электронный научно-технический журнал
ИНЖЕНЕРНЫЙ ВЕСТНИК
Издатель ФГБОУ ВПО "МГТУ им. Н.Э. Баумана". Эл No. ФС77-51036. ISSN 2307-0595

Генератор контрольных заданий по высшей математике: опыт создания и применения

04, апрель 2015

Коновалов Я. Ю.[1,*], Соболев С. К.[1]

УДК: 378.016+512.643

[1]Россия, МГТУ им. Н.Э. Баумана
[*]kon20002000@mail.ru

Введение

Проблема автоматического генерирования заданий по математике для студентов стала в последние 15 лет особенно актуальной [1–5]. Совершенство электронных средств коммуникации достигло такого уровня, что комплект контрольных заданий через несколько недель, а то и дней после своего появления становится известен студентам вместе с готовыми решениями, правда, иногда ошибочными. Подготовка новых качественных комплектов заданий — довольно кропотливая работа. В связи с этим создание генератора, способного выдать сколь угодно много новых вариантов заданий вместе с ответами (как окончательными, так и промежуточными) является очень важной задачей для методических коллективов кафедр вузов.

При этом к условиям задач в формируемых генератором заданиях предъявляются следующие требования.

1. Задачи должны иметь относительно «хорошие» ответы и допускать решение без трудоемких вычислений.

2. Числовые данные в условиях задач не должны быть «странными», скажем, нежелательно, чтобы все они были отрицательны или кратны 77.

3. Условия задач должны быть корректны.

Требования к программе составления заданий таковы.

1. Простота и быстрота использования.

2. Формирование качественных вариантов заданий, удовлетворяющих приведенным выше требованиям и не требующих дополнительной проверки.

3. Автоматическая подготовка заданий к выдаче студентам (верстка для печати вариантов или вывод на мониторы для электронного тестирования).

4. Автоматическая подготовка удобных для проверки (достаточно подробных) ответов.

Рассмотрим возможные способы автоматического составления заданий.

Простейшим вариантом является выбор случайным образом или по некоторым правилам задач из некоторой заранее созданной базы. Очевидное преимущество такого способа — возможность вводить в базу абсолютно любые задачи и вопросы даже по предметам, не связанным с математикой. Однако такой способ имеет и недостатки. Если число задач в базе невелико, результаты будут не намного лучше, чем при использовании неизменных вариантов. В то же время составление обширной базы является трудоемкой задачей. Если база достаточно велика, управление ею превращается в отдельную проблему, не только техническую, но и организационную. Если базу составляют сразу несколько преподавателей, необходимо синхронизировать результаты их работы и централизованно обновлять данные на компьютерах, где эта база используется. Поскольку в случае хищения и несанкционированной публикации база задач потеряет свою ценность, а при случайной ее порче нарушится работа генератора, возникает необходимость в мерах по контролю целостности базы и доступа к ней.

Принципиально иным является способ, при котором задачи генерируются с помощью специальных алгоритмов на основе случайных чисел непосредственно в процессе формирования задания. При этом отпадает необходимость в централизованной базе задач и связи с ней. Благодаря использованию в алгоритмах случайных чисел при каждом запуске генерируются новые задачи с новыми ответами. Дважды запустить программу так, чтобы повторились все сгенерированные задачи и ответы на них, практически невозможно. Таким образом, конфиденциальность новых вариантов не нарушается даже в случае публикации не только ранее сгенерированных вариантов, но и самой программы. Эта особенность может быть полезна при проведении важных аттестационных мероприятий.

При автоматической генерации задач исключается возможность появления ошибок и опечаток. Кроме того, можно составить гораздо более подробные ответы. Главным недостатком подобного подхода является сложность создания таких алгоритмов генерации, которые одновременно обеспечивали бы достаточное разнообразие задач и их соответствие приведенным выше требованиям. Предложенный авторами подход к построению подобных алгоритмов и некоторые их примеры описаны далее.

Возможны и различные гибридные подходы, например, когда задачи для базы генерируются автоматически, а затем преподаватели просматривают их, отсеивая неудачные.

