительно важные, добавьте математические формулы и пр. Также, с использованием семантической разметки, добавьте к тексту произвольные строки кода на любом языке программирования. Подготовьте второй вариант разметки, отличающийся выделенными словами. Замените фрагменты, которые ранее отображались курсивом на фрагменты, отображаемые жирным шрифтом.
2. Сформируйте HTML-страницу с фрагментом расписания занятий (используйте элемент table). Выберите фрагмент расписания таким образом, чтобы хотя бы в одном месте возникала необходимость объединения ячеек таблицы.
3. Подготовьте разметку формы регистрации на произвольном сайте. Для расположения элементов используйте табличную верстку.
4. Используя валидаторы HTML (предустановленный в браузере или <https://validator.w3.org/>), проверьте полученные HTML-страницы на наличие ошибок. Составьте таблицу выявленных ошибок, в которую внесите все ошибки валидации и их фактические проявления в браузере. Устраните все найденные ошибки.

Часть 1

Код программы

Файл "index.html"

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Task 1</title>
    <link rel="preconnect" href="https://fonts.googleapis.com">
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
    <link
href="https://fonts.googleapis.com/css2?family=Inter:wght@400;700&family=JetBrains+Mono:wght@400;700&display=swap"
```

```

rel="stylesheet">
  <link rel="stylesheet" type="text/css" href="style.css">
  <script src="https://cdn.jsdelivr.net/gh/google/code-prettify@master/loader/run_prettify.js"></script>
</head>
<body>
  <div id="main">
    <h1><code class="prettyprint lang-cpp">std::vector</code></h1>
    <div class="block">
      <h2>Intro</h2>
      <p>
        The elements are stored <strong>contiguously</strong>, which means
        that elements can be accessed <em>not only through iterators</em>,
        but also using <strong>offsets</strong> to regular pointers
        to elements. This means that a pointer to an element of a vector
        <em>may be passed to any function</em> that expects a pointer
        to an element of an array.
      </p>
      <p>
        The storage of the vector is handled <strong>automatically</strong>,
        being expanded as needed. Vectors usually occupy more space than static
        arrays, because more memory is allocated to handle future growth. This
        way a vector <em>does not need to reallocate each time an element is
        inserted</em>, but only when the additional memory is exhausted. The
        total amount of allocated memory can be queried using
        <a href="https://en.cppreference.com/w/cpp/container/vector/capacity">
          <code>capacity()</code>
        </a>
        function. Extra memory <em>can be returned</em> to the system via a call to
        <a href="https://en.cppreference.com/w/cpp/container/vector/shrink_to_fit">
          <code>shrink_to_fit()</code>
        </a>
        .
      </p>
      <p>
        Reallocations are usually costly operations in terms of performance.
        The
        <a href="https://en.cppreference.com/w/cpp/container/vector/reserve">
          <code>reserve()</code>
        </a>
        function can be used to eliminate reallocations if the number of
        elements is known beforehand.
      </p>
      <p>
        The <strong>complexity</strong> (efficiency) of common operations
        on vectors is as follows:
      </p>
      <ul>
        <li>
          Random access - <em>constant</em>
          <math>
            \begin{aligned}
            &O(1) \\
            &O(n) \\
            &O(1) \\
            &O(n)
            \end{aligned}
          </math>

```

```
</math>
</li>
- Insertion or removal of elements at the end - <em>amortized constant</em>
    <math>
      \begin{aligned}
        O(1) \\
        O(n)
\end{aligned}

- Insertion or removal of elements - <em>linear in the distance to the end  
of the vector</em>
    <math>O(n)</math>

</ul>
</div>
<div class='block'>


## include <iostream>; include <ivector>; int main() { // Create a vector containing integers std::vector<int> v = {8, 4, 5, 9}; // Add two more integers to vector v.push_back(6); v.push_back(9); // Overwrite element at position 2 v[2] = -1; // Print out the vector for (int n : v) cout <&lt; n <&lt; " "; std::cout <&lt; endl; }

```

</div>
</body>
</html>

Файл “style.css”

```
body {
    background-color: #f2f1f6;
    color: #010100;
    font-size: 14px;
    font-family: 'Inter', sans-serif;
}

div#main {
    width: 50%;
    margin: auto;
}

div.block {
    margin-bottom: 40px;
    padding: 5px 40px 20px 40px;
    background-color: #ffffff;
    border-radius: 30px;
    text-align: justify;
    line-height: 1.5;
    font-size: 17px;

