

сдачи работ и оценок, в котором студенты имеют возможность следить за состоянием своей успеваемости и объемом оставшейся до конца семестра работы. Также я требую, чтобы была сделана работа над ошибками в каждой самостоятельной работе. Правильно выполненная работа над ошибками позволяет студенту повысить первоначальную оценку. Из-за слабой математической подготовки студентов факультета, на котором я работаю, не вижу смысла вызывать к доске студентов для решения целой задачи на оценку, как в школе. Я даю каждому индивидуальное домашнее задание для исключения списывания у более сильных одноклассников. Помимо домашних работ студенты выполняют расчетно-графические работы (на листах А4) и аудиторские контрольные работы, требующие большей аккуратности. Таким образом, к концу семестра каждый студент накапливает большое количество оценок за свои индивидуальные работы (каждый по своему варианту), что позволяет достаточно объективно поставить среднюю оценку за семестр, учитывать и минимизировать стресс от экзамена.

УДК 512.86:004.42

ПРОГРАММА ПОШАГОВОГО РЕШЕНИЯ СИСТЕМЫ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ МЕТОДОМ ЖОРДАНА-ГАУССА В СРЕДЕ КОМПЬЮТЕРНОЙ АЛГЕБРЫ MATHEMATICA

П.Г. ЛАСЫЙ

Белорусский национальный технический университет, г. Минск

Система компьютерной алгебры *Mathematica* является одной из самых востребованных в научной среде и образовательном пространстве. Это объясняется исключительной мощностью этого программного продукта, его дружелюбным интерфейсом и наличием многочисленных пакетов приложений в самых различных областях.

Настоящая программа реализует классический метод Жордана-Гаусса решения произвольной системы линейных алгебраических уравнений. Она состоит из восемнадцати модулей, основными являются два: **SolSys** и **StepByStep**. Первый модуль служит для решения системы и вывода общего и базисного решений, второй управляет пошаговым решением системы. В качестве примера приведем код второго модуля:

```

StepByStep[x]
:= Module[{YesNo, ijMatrix, iVector, wAB, pAB, changeBasis, logBasis},
Print[x];
If[! testA, InputA[]; Return[]]; If[MatrixQ[B], B
= Flatten[B]]; If[! testB, InputB[]; Return[]]; OutPutSystem; wAB
= extendedMatrix; ijMatrix = wBasis = pwBasis = {}; YesNo = "Нет";
changeBasis = "Да"; logBasis = False; While[changeBasis == "Да",
While[YesNo == "Нет", i = j = 0;
While[! testij] || (wAB[[i, j]] == 0), ijInput = ijDialogInput;
If[testijInput, i = ijInput[[1]]; j = ijInput[[2]]]; iVector
= Flatten[Cases[ijMatrix, {i, _}]]; If[logBasis || (iVector
≠ {}), pwBasis = wBasis]; If[iVector ≠ {}, wBasis
= Cases[wBasis, Except[iVector[[2]]]]; ijMatrix
= Cases[ijMatrix, Except[iVector]]]; ijMatrix
= ijMatrix ∪ {{i, j}}; If[! logBasis && (iVector == {}), pwBasis
= wBasis]; wBasis
= wBasis ∪ {j}; PrintStr["Номер разрешающей строки i = " <
> ToString[i] << " , номер разрешающего столбца j = " <<
ToString[j] << " ."]; pAB = wAB; wAB[[i]] = wAB[[i]]/wAB[[i, j]]; For[k = 1, k
≤ m, k ++,
If[k ≠ i, wAB[[k]] = wAB[[k]] - wAB[[i]]wAB[[k, j]]]; For[k = 1, k
≤ m, k ++, wAB[[k, j]] = 0]; wAB[[i, j]
= 1; CellPrint[Cell[BoxData[RowBox[{"(", GBox[pAB, pwBasis], ")"}], "", "(",
GBox[wAB, wBasis], ")"}]], FontWeight → Bold, Background
→ RGBColor[0.8, 1., 0.8],
CellMargins → {{50, Inherited}, {0, 0}}, CellFrame
→ {{3, 0}, {0, 0}}]; If[VerifywBasis & Length[wBasis]!
= rangMatrix[extendedMatrix], PrintStr["Ранг матрицы системы равен" <
> ToString[rangMatrix[A]] << " , ранг расширенной матрицы равен" <
> ToString[rangMatrix[extendedMatrix]] <
> " . Система несовместна!"]; Return[]]; If[! logBasis, YesNo
= "Да/Нет";
While[(YesNo ≠ "Да") && (YesNo ≠ "Нет"), YesNo
= YesNoInput["Есть базис?"]; PrintStr["Есть базис? ..." <>
YesNo <> " ."]; If[VerifywBasis, Which[YesNo =
"Да", PrintStr["Верно! Базис есть."], YesNo =
"Нет", PrintStr["Неверно! Базис есть."]; YesNo = "Да",
Which[YesNo == "Да", PrintStr["Неверно! Базиса нет."]; YesNo
= "Нет", YesNo =
"Нет", PrintStr["Верно! Базиса нет."]]], YesNo
= "Да"]; logBasis = True; BasisNumbers
= wBasis; logStepByStep = True; SolSys[];
logStepByStep = False; If[rangMatrix[A] == n, Return[]]; changeBasis
= "Да/Нет"; While[(changeBasis ≠ "Да") && (changeBasis
≠ "Нет"),
changeBasis = YesNoInput["Изменить базис?"];

