

УДК 51-74
ВАК 05.13.00
РИНЦ 20.01.00

Скаляр и вектор в компьютерных вычислениях

Очков В.Ф., Архипцев А.В.

Национальный исследовательский университет «Московский энергетический институт»

Подробная контактная информация об авторах:

Очков Валерий Федорович, д.т.н., профессор

Национальный исследовательский университет «Московский энергетический институт»

Адрес: 111250, Россия, Москва, Красноказарменная ул., д. 14, раб. тел. +7-495-362-71-71,

моб. тел. +7-916-659-26-38, ochkov@tw.t.mpei.ac.ru, skype OchkovVF

Архипцев Александр Вячеславович, студент

Национальный исследовательский университет «Московский энергетический институт»

Аннотация

В статье рассмотрены два фундаментальных типа данных – скаляр и вектор (массив), без умения работать с которыми невозможно решать на компьютере школьные и вузовские задачи по математике, физике, химии и другим техническим учебным дисциплинам.

Затронуты некоторые фундаментальные основы преподавания информатики в школе и университете.

Ключевые слова: Скаляр, вектор, массив данных, Mathcad, электронная таблица, компьютерные вычисления в школе и в вузе

SCALAR AND VECTOR IN COMPUTATION

Ochkov V.F., Arkhipsev F.V.

National research university «MPEI»

Abstract

The article deals with two fundamental data types - scalar and vector (array), without the ability to work with that cannot be solved on a computer school and university problems in mathematics, physics, chemistry and other technical training courses. Cover some fundamentals of teaching computer science at school and university.

Key words: Scalar, vector, array data, Mathcad, spreadsheet, computer calculations at school and in college

На моих лекций по информатике в Московском энергетическом институте я иногда делаю опросы для того, чтобы уточнить, что мои студенты изучали раньше в школе и что они изучают сейчас на параллельных курсах по математике физике, химии и другим институтским учебным дисциплинам. На вопрос о том, знают ли они, что такое *скалярная* физическая величина и *векторная* физическая величина, все студенты хором отвечают: «Знаем! Нам все уши в школе на уроках физики прожужжали, объясняя это!». «Знаем» они выкрикивают вслух, а про «уши», конечно, умалчивают. Но в студенческой реакции на вопрос можно, конечно, услышать данное продолжение ответа... Все школьники и студенты безошибочно знают, что масса – это скалярная величина, а вес (сила) – векторная и что задачи по физике нужно решать с учетом этих фундаментальных понятий.

Скаляр и вектор присутствует и в языках программирования, электронных таблицах и математических пакетах, которые в настоящее время широко используются

при преподавании информатики и других дисциплин (математика, физика, химия) в школе и вузе.

В языках программирования есть понятие *массива* (array) – одномерного, двумерного и более двухмерного. Электронные таблицы (SuperCalc, VisiCalc, Excel и др.) создавались именно для того, чтобы хранить и обрабатывать эти самые таблицы – одну из разновидностей массивов данных. В математических пакетах, в частности, в пакете Mathcad [1] понятие массива прописано в явном виде. Пользователь может ввести в расчет *массив* с одним столбцом, который в среде Mathcad обозначен как *вектор*. Но при этом нужно четко понимать, что маткадовый вектор несет в себе как минимум две смысловые нагрузки, две функции, которые мы попытаемся уяснить на примере решения двух несложных задач.

На рис. 1 показан расчет ускорения тела с заданной массой, если известны силы, приложенные к телу в трех направлениях – по оси X, Y и Z (наша задача решается в трехмерном пространстве). Но мы можем перевести ее и на плоскость (двухмерное пространство – вектор с двумя компонентами) и на линию (одномерный вектор с одним компонентом, который, тем не менее, остается вектором, а не скаляром). А можно перейти и в четырехмерное и «болемерное» пространство, которое часто описывается в научной фантастике.

Сила (N)	Масса (kg)
1000	20
2000	
-3000	

$\|Сила\| = 3741.7 \text{ N}$
 $Ускорение := \frac{Сила}{Масса}$
 $Ускорение = \begin{bmatrix} 50 \\ 100 \\ -150 \end{bmatrix} \frac{m}{s^2}$
 $\|Ускорение\| = 187.08 \frac{m}{s^2}$
 $Сила := Масса \cdot Ускорение = \begin{bmatrix} 1000 \\ 2000 \\ -3000 \end{bmatrix} \text{ N}$

Рис. 1. Задача о движении тела в пространстве (Mathcad Prime)

