

# ПРЕПОДАВАНИЕ МАТЕМАТИКИ: ПРИМЕНЕНИЕ ТАБЛИЧНЫХ ПРОЦЕССОРОВ ОФИСНЫХ ПАКЕТОВ

Кузов Б.Н.

*Рассматривается преподавание математики с использованием табличного процессора Calc из свободно распространяемого офисного пакета [OpenOffice.org 2.4.0](http://OpenOffice.org). (см. [оценку этого пакета проф. Ильиным В.Д.](#))*

## 1. Введение

Сближение классической и вычислительной математики может быть реализовано на основе современных технологий решения задач с использованием программ массового применения. Конечно, применению компьютерных технологий решения задач всегда должна предшествовать необходимая подготовка в классической математике. Опыт преподавания ряда математических дисциплин (математические методы моделирования, исследование операций (МИРЭА), высшая математика (Институт государственного администрирования)) позволил автору оценить, насколько важен вычислительный аспект математики для будущих инженеров, экономистов и специалистов в области организационного управления. Применение предлагаемого подхода показало, что компьютерные технологии решения задач позволяют больше внимания уделять практическому закреплению основ математических знаний. Таким образом, работа с программами не только формирует практические навыки, необходимые для будущей профессиональной деятельности. Важно, что при этом высвобождается время для углубленного освоения теоретических основ изучаемых методов и расширения круга задач, которые решают студенты. Существенно, что при этом размерность задач и точность получаемых решений максимально приближены к реальным условиям. Кроме программ, работающих в режиме вычислений, целесообразно применять программы, поддерживающие режим символьных вычислений. Работая в этом режиме, студент точнее воспринимает теоретические положения и приобретает дополнительные навыки формализации. Выбор программ во многом зависит от специализации студентов. В любом случае предпочтения заслуживают функционально расширяемые программные системы.

## 2. Предлагаемый подход

При решении задач особое внимание (после анализа постановки и выбора метода) уделяется алгоритмизации процесса решения. Алгоритмизация связана с дискретным представлением данных и учетом зависимости точности результата от точности исходных данных. Решение задач с учётом погрешностей в исходных данных и достижимой точности результатов – обязательная составляющая подготовки будущих инженеров, экономистов и специалистов в области организационного управления. Учёт погрешностей данных, анализ их влияния на точность решений – проблема, актуальность которой имеет непреходящий характер. Двадцать-тридцать лет назад для её решения требовалось прилагать существенные усилия, связанные с разработкой специальных методов и программного обеспечения [1-3]. Современные технологии позволяют намного проще решать подобные проблемы. На первом этапе формирования навыков компьютерного решения задач целесообразно начинать с задач, которые студенты могут решить сначала вручную, а затем – с помощью программы. Только после того, как сформировались достаточно устойчивые навыки, можно переходить к задачам, решение которых

вручную нецелесообразно из-за высокой трудоемкости. Чтобы развить у студента уверенность в правильности полученных результатов, задачи решаются, как правило, разными способами. Подчеркнем, что Calc рассчитан на расширение множества решаемых задач путем создания пользовательских функций и расширений. Вовлечение в этот процесс студентов, освоивших основы языка Basic, повышает степень усвоения математических методов, ориентирует на создание вычислительных процедур, необходимых для решения задач.

Далее рассматриваются технологические приёмы, которые целесообразно использовать в процессе преподавания высшей математики с применением средств Calc. В качестве примеров приведены решения типовых задач математического анализа и матричной алгебры.

## 2.1. ЗАДАЧИ МАТЕМАТИЧЕСКОГО АНАЛИЗА

Основные задачи математического анализа, изучаемые в вузах, включают разделы математики, посвященные исследованиям функций методами дифференциального и интегрального исчисления. При этом рассматриваются числовые последовательности и их пределы, пределы функций, производные и интегралы. При традиционном изложении приемов решения задач из перечисленных разделов главное внимание уделяется математическим преобразованиям, которые позволяют получить требуемый результат. Такой подход, ориентированный на приобретение устойчивых навыков, требует решения вручную большого количества задач, т.е. весьма существенного времени. Совершенно иначе, с затратой минимального времени решаются те же задачи на компьютере. Достаточно освоить несложные стандартные приёмы, и вычисление типовых задач не представит никаких трудностей.

