

Работа №1 «Основные алгоритмические конструкции и структуры данных типа массив»

Цель работы: освоить базовые понятия программирования.

Задачи. В процессе занятий, запланированных на эту работу необходимо выполнить три задания.

Время выполнения: 4 часа.

В первом задании генерируется одномерный массив случайных чисел с заданными характеристиками и вычисляется среднее арифметическое значение (или минимальное, или максимальное значение – в зависимости от варианта) среди значений массива из заданного интервала.

Во втором задании осуществляется вывод на экран значений массива и среднего арифметического или минимального (максимального) значений массива, получаемых по первому заданию.

В третьем задании нужно ответить на вопросы для самоконтроля.

Все вычисления в первом и втором заданиях оформляются в виде функций, запускаемых с помощью события Click. Таким образом, реализуется простой вариант моделирования, когда можно многократно вначале имитировать ряд псевдослучайных чисел и выполнять необходимые вычисления с ними, а затем осуществлять вывод моделируемых значений и результатов расчётов в удобном для восприятия виде.

Краткие теоретические сведения

Исчерпывающий материал по теме работы находится на сайте: pvn.org.ru в лекции №3. Переход к лекции №3 – по ссылке меню [Программа обучения – Основы программирования].

Особое внимание перед выполнением настоящей работы следует обратить на понятие “число” в языке Javascript. Кроме материала в лекции №3 о правилах работы с числами (типы числовых переменных, накопление ошибок, округление и другое) рекомендуется прочитать и твёрдо усвоить материал, размещённый здесь: <http://learn.javascript.ru/number>.

Задания

Задание №1: Рассчитать среднее арифметическое, или минимальное, или максимальное значение по значениям одномерного массива $x_i, i=1,2,\dots, n$, при соблюдении следующих условий:

массив необходимо предварительно создать, заполнив случайными числами x , попадающими в интервал от $-z$ до z , используя метод *Math.random()*;

для расчёта среднего (минимума или максимума) из имеющегося массива случайных чисел необходимо отобрать значения, попадающие в интервал, задаваемый в различных вариантах числами: a и b ; либо a ; либо b (соблюдая условие: $a < b$);

исходные данные (n, z, a, b), соответствующие вашему варианту, нужно ввести с использованием метода *prompt()* или **формы**;

создание массива, заполнение случайными числами, отбор значений и расчёт среднего арифметического (минимума или максимума) нужно оформить в виде функции;

вызов функции нужно оформить посредством обработчика события *onClick* для кнопки, создаваемой с помощью тэга `<button>...</button>`;

вариант выбирается из таблицы 3.5.1 и должен соответствовать двум последним цифрам номера вашей зачётной книжки.

Внимание! В этом задании выполняются расчёты без вывода результатов на экран. Вывод результатов осуществляется с помощью функций, создаваемых в следующем задании.

Таблица 3.5.1 Варианты к заданию №1.

Вариант	Что рассчитать	Что вводить	Условие для отбора значений массива x_i
00	среднее	n, z, b	$x_i < b$
01	минимум	n, z, b	$x_i < b$
02	максимум	n, z, b	$x_i < b$
03	среднее	n, z, a, b	$a \leq x_i \leq b$
04	минимум	n, z, a, b	$a \leq x_i \leq b$
05	максимум	n, z, a, b	$a \leq x_i \leq b$
06	среднее	n, z, a	$x_i \geq a $

07	минимум	n, z, a	$x_i \geq a $
08	максимум	n, z, a	$x_i \geq a $
09	среднее	n, z, b	$ x_i \leq b$
10	минимум	n, z, b	$ x_i \leq b$
11	максимум	n, z, b	$ x_i \leq b$
12	среднее	n, z, a	$ x_i \geq a $
13	минимум	n, z, a	$ x_i \geq a $
14	максимум	n, z, a	$ x_i \geq a $
15	среднее	n, z, a, b	$ a \leq x_i \leq b $
16	минимум	n, z, a, b	$ a \leq x_i \leq b $
17	максимум	n, z, a, b	$ a \leq x_i \leq b $
18	среднее	n, z, a, b	$a \leq x_i \leq b$
19	минимум	n, z, a, b	$a \leq x_i \leq b$
20	максимум	n, z, a, b	$a \leq x_i \leq b$

Задание №2. В этом задании необходимо вывести данные, полученные в первом задании:

исходные данные задания №1

значения массива $x_j, i=1,2,\dots, n$;

среднее арифметическое, минимальное или максимальное значение (одно какое-то, в соответствии с вариантом), и количество значений, отобранных для расчёта среднего арифметического.

При этом должны быть соблюдены следующие условия:

- √ значения массива и среднее арифметическое необходимо округлить до l знаков после десятичной точки с помощью метода *toFixed()*, предусмотренного в языке *Javascript* для округления вещественного числа;
- √ вывод значений массива оформить в виде функции, вызываемой с помощью обработчика события *onClick*, "прикреплённого" к кнопке, созданной с помощью тэга `<button ...> ... </button>`.