    -webkit-box-shadow: 0px 5px 10px 2px rgba(34, 60, 80, 0.2);
    -moz-box-shadow: 0px 5px 10px 2px rgba(34, 60, 80, 0.2);
    box-shadow: 0px 5px 10px 2px rgba(34, 60, 80, 0.2);
}

h1, h2, h3, h4, h5, h6 {
    font-weight: bold;
}

a {
    text-decoration: none;
}

code {
    font-family: 'JetBrains Mono', monospace;
    line-height: 0.5;
}

math {
    font-family: serif;
    font-style: italic;
    letter-spacing: 0px;
}
```

Работа сайта

std::vector

Intro

The elements are stored **contiguously**, which means that elements can be accessed *not only through iterators*, but also using **offsets** to regular pointers to elements. This means that a pointer to an element of a vector *may be passed to any function* that expects a pointer to an element of an array.

The storage of the vector is handled **automatically**, being expanded as needed. Vectors usually occupy more space than static arrays, because more memory is allocated to handle future growth. This way a vector *does not need to reallocate each time an element is inserted*, but only when the additional memory is exhausted. The total amount of allocated memory can be queried using `capacity()` function. Extra memory *can be returned* to the system via a call to `shrink_to_fit()`.

Reallocations are usually costly operations in terms of performance. The `reserve()` function can be used to eliminate reallocations if the number of elements is known beforehand.

The **complexity** (efficiency) of common operations on vectors is as follows:

- Random access - *constant $O(1)$*
- Insertion or removal of elements at the end - *amortized constant $O(1)$*
- Insertion or removal of elements - *linear in the distance to the end of the vector $O(n)$*

Example

```
#include <iostream>
#include <vector>

int main() {
    // Create a vector containing integers
    std::vector<int> v = {8, 4, 5, 9};

    // Add two more integers to vector
    v.push_back(6);
    v.push_back(9);

    // Overwrite element at position 2
    v[2] = -1;

    // Print out the vector
    for (int n : v)
        std::cout << n << ' ';
    std::cout << std::endl;
}
```

Рисунок 1

Проверка валидатором

Showing results for index.html

Checker Input

Show ☐ source ☐ outline ☐ image report

Check by No file chosen

Uploaded files with .xhtml or .xht extensions are parsed using the XML parser.

Document checking completed. No errors or warnings to show.

Used the HTML parser.
Total execution time 14 milliseconds.

Рисунок 2

Часть 2

Код программы

Файл “index.html”

```
<!DOCTYPE html>
<html lang="ru">
  <head>
    <meta charset="utf-8">
    <title>Task 2</title>
    <link rel="preconnect" href="https://fonts.googleapis.com">
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
    <link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@400;500;600;700&display=swap"
rel="stylesheet">
    <link rel="stylesheet" type="text/css" href="style.css">
  </head>
  <body>
    <div id="main">
      <h1>Расписание</h1>
      <div class="block">
        <h2 class="block-name">Вторник</h2>
        <table>
          <tbody>
            <tr>
              <th class="table-head">Время</th>
              <th class="table-head">ЧС</th>
              <th class="table-head">3Н</th>
            </tr>
            <tr>
              <td>8:30 - 10:05</td>
              <td></td>
              <td></td>
            </tr>
            <tr>
              <td>10:15 - 11:50</td>
              <td></td>
              <td></td>
            </tr>
            <tr>
              <td>12:00 - 13:35</td>
              <td></td>
              <td rowspan="2">(лаб) Языки интернет-программирования 805</td>
            </tr>
            <tr>
              <td>13:50 - 15:25</td>
              <td>(сем) Базы данных 255л<br>Фомин М. М.</td>
            </tr>
            <tr>
              <td>15:40 - 17:15</td>
              <td>(сем) Электротехника 619л<br>Ибрагимов С. В.</td>
              <td></td>
            </tr>
          </tbody>
        </table>
      </div>
    </div>
  </body>
</html>
```

```

        <tr>
            <td>17:25 - 19:00</td>
            <td></td>
            <td></td>
        </tr>
        <tr>
            <td>19:10 - 20:45</td>
            <td></td>
            <td></td>
        </tr>
    </tbody>
</table>
</div>
</div>
</body>
</html>

```

Файл “style.css”

```

body {
    font-family: 'Montserrat', sans-serif;
    background-color: #f2f1f6;
}

div#main {
    width: 60%;
    margin: auto;
}

div.block {
    margin-bottom: 40px;
    padding: 1px 20px 20px 20px;
    background-color: #ffffff;
    border-radius: 15px;

