

МЕТОДИЧЕСКИЕ ОСОБЕННОСТИ ПРИМЕНЕНИЯ ПРОГРАММНОЙ СРЕДЫ MATHCAD В УСЛОВИЯХ ОГРАНИЧЕННОГО УЧЕБНОГО ВРЕМЕНИ

Гниломедов П.И.

ФГБОУ ВО «Уральский государственный университет путей сообщения (УрГУПС)», Екатеринбург, e-mail: gpivan@mail.ru

В статье рассмотрены методические аспекты организации обучения по использованию прикладного программного пакета MathCAD в условиях ограниченности учебного времени. Практика показывает, что простое извлечение итогового значения того или иного статистического параметра, без понимания механизма его нахождения, ведет к формализму освоения знания и быстрому забыванию учебного материала. Одной из причин возникновения такой учебной ситуации является ограниченность времени аудиторных занятий, на которых обучающиеся имеют возможность ознакомиться с лицензионными программными пакетами. Автором обоснованы основные методические направления подбора учебных заданий, которые способствуют уменьшению негативных факторов, сопровождающих процесс практического освоения MathCAD. Наиболее значимыми направлениями являются тщательное дозирование и подбор учебного материала по объему и содержанию на основе субъективного опыта обучающихся. В качестве примера практической реализации указанных методических направлений выделен раздел математической статистики, поскольку именно здесь проявляется значительная востребованность в численных и программных методах обработки данных. В рамках решения локальной задачи по составлению вариационного ряда имеющихся данных установлено, что для обучающихся наиболее удобной формой представления упорядоченного множества данных является двумерная матрица. В работе предложен алгоритм сквозного упорядочения элементов двумерного массива, не предусмотренный встроенными функциями сортировки данных. Отмечено, что данный алгоритм позволяет дифференцированно подходить к управлению учебной работой в ходе решения рассматриваемой задачи в зависимости от уровня математической подготовленности и практических умений обучающихся.

Ключевые слова: обучение математике в вузе, аудиторная работа, статистическая обработка, программа MathCAD.

METHODOLOGICAL FEATURES OF APPLICATION OF THE MATHCAD PROGRAM ENVIRONMENT IN CONDITIONS OF LIMITED LEARNING TIME

Gnilomedov P.I.

FGBOU VO «Ural State University of Railway Transport USURT», Ekaterinburg, e-mail: gpivan@mail.ru

The article considers the methodological aspects of organizing training on the use of the applied software package MathCAD in conditions of limited study time. Practice shows that simply extracting the final value of one or another statistical parameter, without understanding the mechanism of its finding, leads to the formalism of mastering knowledge and quickly forgetting the educational material. One of the reasons for the emergence of such a learning situation is the limited time of classroom studies, where students have the opportunity to get acquainted with licensed software packages. The author substantiates the main methodological directions for the selection of training tasks that help reduce the negative factors that accompany the process of practical development of MathCAD. The most significant areas are defined: careful dosing and selection of educational material in terms of volume and content based on the subjective experience of students. As an example of the practical implementation of these methodological areas, the section of mathematical statistics is highlighted, since it is here that there is a significant demand for numerical and software methods of data processing. As part of solving a local problem of compiling a variational series of available data, it was found that for students the most convenient form of representing an ordered set of data is a two-dimensional matrix. The paper proposes an algorithm for end-to-end ordering of elements of a two-dimensional array, which is not provided by the built-in data sorting functions. It is noted that this algorithm allows a differentiated approach to the management of educational work in the course of solving the problem under consideration, depending on the level of mathematical preparedness and practical skills of students.

Keywords: teaching mathematics at a university, classroom work, statistical processing, MathCAD program.

В настоящее время объективной особенностью учебного процесса в техническом вузе стало значительное уменьшение времени аудиторной учебной работы студентов. Наряду с этим специалистами отмечаются слабая математическая подготовка и несформированность умений учебной деятельности у многих студентов младших курсов, поступивших в вуз [1, 2]. Эти факторы негативным образом сказываются на качестве освоения обучающимися учебного материала. Вместе с тем, по мере изучения теоретического и практического содержания основных разделов курса математики студенты должны получить навыки использования прикладных компьютерных программ, поскольку это является важной основой профессиональных компетенций будущих инженеров.

