

УДК 004.4

ПРИМЕНЕНИЕ WEB-ПРОГРАММИРОВАНИЯ ПРИ РЕШЕНИИ МАТЕМАТИЧЕСКИХ ЗАДАЧ

М.А. Гришина, студентка гр. МДИ-112, III курс

Научный руководитель: В.И. Сафонов, к.ф.-м.н., доцент Мордовский
государственный педагогический институт имени М. Е. Евсевьева
г. Саранск

В настоящее время актуальным является использование компьютерной техники в качестве технического средства подключения к локальным или глобальным сетям для получения информации, пересылки и т.д. [1]. Самым крупным объединением компьютерных сетей всего мира является Интернет. Стандарт представления веб-документов – язык разметки гипертекста HTML. Для того чтобы сделать их интерактивными, используется другой стандарт – язык веб-страниц Java Script (JS). Однако JS может быть использован не только для управления веб-страницами, но и для обучения программированию, а также в качестве средства решения ряда математических задач. В пользу этого говорят следующие доводы.

1. Язык JS является алгоритмическим. В нем используются стандартные типы данных. Синтаксис и семантика подобны используемым в таких языках программирования, как Pascal, C и др. Также он является процедурным, т.е. позволяет использовать функции, созданные самим пользователем.

2. Язык JS является широко распространенным. Для его использования не требуется закупка и установка каких-либо сред программирования. Исполняется он любым браузером (например, Internet Explorer), который имеется практически на любом современном компьютере с операционной системой Windows. Для написания кода программы достаточно любого текстового редактора, например, распространенной стандартной программы Windows – Блокнот.

3. JS позволяет использовать специальный объект – числа (Number). Можно производить операции с числами различных типов, представлять их в различных системах счисления и др.

4. JS обладает специальным объектом Math. Он предназначен для хранения некоторых математических констант и математических функций.

5. Кроме перечисленного, можно отметить принадлежность JS к Интернет-технологиям. Он может быть использован при организации дистанционного обучения.

Рассмотрим подробнее объект JS Math. Он предназначен для хранения некоторых математических констант (например, числа π) и выполнения преобразований чисел с помощью математических функций. Доступ к свойствам и методам объекта Math обеспечивается следующими выражениями: Math.свойство или Math.метод(параметры).

Таким образом, JS обладает достаточно большим набором свойств и методов, которые могут быть использованы при организации математических вычислений. Однако возникает вопрос – а не проще ли эти вычисления проводить с использованием калькуляторов, которые также могут содержать различные математические функции? Ответ на него следует из того, что JS является языком программирования, притом, алгоритмическим. Все это означает, что он позволяет организовать линейные вычисления, а также ветвления и циклы, то есть реализовать имеющийся или собственный вычислительный алгоритм. С механизмами реализации алгоритмов в JS можно ознакомиться, например, в [2], [3]. Обратимся к задачам матричной алгебры, решение которых в общем виде не под силу большинству стандартных калькуляторов.

В языке программирования матрицу можно представить в виде массива. Массив – это упорядоченный набор данных, объединенный общим именем. При обработке данных массивы могут оказать существенную помощь программисту.

Имена массива образуются также, как и имена переменных. Элементы массива могут быть любого типа. Обращение к элементам организуется с помощью их индексов, причем нумерация индексов начинается с нуля. Можно организовывать одномерные, двумерные и т.д. массивы. Рассмотрим работу с одномерными массивами. Для их создания используется следующий оператор: `ИмяМассива = new Array (количество элементов)`.

Пример 1. Организовать одномерный массив из трех произвольных чисел и вывести на экран второй элемент.

```
<SCRIPT>
mas=new Array(3)
mas[1]=5
mas[2]=-2
mas[3]=0
alert (mas[2])
</SCRIPT>
```

Отметим, что JS позволяет не указывать количество элементов создаваемого массива: в этом случае создается пустой массив. Элементы добавляются к массиву обычным присвоением.

Одно из важных отличий массива от хранения значений в обычных переменных заключается в упрощении обработки при помощи индексов. В этом случае возможно простая обработка практически любого количества данным, причем, размер программы будет при этом небольшим.

Пример 2. Организовать ввод и вывод одномерного массива, элементы которого – последовательные натуральные числа.

```
<SCRIPT>
var n=4
m=new Array(n)
for (i=1; i<=n; i=i+1)
{ m[i]=i }
```

```
s=""  
for (i=1; i<=n; i=i+1)  
{s=s+m[i]+' '  
alert(s)  
</SCRIPT>
```

Рассмотренные в примерах массивы являются одномерными. Однако элементы массива могут содержать данные любых типов, в том числе, и массивы. Таким образом, если в качестве элементов одномерного массива взять массивы, то получится двумерный массив.

Обращение к элементам такого массива происходит следующим образом: ИмяМассива[1_индекс][2_индекс]. Аналогично можно продолжить увеличение размерности массива. Покажем, как осуществляется ввод и вывод элементов двумерного массива.

Пример 3. Организовать ввод и вывод двумерного массива.

```
<SCRIPT>  
//Ввод двумерного массива X  
x=new Array(3)  
x[0]=new Array (1,2,3)  
x[1]=new Array (4,5,6)  
x[2]=new Array (7,8,9)  
//Вывод двумерного массива X на экран  
s=""  
for (i=0;i<=2;i=i+1)  
{ for (j=0;j<=2;j=j+1)  
{ s=s+x[i][j]+' '  
s=s+'\n'}  
alert(s)  
</SCRIPT>
```

Теперь рассмотрим использование массивов JS для решения задач матричной алгебры.

Пример 4. Найти сумму элементов двумерного массива.

```
<SCRIPT>  
summa=0  
for (i=0;i<=2;i=i+1)  
{  
for (j=0;j<=2;j=j+1)  
{ summa=summa+x[i][j]}  
}  
alert('summa='+summa)  
</SCRIPT>
```

Пример 5. Умножить элементы двумерного массива на некоторое число.

```
<SCRIPT>  
h=2
```

```
for (i=0;i<=2;i=i+1)
{
for (j=0;j<=2;j=j+1)
{x[i][j]=h*x[i][j]}
}
</SCRIPT>
```

Пример 6. Найти сумму элементов двумерных матриц: $A_{n \times n}$ и $B_{n \times n}$.

```
<SCRIPT>
for (i=0;i<=n;i=i+1)
{
for (j=0;j<=n;j=j+1)
{c[i][j]=a[i][j]+b[i][j]}
}
alert('сумма матриц =')
s=""
for (i=0;i<=2;i=i+1)
{for (j=0;j<=2;j=j+1)
{s=s+c[i][j]+' '}}
s=s+'\n'
alert(s)
</SCRIPT>
```

Таким образом, язык сценариев может оказать помощь при решении ряда математических задач, в частности, задач матричной алгебры. Он распространен, достаточно прост в освоении, для его использования не требуется установка специализированного дорогостоящего программного обеспечения. Он может быть использован в школьном курсе информатики при обучении программированию.

Список литературы:

1. Сафонов В.И. Организация информационного взаимодействия в информационно-образовательном пространстве педагогического вуза // Педагогическое образование в России. – 2013. – № 1. – С. 48-52.
2. Дунаев В.Н. Самоучитель Java Script // СПб.: Питер, 2006. – 395 с.
3. Климов А.М. Java Script на примерах // СПб.: ВHV-СПб, 2006. – 256 с.