В расчете, показанном на рис. 1, мы ввели две исходные величины – Ускорение в виде вектора с тремя компонентами (1000, 2000 и -3000 ньютон) и Масса в виде скаляра со значением 20 кг. Затем мы рассчитали длину вектора ||Сила|| – определили абсолютное значение силы, приложенной к телу, и вектор Ускорение (три его составляющие по осям x, y и z). Многие пользователи Mathcad ошибочно полагают, что длина вектора – это число компонентов в нем, а не квадратный корень из суммы квадратов значений компонентов. Эта путаница связана и с тем, что в среде Mathcad число компонентов вектора подсчитывает встроенная функция с именем длина. Оператор же вычисления «настоящей» длины вектора (нормы вектора) $\|\mathbf{v}\|$ появился только в Mathcad Prime. Примечание: Мы воспользовались еще одним удобным инструментом *таблицей*, введенной в Mathcad Prime. Введенную прямоугольную таблицу размером 3 на 2 (три строки и два столбца — см. правую часть рис. 1) мы заполнили частично, дав силе три значения (вектор), а массе только одно (скаляр).

На рис. 2 показано решение в среде предыдущей версии Mathcad – Mathcad 15 нашей задачи об ускорении тела.

$ORIGIN := 1 \quad x := 1 \quad y := 2 \quad z := 3$
 $Длина(v) := \sqrt{\sum_{i=1}^{длина(v)} (v_i)^2}$
 Матрица: $\begin{bmatrix} \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \end{bmatrix}$, \times_n , \times^{-1} , $|\times|$, Индекс []
 $Сила_x := 1000 \text{ N} \quad Сила_y := 2000 \text{ N} \quad Сила_z := -3000 \text{ N} \quad Масса := 20 \text{ kg}$
 $длина(Сила) = 3 \quad Длина(Сила) = 3741.7 \text{ N}$
 $Ускорение := \frac{Сила}{Масса} = \begin{pmatrix} 50 \\ 100 \\ -150 \end{pmatrix} \frac{\text{m}}{\text{s}^2} \quad Длина(Ускорение) = 187.08 \frac{\text{m}}{\text{s}^2}$
 $Сила := Масса \cdot Ускорение = \begin{pmatrix} 1000 \\ 2000 \\ -3000 \end{pmatrix} \text{ N} \quad Длина(Масса \cdot Ускорение) = 3741.66 \text{ N}$

Рис. 2. Задача о движении тела в пространстве (Mathcad 15)

В варианте решения задачи, показанном на рис. 2, значение силы вводится не вектором с тремя компонентами, как на рис. 1, а по координатно: $Сила_x$, $Сила_y$ и $Сила_z$. Для этого в выделенной области расчета (см. верхнюю часть рис. 2), во-первых, системной переменной $ORIGIN$ присвоено значение единицы (по умолчанию оно равно нулю – почему, скажем ниже), а во-вторых, переменным x , y и z присвоены значения 1, 2 и 3, соответственно. Это сделано для того, чтобы компоненты вектора $Сила$ обозначать не

как Сила₁, Сила₂ и Сила₃, а как Сила_x, Сила_y и Сила_z, что более соответствует «физике» задачи – разложение вектора по трем координатам x, y и z.

Глядя на рис. 1 и 2 можно представить себе реальную силу (стрелку-вектор) в 3741.7 ньютон, приложенную к телу и раскладывающуюся по координатам x, y и z со значениями 1000, 2000 и -3000 ньютон, соответственно. Такие векторы-стрелки часто рисуют в учебниках и задачниках по физике, иллюстрируя силы, направления движения, электрические и магнитные поля и прочие векторные физические величины. Кстати, в различных популярных изданиях по физике к векторам силы привязывают... Лебедя, Рака и Щуку из знаменитой басни Крылова, и анализируют, смог бы воз сдвинуться с места.

Второй тип данных, который в среде Mathcad имеет также форму вектора, но не является по сути таковым – это массив данных в одном столбце. Что это такое, мы поясним рис. 3, где показано решение задачи по сглаживанию точек прямой линией.