    -webkit-box-shadow: 0px 5px 10px 2px rgba(34, 60, 80, 0.2);
    -moz-box-shadow: 0px 5px 10px 2px rgba(34, 60, 80, 0.2);
    box-shadow: 0px 5px 10px 2px rgba(34, 60, 80, 0.2);
}

table {
    font-size: 14px;
    border-collapse: collapse;
    width: 100%;
    text-align: center;
    border-style: hidden;
    font-weight: 500;
    border: 1px solid black;
}

th, td {
    padding: 10px;
    border: 1px solid black;
}

```



```
}

.table-head {
    background-color: #e1e0e0;
}
```

Работа сайта

Расписание		
Вторник		
Время	ЧС	ЗН
8:30 - 10:05		
10:15 - 11:50		
12:00 - 13:35		(лаб) Языки интернет-программирования 805
13:50 - 15:25	(сем) Базы данных 255л Фомин М. М.	
15:40 - 17:15	(сем) Электротехника 619л Ибрагимов С. В.	
17:25 - 19:00		
19:10 - 20:45		

Рисунок 3

Проверка валидатором

Showing results for index.html

Checker Input

Show ☐ source ☐ outline ☐ image report

Check by No file chosen

Uploaded files with .xhtml or .xht extensions are parsed using the XML parser.

Document checking completed. No errors or warnings to show.

Used the HTML parser.

Total execution time 6 milliseconds.

Рисунок 4

Часть 3

Код программы

Файл “index.html”

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
```

```

<title>Task 3</title>
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@400;500;600;700&display=swap"
rel="stylesheet">
<link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
<table>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td class="panel">
<form action="../../task-2/index.html" method="post">
<h2 class="header">Sign Up</h2>
<input type="text" id="email" name="email" placeholder="Email"><br>
<input type="password" id="password" name="password" placeholder="Password"><br>
<input type="password" id="confirmed-password" name="confirmed-password" placeholder="Confirm
Password"><br>
<br>
<input type="submit" value="Sign Up" id="submit-button"><br>
</form>
</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</table>
</body>
</html>

```

Файл “style.css”

```

body {
    font-family: 'Montserrat', sans-serif;
    background-color: #f2f1f6;
}

input[type=text], input[type=password] {
    border-radius: 7px;
    margin: 5px;
    border: 1px solid #c5c5c7;
    font-size: 14px;
    padding: 10px;
    background-color: #fafbfb;
}

```

```

.header {
    text-align: center;
}

#submit-button {
    margin: auto;
    font-size: 17px;
    padding: 0.5em 2em;
    border: transparent;
    box-shadow: 2px 2px 4px rgba(0,0,0,0.4);
    background: dodgerblue;
    color: white;
    border-radius: 7px;
    width: 100%;
}

#submit-button:hover {
    background: rgb(2,0,36);
    background: linear-gradient(90deg, rgba(30,144,255,1) 0%, rgba(0,212,255,1) 100%);
}

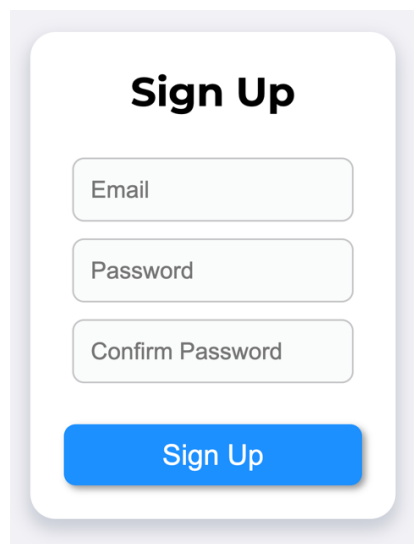
#submit-button:active {
    transform: translate(0em, 0.2em);
}

.panel {
    padding: 1px 20px 20px 20px;
    background-color: #ffffff;
    border-radius: 15px;

    -webkit-box-shadow: 0px 5px 10px 2px rgba(34, 60, 80, 0.2);
    -moz-box-shadow: 0px 5px 10px 2px rgba(34, 60, 80, 0.2);
    box-shadow: 0px 5px 10px 2px rgba(34, 60, 80, 0.2);
}

```

Работа сайта



The image shows a 'Sign Up' form. At the top, the text 'Sign Up' is displayed in a bold, black font. Below this, there are three input fields: 'Email', 'Password', and 'Confirm Password'. Each field has a light gray border and a small shadow. At the bottom of the form is a blue button with the text 'Sign Up' in white. The entire form is set against a light gray background with a subtle shadow.

Рисунок 5

Проверка валидатором

Showing results for index.html

Checker Input

Show ☐ source ☐ outline ☐ image report [Options...](#)

Check by [file upload](#) [Choose file](#) No file chosen

Uploaded files with .xhtml or .xht extensions are parsed using the XML parser.

[Check](#)

Document checking completed. No errors or warnings to show.

Used the HTML parser.

Total execution time 10 milliseconds.

Рисунок 6

Вывод

Созданы несколько страниц на языке разметки html, использована технология css, проведен анализ исходного кода валидатором.