**Вычисление пределов числовых последовательностей и функций.** Начнем с механизма циклических ссылок при отыскании пределов. Циклической ссылкой в Calc называют формулу, которая непосредственно или через другие формулы ссылается на свой адрес. Механизм циклических ссылок позволяет организовать многократно выполняемые действия (вычисления в цикле). При вычислении пределов основное внимание уделяется определению значения  $n$ , для которого предельное значение (в случае существования предела) будет иметь необходимую точность. С учётом того, что алгоритм изменения шага по  $n$  реализуется весьма просто, можно построить различные последовательности, предельные значения которых будут иметь всё увеличивающуюся точность. При этом особого внимания заслуживает выбор шага по  $n$  для знакопеременных последовательностей.

Пусть требуется вычислить предел

$$\lim_{n \rightarrow \infty} \sqrt{n^2 + 3n} - n \quad (1)$$

Для увеличения значения  $n$  при вычислении предела воспользуемся простейшим типом явной циклической ссылки: в ячейку A1 введём формулу: =A1+1. Важно отметить, что при вычислении различных пределов, будут меняться лишь формулы в ячейке A2, написание которых не представляет никаких трудностей (рис. 1):

	A
1	=A1+1
2	=КОРЕНЬ(A1^2+3*A1)-A1

Рис. 1. Формулы для вычисления предела (1).

Перед вычислением предела требуется установить нужные параметры циклических вычислений (число итераций, режим вычислений). При выборе режима вычислений Вручную и выполнении 300 итераций возникнут значения, представленные на рис. 2.

	A
1	300
2	1.496269

Рис. 2. Результат выполнения 300 итераций

Нетрудно увеличить точность предельного значения. После выполнения 1500 итераций значение предела будет равно 1.499251, а выполнив 800000 итераций получим фактически теоретическое значение 1.499999. Дальнейшее увеличение n (при интересующей нас точности) не окажет влияния на предельное значение. Таким образом, выбор максимального числа итераций должен диктоваться величиной изменения значения предела при увеличении числа итераций: если вычисляемое значение при увеличении числа итераций стабилизировалось, значит нет необходимости продолжать вычисления. Покажем теперь, как можно вычислить тот же предел, применяя иной алгоритм – предварительно созданную последовательность значений n (рис. 3).

	A	B	C	D	E	F	G	H
1	300	600	900	1200	1500	1800	2100	2400
2	1.496	1.498	1.499	1.499	1.499	1.499	1.499	1.500

Рис. 3. Величина предела при разных значениях n

Формулы в ячейках B2:H2 получены автозаполнением формулы из ячейки A2 (рис. 4).

	A
1	300
2	=КОРЕНЬ(A1^2+3*A1) - A1

Рис. 4. Формула для вычисления предела

Оба способа быстро привели к одинаковому результату (при равном числе итераций). Чтобы лучше понять исследуемый вычислительный процесс и избежать ошибочных выводов, следует использовать его графическое представление. Эта рекомендация остаётся в силе и для всех иных вычислений, где есть возможность визуализации получаемых результатов. В Calc графики очень просто построить, используя Мастер диаграмм.

В приведённом примере процесс сходимости последовательности (1) к пределу иллюстрируется графиком, представленном на рис. 5.