Очевидно, что оптимальной можно считать такую учебную ситуацию, когда функциональные возможности программного обеспечения могут быть реализованы в ходе освоения теоретического материала и практических методов решения задач. Объективно потребности в программном сопровождении решения задач увеличиваются по мере продолжения обучения и усложнения содержания осваиваемых дисциплин. Важным является то, что многие программные математические средства, которые необходимы в будущей профессиональной работе обучающихся, например MathCAD, MATLAB и другие, обычно изучаются непосредственно на практических занятиях в учебном заведении. И именно в этом сегменте учебного процесса проявляется проблема, обусловленная недостатком времени аудиторной работы, которая ведет к формализму умений использования прикладных программ. Суть проблемы заключается в том, что обучающиеся осваивают теоретическую основу какого-либо раздела, приемы решения соответствующих задач, но при этом у них отсутствует практика осознанного применения основных функций прикладных программных пакетов. Либо, наоборот, в процессе использования программных математических пакетов обучающиеся получают возможность находить ответ в поставленной задаче, но при этом не понимают теоретических основ алгоритмов решений и выполняемых действий. Очевидно, что получение итогового значения того или иного искомого параметра с использованием встроенных программ, без представления о математической сущности теоретического и практического содержания, ведет к формальному освоению учебного материала и его быстрому забыванию.

Цель исследования – обосновать основные направления методической работы преподавателя по освоению обучающимися встроенных программных пакетов математической среды MathCAD на занятиях в условиях значительного ограничения учебного времени.

Методы исследования: анализ условий учебной деятельности студентов, экспертная оценка, выполнение обучающимися экспресс-заданий на практических занятиях.

Работа проводилась во время практических занятий со студентами Уральского государственного университета путей сообщения, обучающихся по направлению подготовки «Подвижной состав железных дорог», в пределах нормативного времени, предусмотренного учебной программой: одно двухчасовое занятие аудиторной работы в две недели. Таким образом, обучающиеся имели доступ к лицензионному прикладному программному пакету MathCAD 1 раз в 2 недели. В ходе исследования наблюдалась группа студентов 2-го курса в количестве 70 человек.

Анализ выделенных выше факторов, сопровождающих учебный процесс, позволил сделать вывод о необходимости выбора небольшого по объему сегмента в изучаемом материале, понятного всем обучающимся в практическом исполнении, применительно к которому можно использовать функциональный аппарат программного пакета MathCAD.

Практика показывает, что значительный рост востребованности численных и программных методов проявляется при изучении учебного материала из раздела математической статистики. На этапе освоения способов обработки данных обучающиеся знакомятся с понятиями математических параметров и алгоритмами их нахождения. В этот период приходится выполнять большой объем «ручной», часто рутинной, работы с массивами данных [3]. Заметим, что многие программные продукты позволяют сразу извлечь необходимые математические параметры, которые описывают основные свойства выборки. Прежде всего, это статистические параметры: выборочная средняя, дисперсия, среднеквадратическое отклонение и др. [4, с. 283–285]. Безусловно, такие программы являются удобным инструментом, значительно ускоряющим математическое описание изучаемых объектов. Однако, как показывает опыт, использование готовых программных функций без предварительного изучения теоретических основ процесса обработки данных нецелесообразно, так как ведет к формализму практического применения математических пакетов.

В ходе анализа учебного материала была выделена не сложная, но довольно трудоемкая промежуточная локальная задача, которую приходится решать практически всем обучающимся: составление вариационного ряда – упорядоченного множества исходных данных. Отметим важные методические особенности этой промежуточной задачи. Во-первых, она функционально понятна всем обучающимся, поскольку не выходит за границы их субъективного опыта. Во-вторых, в ходе «ручной» работы нетрудно убедиться в том, что потребность машинного способа упорядочения элементов возрастает с увеличением числа данных. В-третьих, реализация функциональных возможностей математического пакета обладает значительным обучающим потенциалом, дифференцируемо реализуемым, в

зависимости от уровня подготовленности обучающихся и имеющегося на его освоение времени.