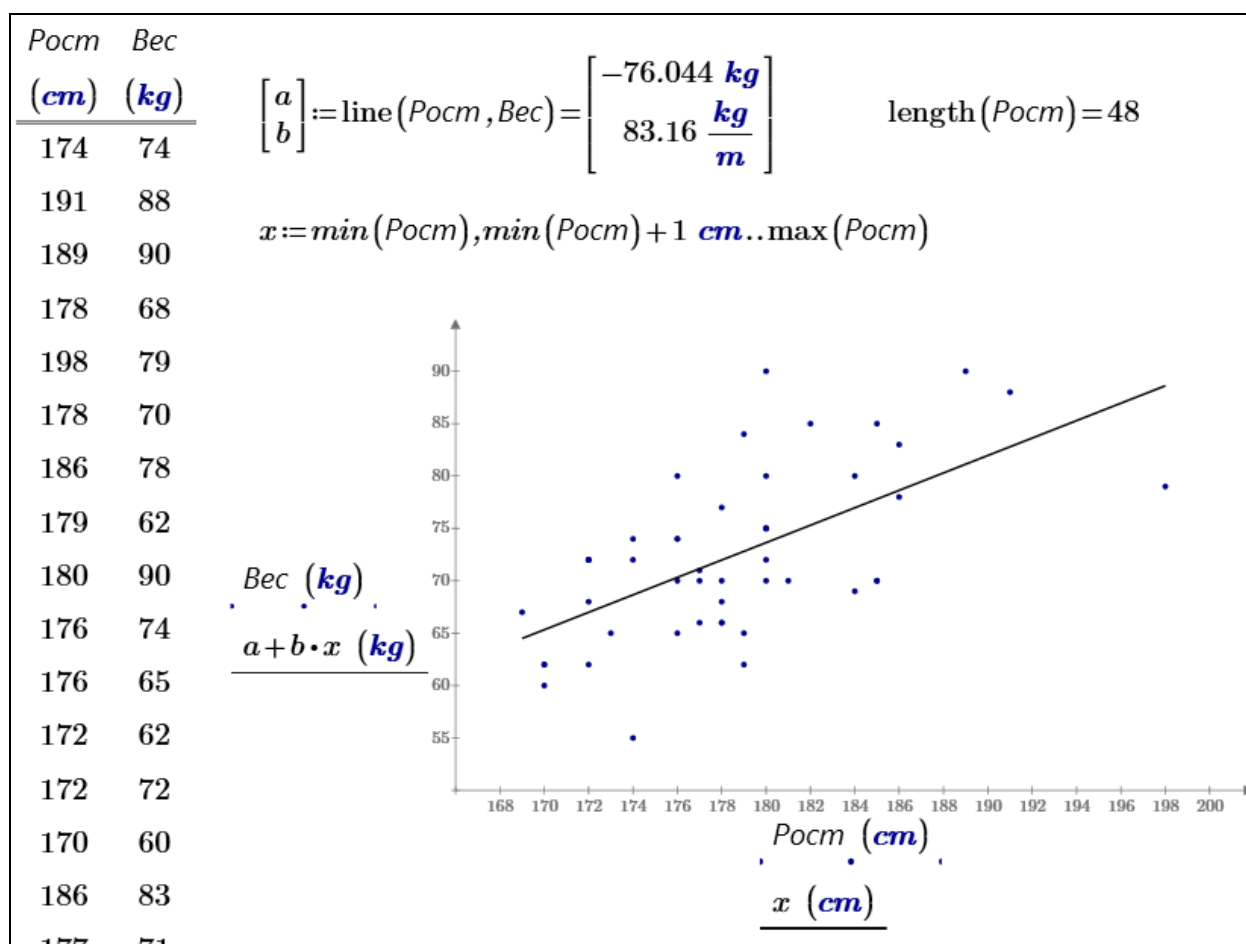


Рис. 3. Сглаживание точек в среде Mathcad

Суть задачи такова. На одной из своих лекций по Информатике первый автор этой статьи сделал такую необычную переключку студентам-юношам (один из них – второй автор; студент-девушки в этом опросе не участвовали). Каждый студент после оглашения его фамилии вставал и называл свой рост и вес. Эти данные вводились в расчет, показанный на рис. 2, двумя векторами, объединенными в таблицу. Темой же

лекции была обработка статистических данных, в частности, линейная регрессия – проведение через рассеянные точки прямой линии. Вот мы на лекции эти данные по росту и весу студентов («рассеянные точки») и обработали – получили параметры прямой линии, сглаживающие эти точки. На сайте <http://communities.ptc.com/videos/3543> можно видеть анимацию этой переключки: через две точки первых двух студентов была проведена прямая линия (интерполяция), потом на графике появлялись новые точки по новым студентам, прямая линия при этом стала колебаться вблизи точек (аппроксимация-сглаживание), а потом стабилизировалась... Переменные a и b (два параметра прямой линии) при этом стали константами.