Параметрические схемы

Алгоритмы генерации задач, в которых все данные задачи и ответ явно зависят от некоторого набора числовых параметров, будем называть параметрическими схемами. Например, квадратное уравнение $a(x-x_1)(x-x_2)=0$ имеет корни x_1 и x_2. Каждому набору x_1, x_2 и a соответствуют свои уравнение, решение и ответ. Таким способом легко составлять задачи, решаемые с помощью стандартных формул. В работах [1–3] описан генератор заданий, основанный на параметрических схемах, и предложен язык для их описания. Такой подход позволяет перейти от базы задач к базе параметрических схем, составление которых менее трудоемко. Помимо простоты реализации параметрические схемы позволяют легко обеспечить выполнение приведенных выше требований к задачам путем ограничения диапазона значений параметров.

Рассмотрим примеры параметрических схем, применяемых авторами. Для задачи о нахождении неопределенного интеграла вида

$$J = \int \frac{Ax + B}{\sqrt{\alpha x^2 + \beta x + \gamma}} dx \quad (\alpha \neq 0)$$

параметрами служат коэффициенты условия. Предполагается, что $\alpha \neq 0$, подкоренное выражение не является полным квадратом ($D = \beta^2 - 4\alpha\gamma \neq 0$), его производная не пропорциональна числителю $2\alpha B \neq \beta A$ и $D > 0$, если $\alpha < 0$. Взяв интеграл в общем виде, получим ответ:

$$J = \frac{A}{\alpha}\sqrt{\alpha x^2 + \beta x + \gamma} + \frac{2\alpha B - \beta A}{2\alpha\sqrt{\alpha}} \ln\left|\left(x + \frac{\beta}{2\alpha}\right) + \sqrt{x^2 + \frac{\beta}{\alpha}x + \frac{\gamma}{\alpha}}\right| \text{ при } \alpha > 0;$$

$$J = \frac{A}{\alpha}\sqrt{\alpha x^2 + \beta x + \gamma} + \frac{\beta A - 2\alpha B}{2\alpha\sqrt{-\alpha}} \arcsin\left(\frac{2\alpha\, x + \beta}{\sqrt{\beta^2 - 4\alpha\gamma}}\right) \text{ при } \alpha < 0.$$

Таким образом, эта параметрическая схема представляет собой задачу, решенную в общем виде.

При генерации задачи об интегрировании рациональной дроби с кубическим многочленом в знаменателе нужно, чтобы хотя бы один из корней знаменателя мог быть найден подбором. Проще всего добиться этого путем явного задания коэффициентов простейших дробей. Условие задачи получают приведением дробей к общему знаменателю. Таким образом, в этой параметрической схеме условие является функцией ответа.

Алгоритм генерации условий задачи о диагонализации квадратичной формы ортогональным преобразованием, приведенный в [4], представляет собой параметрическую схему. Разновидностью параметрических методов является и аналитический подход к генерированию матриц специального вида [5].

Непараметрические алгоритмы

Параметрические схемы эффективны только в том случае, когда зависимость условий и ответов от параметров достаточно проста. Рассмотрим задачу об ортогонализации системы векторов. Зададим координаты векторов элементами матрицы требуемого ранга. Для получения точного ответа проведем ортогонализацию с помощью целочисленной модификации метода Грама — Шмидта [4]. Полученные в результате ортогональные векторы будут однозначно определены координатами исходных векторов, но зависимость эта достаточно сложна.