Составление вариационного ряда – это первый этап обработки экспериментальных данных, как правило, представленных в некоторой таблице. Упорядоченное расположение значений измеряемой величины – экспериментальных данных – значительно облегчает первоначальную оценку содержания выборки [5]. Практика показывает, что упорядочение данных обучающиеся обычно проводят вручную. Для выявления условий, способствующих более эффективному восприятию учебной задачи и результатов использования программных функций, им были предложены экспресс-задания, в которых было необходимо упорядочить по величине множество данных, представленное в произвольном виде. Результаты выполнения заданий показали, что подавляющее большинство обучающихся в случае наличия большого числа данных (10 и более) итоговый результат представляют в виде таблицы с несколькими строками и столбцами, в которой все элементы расположены в порядке сплошного возрастания их значений от столбца к столбцу или от строки к строке. Наименьшее значение измеряемой величины находится в левом верхнем углу таблицы, а наибольшее – в правом нижнем углу. Кроме того, если предложенный для сортировки исходный массив данных имел значительный объем и был представлен в виде матричной таблицы с определенным числом строк и столбцов, то практически все обучающиеся итоговый результат также представляли в аналогичной матричной форме такой же размерности. Очевидно, что такой вид привычен для обучающихся. Это позволяет сделать вывод о том, что наиболее удобной для дальнейшей работы формой упорядоченного массива данных будет его наглядное представление в виде двумерной матрицы со сквозным упорядочением всех элементов по строкам или столбцам, поочередно от ряда к ряду.

Именно здесь выявляется важная в рамках данной учебной задачи проблема. Дело в том, что в программном пакете MathCAD упорядочение элементов матрицы предполагает только сортировку столбцов по элементам выбранной строки с помощью встроенной функции $rsort(A, i)$ или аналогичную сортировку строк по элементам выбранного столбца с помощью функции $csort(A, i)$ [6]. Сквозного упорядочения сразу всех элементов двумерной матрицы массива данных, в привычном понимании, начиная с элемента, находящегося в верхнем левом углу и далее по чередующимся столбцам или строкам, в программной базе функций нет, и получить сразу сквозное упорядочение элементов во всей матрице не представляется возможным. Обозначенная проблема влечет за собой методическую задачу построения алгоритма упорядочения элементов посредством функциональной базы MathCAD, который является удобной и понятной альтернативой их ручной сортировке.

Анализ содержания встроенных матричных функций позволил определить порядок действий по сплошному упорядочению всех элементов двумерного массива, чтобы конечный результат был представлен в исходной форме – в виде двумерной матрицы (рис. 1).



Рис. 1. Алгоритм использования встроенных функций MathCAD

Рассмотрим ход реализации предложенного алгоритма. Пусть таблица исходных данных представлена в виде прямоугольной матрицы размерности 10×10 (рис. 2).

$$A := \begin{pmatrix} 1.60 & 1.59 & 0.92 & 2.19 & 3.09 & 1.14 & 1.39 & 1.34 & 2.07 & 2.27 \\ 1.02 & 1.56 & 1.28 & 1.97 & 2.64 & 1.37 & 2.77 & 0.25 & 1.82 & 2.04 \\ 1.92 & 2.58 & 0.88 & 1.22 & 2.55 & 0.78 & 0.82 & 2.64 & 2.27 & 2.15 \\ 2.54 & 1.73 & 1.56 & 1.64 & 1.85 & 0.68 & 2.10 & 1.02 & 2.29 & 1.91 \\ 2.49 & 1.67 & 1.76 & 1.86 & 1.78 & 1.20 & 2.31 & 1.39 & 1.40 & 1.17 \\ 2.81 & 1.48 & 1.80 & 2.11 & 2.05 & 1.32 & 2.92 & 2.23 & 1.23 & 2.51 \\ 0.49 & 2.95 & 1.59 & 1.86 & 1.76 & 0.52 & 1.73 & 2.06 & 2.44 & 1.60 \\ 1.64 & 2.29 & 3.08 & 1.24 & 1.15 & 1.44 & 1.47 & 2.30 & 1.07 & 1.28 \\ 2.43 & 3.19 & 0.75 & 2.90 & 0.73 & 1.54 & 2.18 & 2.13 & 0.86 & 1.29 \\ 1.56 & 2.23 & 1.12 & 2.16 & 1.51 & 1.86 & 1.55 & 0.97 & 2.20 & 1.53 \end{pmatrix}$$