Координаты сглаживаемых точек в задаче на рис. 3 хранятся в двух массивах данных с именами **Рост** и **Вес** (под весом мы, конечно имеем ввиду, массу-скаляр, а не силу-вектор). Эти массивы по форме, повторяем, являются векторами, но это не векторы в «физическом» смысле того слова, который подчеркнут на рис. 1 и 2. У векторов **Рост** и **Вес** в задаче, показанной на рис. 3, конечно, можно посчитать длину (квадратный корень из суммы квадратов компонентов), но эта величина не будет иметь никакого «физического» смысла, того смысла, который ясно просматривается в задаче, показанной на рис. 1 и 2. На рис. 3 фактически показана база данных с двумя полями (**Рост** и **Вес**) и с 48 записями (на рис. 3 показаны не все записи), которую мы храним в виде двух векторов одного размера. Эту базу данных можно отображать графически, обрабатывать статистически (что и показано на рис. 3), сортировать по полям, фильтровать по записям и т.д., но к какому-то n -мерному пространству эти «вектора» никакого отношения не имеют.

Второе отличие массивов, показанных на рис. 1 и 2, с одной стороны, и на рис. 3, с другой, в том, что нумерация *компонентов* вектора, как правило, начинается с единицы, а *элементов* массива – с нуля. На рис. 3 эта установка (**ORIGIN=0**) зафиксирована умолчанием. Еще одно отличие: составляющие векторов, показанных на рис. 1 и 2, обычно называют *компонентами*, а векторов, показанных на рис. 3, – *элементами*.

Как, наверно, уже отметил читатель, задачи, отображенные на рис. 1, 2 и 3, решаются с использованием *единиц физических величин*: метров (**m**), секунд (**s**), килограммов (**kg**) и др. Это очень мощный инструмент пакета Mathcad [2], позволяющий эффективно контролировать правильность расчетов (не складывать метры с килограммами, грубо говоря) и организовать удобный ввод-вывод численных значений переменных. Но тут встает интересный вопрос в технологическом и смысловом плане – может ли массив/вектор Mathcad иметь элементы с различной размерностью, когда, например, один элемент имеет размерность длины, второй – времени, а третий вообще безразмерный. Традиционный Mathcad, например, Mathcad 15 такой возможности не

допускал. В среде Mathcad Prime это стало допустимым. Технологическая сторона вопроса тут решена, а что со смысловой стороной? Есть ли реальные математические задачи, решение которых требует использования векторов и массивов с различной размерностью элементов?! На рис. 4 такая задача показана: даны две точки на плоскости с координатами x_1-y_1 и x_2-y_2 , необходимо найти значения коэффициентов a и b уравнения прямой $y(x) = a + b \cdot x$, проходящей через эти две точки (пришли два первых студента со своим ростом и весом – см. задачу на рис. 3). На рис. 4 показано, как можно эту задачу решить в среде Mathcad Prime.