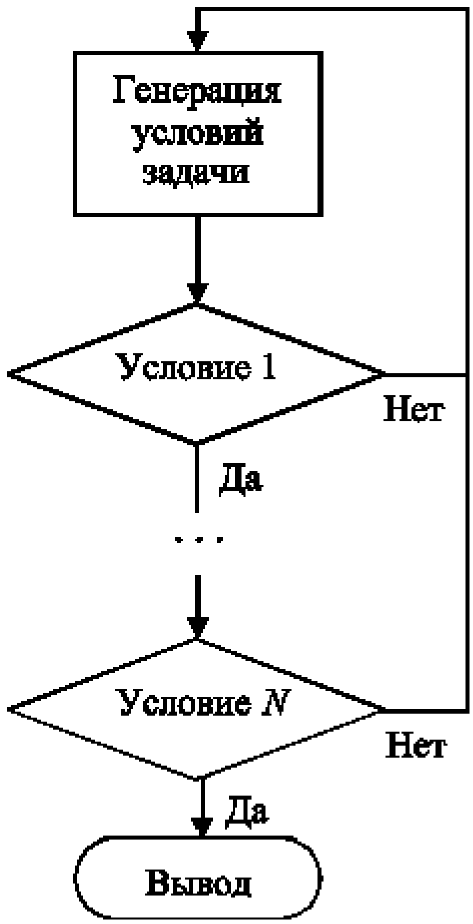

Рис. Общая схема алгоритма генерации задачи с проверками

Алгоритмы (в том числе детерминированные) генерации задач, в которых не используется явная зависимость условий и ответов от параметров, будем называть *непараметрическими*. Для таких алгоритмов явная зависимость между условиями и параметрами неизвестна, поэтому оценить соответствие условий задачи приведенным выше требованиям можно только после генерации условий. Если полученная задача не удовлетворяет условиям, выполняется повторная ее генерация. Если процесс генерации и проверка выполняются достаточно быстро, алгоритм в целом получается эффективным (см. рисунок).

Такая структура алгоритма является общей для всех непараметрических методов генерации задач. Несмотря на кажущуюся неэффективность, она имеет свои преиму-щества, так как нет необходимости придумывать способ генерации, гарантирую-щий качество полученного результата. Достаточно лишь предложить алгоритм, относительно часто выдающий хорошие условия, и грамотно организовать проверку, чтобы отсечь неудачные.

Заметим, что и для параметрических схем иногда приходится выполнять проверку. Это может быть вызвано тем, что к условиям задач предъявляются особенно жесткие требования, как в [6], или тем, что для гарантированного достижения «хороших» значений приходится настолько сильно ограничивать значения параметров, что число генерируемых задач неприемлемо уменьшается.

Практическая реализация

Кратко опишем некоторые алгоритмы, примененные авторами в автоматических генераторах контрольных работ и домашних заданий по аналитической геометрии. Подробное описание этих и некоторых других алгоритмов приведено в [4]. Согласно учебным программам МГТУ им. Н.Э. Баумана, контрольная работа по теме «Матрицы и системы линейных алгебраических уравнений (СЛАУ)» содержит следующие задачи: построить неполную кривую второго порядка; вычислить значение матричного многочлена; найти ранг матрицы в зависимости от параметра; решить матричное уравнение; найти все решения системы линейных уравнений с неполным рангом.

Алгоритмы генерации первых двух задач представляют собой простейшие параметрические схемы. Для генерации третьей — пятой задач применяются непараметрические алгоритмы, основанные на свойствах элементарных преобразований сохранять ранг и невырожденность матриц.

Рассмотрим алгоритм генерации матричных уравнений вида $AX = B$, где A и B — известные квадратные матрицы, а X — искомая, при этом A — невырожденная матрица с небольшим (для простоты вычислений) определителем. Чтобы студент не мог случайно получить правильный ответ, неправильно обратив матрицу A, она должна быть несимметричной и ее определитель не должен быть равен единице. Зададим ее определитель небольшим случайным числом $|A| = d$, отличным он нуля и единицы. Определитель верхнетреугольной матрицы, у которой один из диагональных элементов равен d, а остальные — единице (звездочкой обозначены случайные числа):

$$\begin{pmatrix} d & & & * \\ 0 & 1 & & \\ & & \ddots & \\ & 0 & & 1 \end{pmatrix},$$

а также равен d.

Подвергнем эту матрицу серии элементарных преобразований, прибавляя к случайно выбранной строке (столбцу) другую случайно выбранную строку (столбец), умноженную на случайный коэффициент. Назовем такую процедуру *перемешиванием.* Полученная матрица A сохранит определитель d. Проверим, что элементы полученной матрицы не очень велики, она не содержит избыточного числа нулей и несимметрична. Если полученная матрица не удовлетворяет этим требованиям, генерируем ее заново. Матрицу X составляем из случайных чисел, находим матрицу B как $B = AX$ и проверяем ее на наличие избыточно больших элементов. Аналогичным образом составляются матричные уравнения вида $XA = B$ и $AXB = C$.