Рис. 2. Листинг с исходной матрицей данных

Поскольку операция сортировки элементов может быть применена только к одномерному массиву, то на первом этапе нужно провести ряд действий, которые двумерную матрицу преобразуют в одномерную матрицу-строку, содержащую все исходные данные. Для этого, выделив каждый столбец в виде отдельной матрицы с помощью операции $A^{(i)}$, транспонируем получившиеся одномерные матрицы-столбцы операцией $A^{(i)T}$ и посредством встроенной функции слияния матриц `augment (A, B, C, ...)` составим единую матрицу-строку. Затем, используя встроенную функцию сортировки столбцов `rsort (A, i)` (здесь $i = 1$ для единственной первой строки), получаем упорядоченный вариационный ряд в виде одномерной матрицы-строки (рис. 3).

$$A1 := \text{augment} \left(A^{(1)T}, A^{(2)T}, A^{(3)T}, A^{(4)T}, A^{(5)T}, A^{(6)T}, A^{(7)T}, A^{(8)T}, A^{(9)T}, A^{(10)T} \right)$$

$$A0 := \text{rsort} (A1, 1)$$

Вариационный ряд – упорядоченный ряд элементов матрицы-строки A1

$$A_0 =$$

	1	2	3	4	5	6	7	8	9	10
1	0.25	0.49	0.52	0.68	0.73	0.75	0.78	0.82	0.86	...

Рис. 3. Формирование вариационного ряда в виде матрицы-строки

Важно, что это уже приемлемый предварительный результат, который обучающиеся в зависимости от уровня своей подготовки могут использовать в ручном режиме для дальнейшей работы. В то же время представление упорядоченного множества элементов в таком виде имеет существенный недостаток – использование большого по объему массива данных в виде одной строки становится неудобным при практическом применении. На следующем этапе нужно привести одномерный массив к исходному прямоугольному виду, в котором все элементы имеют сквозную упорядоченность по строкам и столбцам. Для этого выделим из получившейся одномерной матрицы подматрицы-строки $B_1, B_2, B_3, \dots, B_{10}$, состоящие из десяти элементов, с помощью встроенной функции `submatrix (A0, ir, jr, ic, jc)`. Здесь для одной и единственной строки $ir = jr = 1$, а номера столбцов соответствуют начальному и последнему порядковому номеру каждого последующего десятка элементов матрицы, начиная с первого (рис. 4).

$B_1 = \text{submatrix} (A_0, 1, 1, 1, 10)$ $B_6 = \text{submatrix} (A_0, 1, 1, 51, 60)$
 $B_2 = \text{submatrix} (A_0, 1, 1, 11, 20)$ $B_7 = \text{submatrix} (A_0, 1, 1, 61, 70)$
 $B_3 = \text{submatrix} (A_0, 1, 1, 21, 30)$ $B_8 = \text{submatrix} (A_0, 1, 1, 71, 80)$
 $B_4 = \text{submatrix} (A_0, 1, 1, 31, 40)$ $B_9 = \text{submatrix} (A_0, 1, 1, 81, 90)$
 $B_5 = \text{submatrix} (A_0, 1, 1, 41, 50)$ $B_{10} = \text{submatrix} (A_0, 1, 1, 91, 100)$

Рис. 4. Разбивка вариационного ряда на подматрицы из десяти элементов

Далее, после транспонирования этих матриц в матрицы-столбцы с помощью операций (B_i^T) , используя повторно функцию слияния матриц `augment (A, B, C, ...)`, формируем объединенную двумерную матрицу B исходной размерности 10×10 (рис. 5).

$$B := \text{augment} \left((B1^T)^{(1)}, (B2^T)^{(1)}, \dots, (B10^T)^{(1)} \right)$$