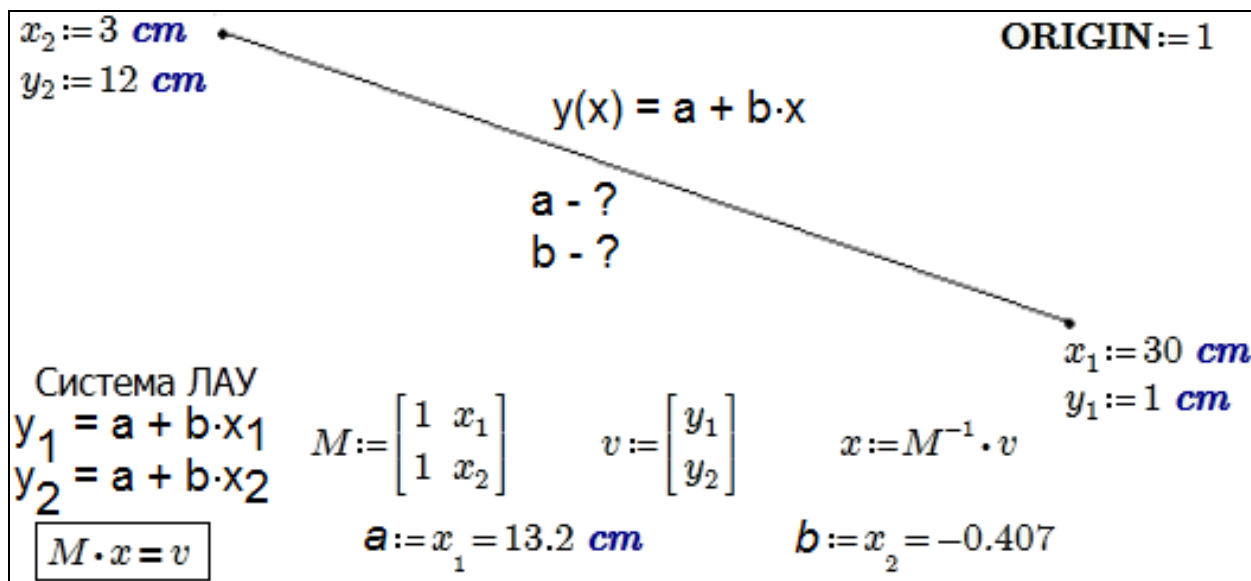


Рис. 4. Задача об уравнении прямой линии

Задачу о прямой линии, проходящей через две точки на плоскости, можно свести к решению системы двух линейных алгебраических уравнений (ЛАУ) в векторно-матричной форме $M \cdot x = v$, где M – матрица коэффициентов при неизвестных x , а v – вектор свободных членов. Решение системы – это нахождение значений вектора x , состоящего из двух компонентов, первый из которых хранит величину с размерностью длины, а второй безразмерный. Матрица M также хранит разноразмерные величины, одни из которых (первый столбец) безразмерные, а другие (второй столбец) имеют размерность длины. Здесь также возникает вопрос – можно ли конструкции v и x , задействованные в задаче на рис. 4, называть векторами с тем «физическим» смыслом, который просматривается в этом слове в задаче, показанной на рис. 1 и 2. Нет, конечно! Тем не менее, в курсе высшей математики под названием «Линейная алгебра», один из разделов которого посвящен решению систем линейных алгебраических уравнений, конструкции x и v называются векторами, а конструкцию M – матрицей. Кстати о матрицах. Если вектор – это по критериям классического программирования (BASIC, Pascal и др. языки) одномерный массив, то матрица – двухмерный.

В программировании используют и *многомерные* (трех-, четырех- и т.д.) массивы. Если скаляр можно уподобить точке, вектор – отрезку прямой линии, а матрицу – прямоугольнику, то трехмерный массив – параллелепипеду. В среде Mathcad эту объемную конструкцию можно создать в виде *вложенного массива*, пример которого показан на рис. 5.

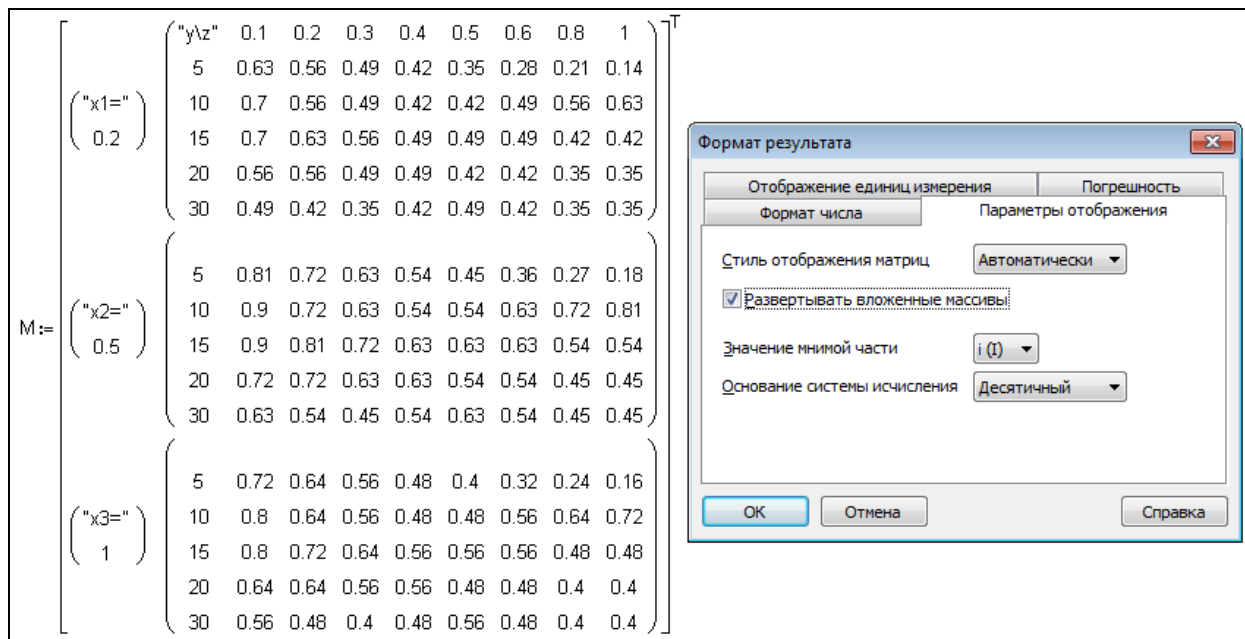


Рис. 5. Пример вложенного массива в Mathcad

Вложенный массив – это вектор или матрица («простой» массив), один или более элементов которого не скаляр, а вектор или матрица. Если говорить о «физике» решаемых в среде Mathcad задач, то тут нужно упомянуть *тензор*, который соответствует трех- и более мерному массиву.