Практика показала, что иногда при вычислении определителя A возникают большие числа, что значительно затрудняет решение задачи. Такая ситуация вполне допустима для домашнего задания, но крайне нежелательна в контрольной работе. В связи с этим в алгоритм генерации матрицы A была добавлена дополнительная проверка простоты вычисления определителя. Избыточно трудоемкие задачи встречались относительно редко, поэтому существенного влияния на эффективность алгоритма эта проверка не оказала.

Таким образом, можно отметить еще одно преимущество организации алгоритма по схеме, представленной на рисунке. Если имеющийся алгоритм не позволяет получить достаточно качественные задачи, можно попробовать исправить дело введением дополнительной проверки, вместо того чтобы создавать новый алгоритм.

Для генерации условий третьей задачи составим матрицу A, состоящую из единичной матрицы порядка r, прямоугольной матрицы случайных чисел $k_{i,j}$ из r строк и $m-r$ столбцов, строки из нулей и элемента $\lambda-\lambda_0$ на месте с номером $q>r$ и $n-r-1$ нулевых строк. Ранг этой матрицы равен r при $\lambda=\lambda_0$ и $r+1$ при $\lambda\neq\lambda_0$:

$$A=\left(\begin{array}{ccc:ccccc}
1 & & 0 & & & & & \\
 & \ddots & & & & k_{i,j} & & \\
0 & & 1 & & & & & \\
\hdashline
 & 0 & & 0 & \cdots & \lambda-\lambda_0 & \cdots & 0 \\
\hdashline
 & 0 & & & & 0 & &
\end{array}\right).$$

Прибавим к $(r+1)$-й строке случайную линейную комбинацию первых r строк. Все строки, кроме $(r+1)$-й, подвергнем перемешиванию. Полученная матрица также имеет ранг r при $\lambda=\lambda_0$ и $r+1$ при $\lambda\neq\lambda_0$. Поделим каждую строку на наибольший общий делитель ее элементов. Проверим число нулей, отсутствие больших элементов, пропорциональных строк и столбцов. Кроме того, строка и столбец полученной матрицы, содержащие λ, не должны быть пропорциональны другим строкам и столбцам после отбрасывания элементов, стоящих на месте q (для строк) и $r+1$ (для столбцов) соответственно. Несоблюдение этих требований недопустимо упрощает задачу.

Тот же метод лежит в основе алгоритма генерации задач на решение СЛАУ. Пусть требуется составить СЛАУ из n строк и m столбцов ранга r ($r\le m$, $r\le n$). Построим расширенную матрицу системы в ступенчатом виде:

$$\left(\begin{array}{ccc:c|c}
1 & & 0 & & b_1 \\
 & \ddots & & k_{i,j} & \vdots \\
0 & & 1 & & b_r \\
\hdashline
 & 0 & & 0 & 0
\end{array}\right).$$

Решение этой системы уравнений:

$$X = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} = \sum_{j=1}^{m-r} c_j \begin{pmatrix} -\mathbf{k}_j \\ 0 \\ \vdots \\ 1_{r+j} \\ \vdots \\ 0 \end{pmatrix} + \begin{pmatrix} b_1 \\ \vdots \\ b_r \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \qquad \text{где } \mathbf{k}_j = \begin{pmatrix} k_{1j} \\ \vdots \\ k_{rj} \end{pmatrix},$$

а 1_{r+j} означает, что элемент 1 стоит на ($r+j$)-м месте. Подвергнем строки матрицы перемешиванию. Ответ при этом не изменится. Для однородной СЛАУ задаем $b_j = 0$. Как и в предыдущих задачах, следует проверить отсутствие пропорциональных строк и столбцов, а также избытка нулей.