```

```
PrintStr["Изменить базис? ..." <> changeBasis <> "."]; If[changeBasis =
= "Да", YesNo = "Нет",
OutPutSys = True]]
```

Для работы с программой создана палитра **PLinSystem2020**, которая содержит восемь команд:

Ввод $m, n \rightarrow$	<i>Inputmn</i> [■]
Ввод матрицы $A \rightarrow$	<i>InputA</i> [■]
Формульный ввод $A \rightarrow$	<i>FinputA</i> [■]
Ввод столбца $B \rightarrow$	<i>InputB</i> [■]
Формульный ввод $B \rightarrow$	<i>FinputB</i> [■]
Решение системы \rightarrow	<i>SolSys</i> [■]
Step by step решение \rightarrow	<i>StepByStep</i> [■]
О программе \rightarrow	<i>About</i> [■]

Первая команда *Inputmn*[■] используется для ввода числа уравнений m и числа неизвестных n системы.

Вторая и третья команды служат для ввода матрицы A коэффициентов системы. По команде *InputA*[■] коэффициенты вводятся поэлементно, причем они могут быть рациональными или действительными числами, а также значениями любых числовых функций программы *Mathematica*. Команда *FinputA*[■] позволяет записать единое аналитическое выражение для коэффициентов в виде:

$$a_{i_j_} := f[i, j];$$

где $f[i, j]$ – любая числовая функция программы *Mathematica*.

Аналогично можно ввести и элементы столбца B правых частей системы – команды *InputB*[■] и *FinputB*[■].

При вводе всех данных следует придерживаться синтаксиса программы *Mathematica* ([1, 2] или Help программы).

Для решения системы в автоматическом режиме используется команда *SolSys*[■]. По этой команде программа выводит жорданову форму расширенной матрицы системы и предлагает ввести список номеров базисных неизвестных, после чего, в случае совместности системы, она выводит ее общее решение и соответствующее базисное решение. Для удобства базисные неизвестные выводятся на красном фоне, их значения – на желтом.

Полезной для обучения методу Жордана-Гаусса является команда пошагового решения системы **StepByStep**[■]. Она в цикле, шаг за шагом, запрашивает номера разрешающей строки и разрешающего столбца, пересчитывает и выводит расширенную матрицу системы до тех пор, пока не будет получен базис или не будет установлено, что система несовместна. В случае совместности выводится общее и соответствующее базисное решение системы. Затем программа предлагает изменить базис и, если предложение принято, то она, после ввода номеров разрешающей строки и разрешающего столбца, пересчитывает базис и для нового базиса выводит жорданову форму расширенной матрицы системы, а также общее и соответствующее базисное решение.

Последняя команда **About**[■] выводит информацию о работе с программой.

Помимо использования этой программы при обучении методу Жордана-Гаусса, ее несложно модифицировать, например, для решения задачи линейной оптимизации симплекс-методом.

Список литературы

1 Wolfram Language&System / Documentation Center [Электронный ресурс]. – Режим доступа : <https://reference.wolfram.com/language/>. – Дата доступа : 19.10.2020.

УДК 378.14:51

О МАТЕМАТИЧЕСКОЙ ПОДГОТОВКЕ СТУДЕНТОВ ТЕХНИЧЕСКИХ СПЕЦИАЛЬНОСТЕЙ БНТУ

Г.И. ЛЕБЕДЕВА

Белорусский национальный технический университет, г. Минск

Высшая математика является одной из важнейших дисциплин, читаемых в техническом университете. Она является базовой для изучения широкого круга дисциплин. Трудно представить механику, сопромат, детали машин и так далее без математики. Будущий инженер не только должен знать высшую математику, но и грамотно применять её аппарат на практике.

В настоящее время получает широкое распространение математическое моделирование различных процессов и объектов. Выпускник вуза не только должен уметь составлять математические модели, но и должен предлагать методы их оптимального решения.