Послесловие

Если вернуться к началу статьи, где упомянут опыт общения автора со студентами, изучающими информатику в школе и вузе, то можно отметить такой феномен. Автор читает студентам курс информатики почти четверть века (см. <http://twf.mpei.ac.ru/ochkov/Potoki.htm>). Двадцать лет назад бывшие школьники, а теперь студенты были очень заинтересованы в изучении программирования, они, как они выражаются «тащились» от программирования. Теперь же у многих студентов на программирование возникла некая «аллергия». Причин у этого печального явления, как мне кажется, две. Первая в том, что это общешкольная проблема, связанная с неприятием того, что преподают в школе. У многих людей, например, наблюдается стойкое отвращение к книгам, какие изучали в школе. Двадцать лет назад программирование в школе почти не изучали, поэтому в те времена у студентов был огромный интерес к этому предмету. А теперь его, увы, нет.

Вторая причина потери у школьников, а потом и у студентов интереса к программированию на мой взгляд заключается в том, что преподаватели часто дают студентам для программирования уж очень «занудные» задачи. Эти задачи, может быть, лет двадцать назад были сами по себе интересны. Но сейчас для решения таких задач появились новые средства, исключаящие традиционное программирование. И школьники-студенты это интуитивно чувствуют или даже прямо знают об этом. Конкретный пример. На рис. 2 можно видеть, как современными средствами Mathcad (оператором суммы) была создана функция пользователя с именем ДЛИНА, возвращающая «физическую» длину вектора. Но многие преподаватели информатики в школе и в вузе по-прежнему предлагают создавать такие и другие подобные функции средствами традиционного программирования (BASIC, Pascal, C и др.), не зная о том, что в современных программных оболочках это уже давно реализовано встроенными функциями и операторами. Кстати, функцию пользователя с именем ДЛИНА можно создать средствами программирования Mathcad (см. рис. 6) с помощью цикла for (цикла с параметром).

```

ORIGIN ≡ 1
Length(v) := "Длина вектора"
Sum ← 0
for i ∈ 1..длина(v)
    Sum ← Sum + (vi)2
return √Sum

Сила := ( 1000 N
          2000 N
          -3000 N )
длина(Сила) = 3
Length(Сила) = 3741.66 N

```

Рис. 6. Программа в среде Mathcad

В Mathcad есть, например, очень интересная функция `match(z, A)`, возвращающая координаты элемента со значением z в матрице A . Если элементов со значением z в матрице A окажется несколько, то функция `match` вернет тот самый вложенный массив (см. рис. 5) – вектор, элементы которого будут новыми векторами с двумя элементами – с номером строки и номером столбца матрицы A , где «сидит» величина z . Можно, конечно, поручить школьникам или студентам составить соответствующую программу с циклами и альтернативами, реализующую функцию `match`. Но лучше давать студентам более занимательные задачи, задачи, решения у которых еще нет или оно еще не так «вылизано». Заставлять студентов создавать функцию `match` это все равно, что заставить их написать программу расчета синуса через разложение в ряд Тейлора, умалчивая при этом, что синус есть в любом калькуляторе и встроен в любой язык программирования.

Эту проблему можно поставить шире и перенести ее на другие школьные и вузовские дисциплины – физику [3], математику [4], химию [5].

Дело в том, что изучение многих учебных дисциплин ориентировано на докомпьютерные и даже докалькуляторные методы решения задач. Возьмем, к примеру, физику, с которой началась наша статья. Решая задачу по этой дисциплине, школьник или студент должен вспомнить или найти в учебнике формулу или набор формул, по которой можно решить конкретную задачу. Сами же эти формулы являются, как правило, частными случаями решения систем уравнений (алгебраических, дифференциальных, интегральных и др.), отображающих фундаментальные законы сохранения вещества, энергии; перехода энергии из одной формы в другую и т.д. и т.п. Упор на использование готовых решений, а не на постановку задачи в общем виде был сделан когда-то давно по простой и понятной причине. В те времена, когда создавались методики решения этих задач, не было эффективных и доступных средств решения уравнений и систем, отображающих фундаментальные законы физики, химии и других научных (учебных) дисциплин. Теперь же такие средства появились. И это требует кардинального пересмотра методик решения задач и переписывания почти всех учебников и задачников.

Конкретный пример. Нужно рассчитать, на какую высоту поднимается брошенный вертикально вверх камень [3]. Решить данную задачу можно через «выуженную» из памяти или справочника (учебника, задачника) формулу, где эта высота высчитывается с учетом начальной скорости и ускорения свободного падения. Если же задачу усложнить – бросать камень не строго вертикально, а под некоторым углом к вертикали, то нужно будет искать новые формулы. Но можно подойти к задаче по иному – видеть в ней систему дифференциальных уравнений и решать ее аналитически (получить те же формулы с частными решениями) или численно.