Определители, СЛАУ и системы векторов заданного ранга широко используют при решении многих задач аналитической геометрии. Это позволяет применять для их генерации описанные выше алгоритмы.

На основании алгоритма генерации матриц с заданным определителем несложно построить невырожденные матрицы, столбцы (или строки) которых можно рассматривать как координаты векторов базиса. На этом свойстве основаны алгоритмы генерации трехмерных задач.

Рассмотрим, например, задачу о вычислении объема параллелепипеда, заданного координатами своих вершин, и его высоты, опущенной на данную грань. Эта задача входит в домашнее задание и в контрольную работу по теме «Аналитическая геометрия». Для генерации векторов $\overline{AB}$, $\overline{AD}$ и $\overline{AA_1}$, на которые натянут параллелепипед, применим алгоритм генерации матрицы размерности 3 с определителем $d \neq 0$. В качестве вершины A выберем точку с небольшими случайными координатами. Остальные вершины получим, отложив векторы $\overline{AB}$, $\overline{AD}$ и $\overline{AA_1}$ от точки A. Объем параллелепипеда равен определителю d. Площадь основания и высоту находим по стандартным формулам $S_{ABCD} = \left|\overline{AB} \times \overline{AD}\right|$ и $h = V / S_{ABCD}$.

Если в задаче фигурируют два линейно независимых вектора в $\mathbb{R}^2$ или $\mathbb{R}^3$, применять алгоритм генерации матрицы ранга 2 нет необходимости. Проще и быстрее задать их координаты случайными числами, а затем убедиться, что они не пропорциональны. Примерами подобных задач являются задача о вычислении угла между векторами, прямыми или плоскостями, о нахождении уравнения плоскости, проходящей через три точки или точку и прямую, и т. д.

При реализации автоматического генератора заданий на кафедре «Высшая математика» МГТУ им. Н.Э. Баумана авторами ставилась задача обеспечения генерации условий задач и автоматической их верстки для печати. Условия и ответы генерируются написанной на языке C программой в виде файлов в формате TeX. Затем с помощью системы LaTeX осуществляется компиляция этих файлов в файлы *.pdf, пригодные для печати. Для наиболее удобного использования был создан пакетный файл, содержащий команды запуска генератора и компиляции условий и ответов. Таким образом, система позволяет полностью подготовить контрольную работу путем запуска одного пакетного файла. Пример готового к печати листа с четырьмя вариантами представлен на с. 9.

Альтернативные возможности

Рассмотрим другие способы создания автоматического генератора задач. Вместо того чтобы писать собственную программу генерации задач, ее можно реализовать в существующих системах символьной математики, таких, как MatLab или Matematika. Особенно заманчиво использовать их возможности при построении условий задач на дифференцирование и интегрирование. Однако нерешенной пока остается проблема организации проверки качества задачи и автоматической ее коррекции в случае несоответствия требованиям.

Для вывода на печать условий сгенерированных задач оптимальной средой, видимо, является TeX. При разработке программы-генератора его использование дает следующие преимущества. Условия и ответы хранятся в текстовом файле в пригодном для чтения виде, формулы задаются текстовыми командами. Это позволяет при отладке программы анализировать сам сгенерированный файл, а не только результаты его неудачной компиляции. Кроме того, TeX имеет систему информирования об ошибках, схожую с системами, имеющимися в компиляторах языков программирования. Возможность пакетной обработки позволяет использовать процедуру компиляции файла в TeX как один из шагов программы без необходимости запускать TeX вручную. Имеются встроенные средства создания рисунков и графиков, что позволяет снабдить условия и ответы иллюстрациями без использования дополнительного программного обеспечения. TeX автоматически осуществляет верстку, рассчитывая расположение символов, места переноса и т. д. Форматирование задается командами высокого уровня, такими, как центрирование строки, создание таблицы и т. д. Это позволяет осуществить даже довольно сложное форматирование.

Ниже приведена страница с вариантами контрольной работы, подготовленная для печати.

Вариант 0.

1. Построить кривую $y = -2 - \frac{4}{3}\sqrt{x^2 + 8x + 25}$.