В качестве входных (фактических) параметров этой функции должны быть: массив $x_j, i=1,2,\dots, n$ и переменная l (число знаков для округления значений массива);

- √ вывод среднего арифметического (или минимального, максимального значения) и

количества значений, отобранных для расчёта среднего арифметического (или минимального, максимального значения), также оформить в виде функции, вызываемой с помощью обработчика события onClick, "прикреплённого" к ещё одной кнопке, созданной с помощью тэга `<button ...> ... </button>` ;

В качестве входных (фактических) параметров этой функции должны быть: среднее арифметическое, полученное в предыдущем задании, и переменная *l* (число знаков для округления значений массива);

- √ выводиться данные должны в специально созданные для этого блоки с помощью тэгов `<div ...> ... </div>` , именованные с помощью идентификатора *id*;
- √ вывод значений массива необходимо оформить в виде таблицы с двумя столбцами: первый столбец - номер по порядку элемента массива ; второй столбец — значения элементов массива; первая строка таблицы («шапка» таблицы) должна содержать наименования столбцов.

Методика выполнения

Методика выполнения подробно описана на сайте: rvn.ho.ua в лекции №3. Переход к лекции №3 – по ссылке меню [Программа обучения – Основы программирования]. Настоящая лабораторная работа выполняется после прохождения всех подразделов лекции №3, предусмотренных лекционными занятиями и заданиями для самостоятельной работы.

В целях усвоения материала лекции №3 для самостоятельной работы студентов предусмотрено выполнение упражнений, в которых имеются фрагменты, полностью охватывающие задания настоящей лабораторной работы.

В процессе лекционной работы и выполнения упражнений студенты уже должны были создать присоединяемые js-файлы (другими словами – библиотеки) с функциями, используемыми в этой работе. Разрешается использовать эти, ранее созданные библиотеки, в настоящей работе.

Следует обратить особое внимание на понятия “локальности” и “глобальности” видимости переменных в связи с понятием “функция” при передаче входных параметров в функцию и при возвращении результатов из функции.

3.5.4 Рекомендации по обработке и оформлению полученных результатов

Работа оформляется в виде веб-документа и сохраняется в файле Petrov.html с использованием ранее созданного шаблона.

Все функции должны быть помещены в файл с расширением js, присоединяемый к

файлу Petrov.html.

Все файлы работы (Petrov.html, css-файл и js-файл) должны быть помещены в папку Petrov.

В конце файла Petrov.html должны быть помещены ответы на контрольные вопросы, помещённые ниже.

3.5.5 Вопросы для самоконтроля

1. Для чего нужны тэги `<script></script>` ?
2. Является ли код языка Javascript чувствительным к регистру букв ?
3. Способы создания переменной?
4. Может ли имя переменной начинаться с цифры?
5. Можно ли использовать символ пробела при образовании имён переменных ?
6. Типы переменных?
7. Операция присваивания ?
8. Арифметические операции ?
9. Операции сравнения ?
10. Данные логического типа ?
11. Логические операции ?
12. Операторные скобки ?
13. Ключевое (служебное или зарезервированное) слово, предназначенное для образования переменных ?
14. Назначение функции `Number()` ?
15. Назначение функции `String()` ?
16. Метод переменной для округления числа до требуемого количества знаков после десятичной точки ?
17. Метод `window.alert()` ?
18. Метод `window.confirm()`?
19. Метод `window.prompt()`?
20. Присвоение значения переменной с использованием метода `prompt()` ?
21. Метод `document.write()` ?
22. Однострочный комментарий ?
23. Многострочный комментарий ?
24. Создание одномерного массива?
25. Присвоение значений элементам одномерного массива?
26. Свойство, в котором хранится размер одномерного массива ?
27. Индекс последнего по счёту элемента одномерного массива ?
28. Синтаксис полного оператора `if` ?
29. Синтаксис сокращённого оператора `if` ?
30. В каком случае можно не использовать операторные скобки в операторе `if`?
31. Синтаксис оператора цикла `for` ?
32. Будет ли ошибкой, если в теле оператора цикла `for` не будет ни одного оператора?

33. Чему равна переменная-параметр в цикле `for` после завершения оператора цикла?
34. Обработчик события?
35. Чувствительны ли имена функций к регистру букв ?
36. Пример создания функции?
37. Могут ли зарезервированные (служебные, ключевые) слова Javascript использоваться в качестве имени функции ?
38. Может ли имя функции начинаться с цифры ?
39. Внутри каких скобок указываются параметры функции: фигурных, круглых, квадратных или вообще не нужно скобок вокруг параметров ?
40. Каким символом разделяются параметры функции ?
41. Ключевое (служебное или зарезервированное) слово, предназначенное для образования функции ?
42. Формальные параметры функции?
43. Локальная область видимости переменных ?
44. Возвращение результата из функции ?
45. Как передать из функции несколько значений переменных?
46. Как передать из функции несколько одномерных массивов?
47. Создание двумерного массива ?
48. Вычисление суммы по данным одномерного массива ?
49. Вычисление суммы по данным двумерного массива ?
50. Функция (метод) для вычисления случайного числа ?
51. Метод объекта `Math` для округления вещественных чисел ?
52. Свойство объекта `Math`, содержащее число "пи" ?
53. Метод `getElementById()` ?
54. В чём различия вывода данных методом `document.write()` и с помощью свойства `innerHTML`?
55. Пример присвоения переменной значения, вводимого в текстовое поле формы, если форма имеет имя "forma", а имя текстового поля - "input" ?
56. Какова область видимости переменной, создаваемой в функции, если при её создании используется зарезервированное слово `var` ?
57. Не менее трёх событий, связанных с манипулятором мышь?
58. В каком случае можно не использовать операторные скобки в операторе `if`?