В этом случае не нужно будет искать формулы – нужно будет только знать фундаментальные законы природы, уметь переводить их на язык математики (составлять те же дифференциальные и прочие уравнения) и решать их современными компьютерными средствами. Еще одно преимущество данного подхода состоит в том, что с задач можно снять многие ограничения, снижающие ценность результатов. Так в задаче о полете камня можно учесть сопротивление воздуха, изменение его плотности по высоте, изменение значения ускорения свободного падения по высоте, кривизну земной поверхности и многое другое. Все это можно учесть, если перейти от использования набора формул к решению исходных уравнений. На рис. 7 в качестве примера показано, как можно «изящно» решить задачу о разгоне автомобиля, имеющего массу m , площадь поперечного сечения S и движущемся в воздухе с плотностью ρ при коэффициенте трения k .

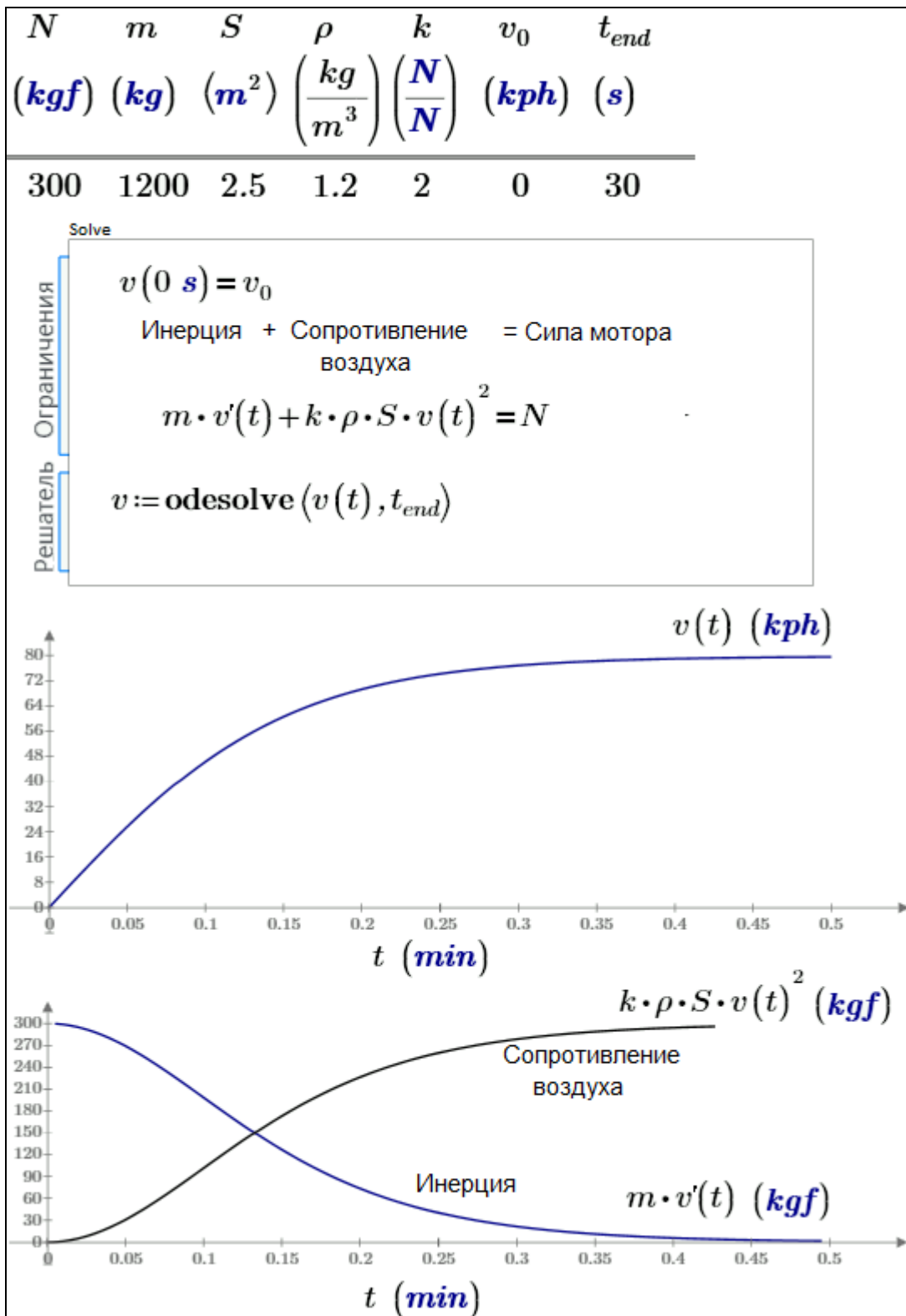


Рис. 7. Задача о старте автомобиля

В задаче при необходимости можно также учесть изменение движущей силы N во времени, сопротивление качения колес по дороге и другие факторы.