2 балла

2. Найти значение многочлена $f(x)$ от матрицы A:

$$f(x) = x^3 - 3x^2 - 4x - 1, \quad A = \begin{pmatrix} -7 & 7 \\ -6 & 6 \end{pmatrix}.$$

2,5 балла

3. Найти ранг матрицы A при различных значениях параметра λ:

$$A = \begin{pmatrix} 1 & -2 & 1 & 1 \\ -7 & 7 & -9 & \lambda \\ 4 & -4 & 5 & -2 \\ -2 & 5 & -2 & -3 \end{pmatrix}.$$

2 балла

4. Решить матричное уравнение

$$\begin{pmatrix} -4 & -1 & 1 \\ -5 & -2 & 1 \\ -5 & -2 & 2 \end{pmatrix} \cdot X = \begin{pmatrix} 10 & -13 & 11 \\ 14 & -13 & 18 \\ 11 & -11 & 16 \end{pmatrix}.$$

3 балла

5. Исследовать систему уравнений и найти ее решение, если оно существует.

$$\begin{cases} x_1 - 3x_2 + x_3 + x_4 = 19 \\ 2x_1 - 5x_2 + 6x_3 + 2x_4 = 25 \\ -2x_1 + 4x_2 - 7x_3 - x_4 = -17 \\ 2x_1 - 3x_2 + 5x_3 - x_4 = 14 \end{cases}.$$

3,5 балла

Вариант 1.

1. Построить кривую $y = -6 - \sqrt{x^2 + 6x + 25}$.

2 балла

2. Найти значение многочлена $f(x)$ от матрицы A:

$$f(x) = -x^2 - 2x + 1, \quad A = \begin{pmatrix} 1 & 1 & 2 \\ 1 & 0 & -2 \\ 2 & 4 & -2 \end{pmatrix}.$$

2,5 балла

3. Найти ранг матрицы A при различных значениях параметра λ:

$$A = \begin{pmatrix} 1 & 8 & \lambda & 6 \\ -2 & 3 & -1 & 7 \\ -1 & -3 & 4 & -1 \\ 1 & -2 & 1 & -4 \end{pmatrix}.$$

2 балла

4. Решить матричное уравнение

$$\begin{pmatrix} -2 & 1 \\ -1 & 3 \end{pmatrix} \cdot X \cdot \begin{pmatrix} 7 & 5 \\ -9 & -7 \end{pmatrix} = \begin{pmatrix} -8 & -4 \\ -29 & -27 \end{pmatrix}.$$

3 балла

5. Исследовать систему уравнений и найти ее решение, если оно существует.

$$\begin{cases} -x_1 + x_2 - 7x_4 = -28 \\ 3x_1 - x_2 - x_3 + 8x_4 = 22 \\ -4x_1 + x_2 + 2x_3 - 7x_4 = -15 \\ -3x_1 + x_2 + 2x_3 - 5x_4 = -14 \end{cases}.$$

3,5 балла

Вариант 2.

1. Построить кривую $x = 6 + \sqrt{-y - 3}$.

2 балла

2. Найти значение многочлена $f(x)$ от матрицы A:

$$f(x) = x^2 - x - 3, \quad A = \begin{pmatrix} -3 & 2 & 1 \\ -3 & -1 & 1 \\ -1 & 2 & 0 \end{pmatrix}.$$

2,5 балла

3. Найти ранг матрицы A при различных значениях параметра λ:

$$A = \begin{pmatrix} 2 & 3 & \lambda & 2 \\ -1 & -1 & 1 & 1 \\ 2 & 2 & -2 & -3 \\ -3 & -4 & 2 & 1 \end{pmatrix}.$$

2 балла

4. Решить матричное уравнение

$$\begin{pmatrix} -2 & -7 \\ -1 & -6 \end{pmatrix} \cdot X \cdot \begin{pmatrix} -5 & 7 \\ 3 & -5 \end{pmatrix} = \begin{pmatrix} -8 & -4 \\ 6 & -22 \end{pmatrix}.$$

3 балла

5. Исследовать систему уравнений и найти ее решение, если оно существует.

$$\begin{cases} 4x_1 - x_2 + x_4 = 13 \\ 8x_1 + 9x_2 - 4x_3 - x_4 = 15 \\ 2x_1 + 5x_2 - 2x_3 - x_4 = 1 \\ -3x_1 - 2x_2 + x_3 = -7 \end{cases}.$$

3,5 балла

Вариант 3.