Объединенная матрица B0 размерностью 10×10

$$B = \begin{pmatrix} 0.25 & 0.92 & 1.22 & 1.39 & 1.56 & 1.73 & 1.91 & 2.15 & 2.29 & 2.64 \\ 0.49 & 0.97 & 1.23 & 1.40 & 1.56 & 1.76 & 1.92 & 2.16 & 2.30 & 2.64 \\ 0.52 & 1.02 & 1.24 & 1.44 & 1.59 & 1.76 & 1.97 & 2.18 & 2.31 & 2.77 \\ 0.68 & 1.02 & 1.28 & 1.47 & 1.59 & 1.78 & 2.04 & 2.19 & 2.43 & 2.81 \\ 0.73 & 1.07 & 1.28 & 1.48 & 1.60 & 1.80 & 2.05 & 2.20 & 2.44 & 2.90 \\ 0.75 & 1.12 & 1.29 & 1.51 & 1.60 & 1.82 & 2.06 & 2.23 & 2.49 & 2.92 \\ 0.78 & 1.14 & 1.32 & 1.53 & 1.64 & 1.85 & 2.07 & 2.23 & 2.51 & 2.95 \\ 0.82 & 1.15 & 1.34 & 1.54 & 1.64 & 1.86 & 2.10 & 2.27 & 2.54 & 3.08 \\ 0.86 & 1.17 & 1.37 & 1.55 & 1.67 & 1.86 & 2.11 & 2.27 & 2.55 & 3.09 \\ 0.88 & 1.20 & 1.39 & 1.56 & 1.73 & 1.86 & 2.13 & 2.29 & 2.58 & 3.19 \end{pmatrix}$$

*Рис. 5. Итоговый вид вариационного ряда в виде матрицы размерности 10×10 (в функции *augment* часть упорядоченных элементов от $(B3^T)^{(1)}$ до $(B9^T)^{(1)}$ не показана)*

Видно, что полученный вариационный ряд представлен в удобной для восприятия форме – матрицы с исходной размерностью 10×10 – и имеет сквозное упорядочение элементов по возрастанию их значений сверху вниз, от столбца к столбцу, начиная с наименьшего верхнего левого элемента. В заключение отметим, что вариационный ряд, в котором упорядочение элементов происходит по строкам, также является привычным и удобным для восприятия и практического использования. Такие вариационные ряды могут быть составлены с использованием аналогично построенного алгоритма и тех же встроенных функций. Такую задачу и ее решение при необходимости обучающиеся могут рассмотреть самостоятельно.

Выводы

В процессе освоения учебного материала, который предполагает использование программных математических продуктов в условиях серьезного ограничения учебного времени, с высокой долей вероятности может возникнуть ситуация непонимания и формального использования обучающимися функциональных возможностей прикладных математических пакетов. В ходе работы было установлено, что оптимальным направлением методической работы является тщательный подбор учебного материала, для сопровождения которого планируется использование программных продуктов. Объем и содержание материала должны определяться в зависимости от степени освоения обучающимися теоретического аппарата и понимания ими основ практических действий.

Подбор сложных задач, с точки зрения уровня реализации программных продуктов, когда использование имеющейся операционной базы не дает возможности напрямую получить требуемый результат, должен предполагать функционально понятную всем

обучающимся схему «ручных» действий в рамках их субъективного опыта. Алгоритм использования встроенных математических функций, построенный на основе такой схемы, позволяет дифференцированно подходить к управлению учебной работой в зависимости от уровня математической подготовленности и практических умений обучающихся.

Список литературы

1. Токтарова В.И., Федорова С.Н. Математическая подготовка студентов: причины негативных тенденций // Высшее образование в России. 2017. № 208. С. 85-92.
2. Мустафина Д.А., Ребро И.В., Рахманкулова Г.А. Особенности математической подготовки студентов технических вузов в условиях дефицита времени // Научное обозрение. Педагогические науки. 2019. № 3-1. С. 26-30.
3. Ие О.Н. Использование среды mathcad при обучении студентов технических специальностей теории вероятностей // Современные тенденции развития методики обучения математике в высшей школе. 2017. № 45. С. 44-49.
4. Кричевец А.Н., Шикин Е.В., Дьячков А.Г. Математика для психологов: учебник. М.: Флинта: Московский психологический социальный институт, 2003. 376 с.
5. Гниломедов П.И. Особенности использования MathCAD в процессе обучения математическому и имитационному моделированию студентов транспортного вуза: сб. науч. тр. / под науч. ред. Г.А. Тимофеевой, д-ра физ.-мат. наук и О.В. Куликовой, канд. пед. наук. Екатеринбург: УрГУПС, 2022. Вып. 4. С. 55-62.
6. Exponenta.ru. [Электронный ресурс]. URL: http://old.exponenta.ru/SOFT/MATHCAD/UsersGuide/chapter12/12_5.asp (дата обращения: 02.07.2023).