И последнее. Один из авторов рассказал учителям одной очень хорошей московской школы о своих наработках по использованию математических пакетов в преподавании математики, физики, химии [3-5]. Так вот, эти учителя мне сказали, что все

это очень хорошо и заманчиво, но наша школа сейчас называется по-другому – не «Средняя общеобразовательная школа номер такой-то», а «Курсы по подготовке к сдаче ЕГЭ по математике, физике, химии...». Вопросы же для ЕГЭ специально составляются так, чтобы при ответах на них не нужно было использовать не только компьютер, но даже и простейший калькулятор... Хотя в настоящее время студент, инженер, ученый и даже простой офисный работник не может обойтись без этих вычислительных средств. (Это к вопросу об отрыве школы от реальной жизни). Раньше, до эры ЕГЭ учитель мог ориентироваться на сильных в математике учащихся и давать на уроках интересные, творческие задачи, выходящие за рамки учебников. Слабым же ученикам можно было, несколько кривя душой, ставить «тройки», особо не напрягая их. Главное, чтобы они не мешали остальным. Теперь же приходится всех «стричь под одну гребенку»: натаскивать всех без исключения на сдачу ЕГЭ, по результатам которых судят о работе самого учителя и о шансах учеников на поступление в тот или иной вуз. Это еще один камень в огород ЕГЭ. Да, система ЕГЭ подтянула слабых учителей и учеников к среднему уровню. Но она также, увы, опустила сильных учителей и учеников к этому самому пресловутому среднему уровню.

Да, для страны, превращающейся в сырьевой придаток других развитых стран, такая образовательная модель является самой оптимальной. Но если же мы не на словах, а на деле хотим слезть с «нефтяной иглы» и сохранить статус научной державы, то образовательную модель, которую нам навязывают чиновники от образования, нужно менять.

Представим себе такой ЕГЭ по математике, физике, химии... Тестируемый (школьник, студент, или кто-то другой, занимающийся самообразованием, в том числе и через Интернет) подсаживается к компьютеру, который выдает задание. Выполнить же это задание можно всеми возможными способами: решить задачу в уме или на листе бумаги, решить задачу в среде какой-нибудь компьютерной программы, поискать аналогичные решения в Интернете, вывесить задание на каком-либо форуме и ждать ответа, послать задание другу, организовать голосование Интернет-сообщества по вариантам ответов, если они даны и т.д. и т.п. Компьютер же после всестороннего анализа действий тестируемого должен сам решить, какую оценку ему поставить.

Выводы

Без четкого понимания таких базовых понятий как скаляр, вектор, матрица, таблица, массив (одно-, двух- и многомерный), база данных, база знаний и т.д. освоение информатики будет идти большим трудом. На «курсах по подготовке к сдаче ЕГЭ по математике, физике, химии...», т.е. в средней школе можно и нужно организовать кружки

под названием «Курсы по подготовке к сдаче ЕГЭ по математике, физике, химии... с помощью компьютера», на которых можно будет разбирать задачи приведенные, например, в [3-5].

Литература:

1. Очков В.Ф. Mathcad 14 для студентов и инженеров: русская версия БХВ-Петербург, 2009 (http://twt.mpei.ac.ru/ochkov/Mathcad_14/RusIndex.html)
2. Очков В.Ф. Физические и экономические величины в Mathcad и Maple (Серия «Диалог с компьютером»). М.: Финансы и статистика, 2002 (http://twt.mpei.ac.ru/ochkov/Units/Forword_book.htm)
3. Очков В.Ф. Задачи по физике: новый подход к решению // Открытое образование № 6, 2012 (<http://twt.mpei.ac.ru/ochkov/Mathcad-15/Physic.pdf>)
4. Очков В.Ф. Преподавание математики и математические пакеты // Открытое образование, №2, 2013 (<http://twt.mpei.ac.ru/ochkov/Mathcad-15/OchkovMath.pdf>)
5. Коробов В.И., Очков В.Ф. Химические расчеты в среде Mathcad. Из-во Днепропетровского национального университета, 2012 (http://www.exponenta.ru/educat/news/ochkov/book_ochkov11.asp)