1. Построить кривую $y = 1 - \sqrt{6x}$.

2 балла

2. Найти значение многочлена $f(x)$ от матрицы A:

$$f(x) = x^2 - x - 2, \quad A = \begin{pmatrix} -1 & 3 & -1 \\ 1 & 0 & 2 \\ -2 & 3 & 1 \end{pmatrix}.$$

2,5 балла

3. Найти ранг матрицы A при различных значениях параметра λ:

$$A = \begin{pmatrix} -1 & 5 & -5 & -4 \\ 1 & 2 & -9 & -3 \\ 5 & \lambda & -3 & 6 \\ 1 & -2 & -1 & 1 \end{pmatrix}.$$

2 балла

4. Решить матричное уравнение

$$\begin{pmatrix} -3 & 1 \\ 5 & -3 \end{pmatrix} \cdot X \cdot \begin{pmatrix} 3 & 2 \\ -4 & -1 \end{pmatrix} = \begin{pmatrix} -4 & 9 \\ -8 & -27 \end{pmatrix}.$$

3 балла

5. Исследовать систему уравнений и найти ее решение, если оно существует.

$$\begin{cases} -4x_1 - 2x_2 - 3x_3 - 3x_4 = 7 \\ -x_1 - 8x_2 + 2x_3 + x_4 = 2 \\ -x_1 + 7x_2 - 5x_3 - 3x_4 = -10 \\ -3x_1 - 3x_2 - 2x_3 - 2x_4 = 3 \end{cases}.$$

3,5 балла

Для пользователя серьезным преимуществом является то, что TeX является свободным программным обеспечением, распространяемым бесплатно, и может работать под разными операционными системами (кроссплатформенность). Таким образом, практически не существует ни административных, ни технических препятствий для установки TeX на любой компьютер.

Если условия планируется выводить на экран компьютера, например для организации интерактивного тестирования, вместо TeX можно использовать html+mathml.

Заключение

Авторами проанализированы различные принципы построения генераторов заданий по математике, их преимущества и недостатки. Описан опыт создания и применения авторами на кафедре «Высшая математика» МГТУ им. Н.Э. Баумана генератора контрольных работ и домашних заданий по курсам аналитической геометрии и линейной алгебры.

Список литературы

1. Карнаухов В.М. Latex-генератор контрольных работ. М.: ФГБОУ ВПО МГУП. 2014. 177 с.
2. Карнаухов В.М., Русаков А.А. Компьютерный способ подготовки раздаточного материала контрольных работ по математике. // Материалы Междунар. науч.-практ. конф. «Информатизация образования-2012». Орел: ООО «Картуш». 2012. 368 с. С. 99–103.
3. Карнаухов В.М. Приложение LATEX. Генератор вариантов контрольных работ. / Монография. Saarbrücken, Germany: LAP LAMBERT Academic Publishing. 2012. 100 с.
4. Коновалов Я.Ю., Соболев С.К., Ермолаева М.А. Методические аспекты автоматической генерации задач по линейной алгебре // Инженерный журнал: наука и инновации. 2013. вып. 5. 14 с. Режим доступа: http://engjournal.ru/catalog/pedagogika/hidden/740.html (дата обращения: 05.12. 2014).
5. Соболев С.К. Генерирование матриц специального вида: аналитический подход. Инженерный журнал: наука и инновации, 2013, вып. 5. URL: http://engjournal.ru/catalog/pedagogika/hidden/745.html (дата обращения 05.12.2014).
6. Шестаков А.П. Использование LaTeX при генерации дидактических материалов по математике. // Информатика. 2008. № 15 (568). с. 13–21. Режим доступа: http://информатика.1сентября.рф/view_article.php?ID=200801503 (дата обращения: 05.12. 2014).