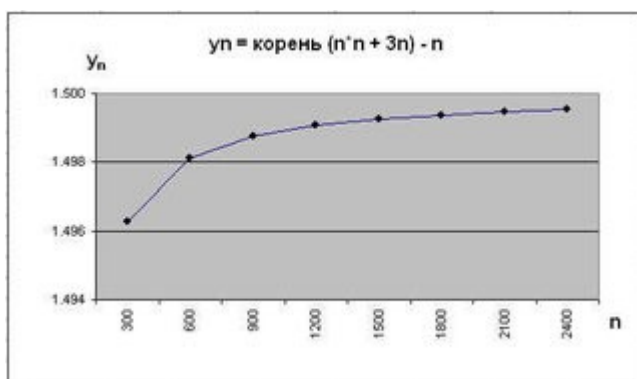


Рис. 5. Визуализация процесса достижения предельного значения

Рассмотрим теперь знакопеременную последовательность чисел. Пусть требуется вычислить предел

$$\lim_{n \rightarrow \infty} \frac{(-1)^n n}{n+1} \quad (2)$$

Если число итераций будет чётным, то мы не почувствуем изменения знака у членов последовательности (2). При правильном задании  $n$  (чередование чётных и нечетных значений) получим график, свидетельствующий об отсутствии предела рассматриваемой последовательности (рис. 6):

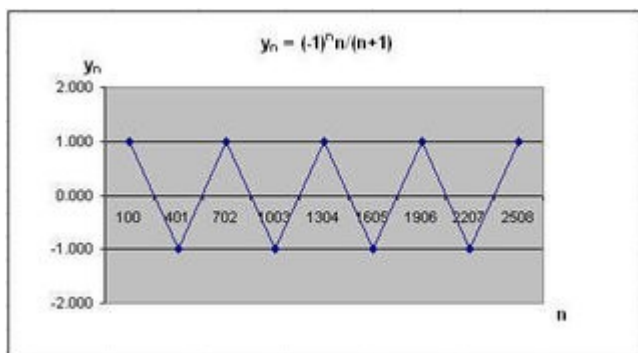


Рис. 6. Визуализация поведения исследуемого предела

Использование механизма циклических ссылок позволяет организовать вычисления более компактно, тогда как при явном задании последовательности значений  $n$  проще получить графическое представление процесса вычислений. В обоих случаях решение задачи осуществляется по стандартной схеме. Освоение этих схем не представляет никаких трудностей. При вычислении пределов функций возможно использование тех же приёмов, которые были рассмотрены при вычислении пределов последовательностей.

**Вычисление значений производных.** Всё, что говорилось в предыдущем разделе об алгоритмизации процесса решения задачи, её дискретного представления и получения результатов требуемой точности справедливо и в этом случае. Численное значение производной функции в заданной точке можно получить, основываясь на определении производной, или с помощью формул численного дифференцирования. Рассмотрим вначале первый из вариантов. Пусть требуется вычислить значение производной функции  $y = (x + 2)^{1/2}$  в точке  $x = 2$ . На рис. 7 приведены соответствующие формулы для реализации циклических вычислений.

А	
1	=A1+1
2	=A8=(-0.5)^A1
3	2
4	=(КОРЕНЬ(A3+A2+2)-КОРЕНЬ(A3+2))/A2

Рис.7. Формулы для вычисления значения производной

Для получения значения производной таблично заданных функций можно воспользоваться различными формулами. Простейшей из них является трёхточечная формула:

$$y'_0 \approx \frac{1}{2\Delta x} (-y_{-1} + y_1),$$

где  $y_{-1}$ ,  $y_0$  и  $y_1$  – три последовательных значений функции, а  $\Delta x$  – шаг дискретизации.

Если используемые данные имеют небольшую точность, следует использовать формулы с большим числом точек. Например, пятиточечная формула имеет вид:

$$y'_n \approx \frac{1}{10\Delta x} (-2y_{n-2} - y_{n-1} + y_{n+1} + 2y_{n+2})$$

Используя интегрированную среду разработки Basic и обладая базовыми знаниями в области программирования, студенты могут решать сформулированные задачи, создавая пользовательские функции. Так как формулы численного дифференцирования весьма чувствительны к ошибкам в исходных данных, поэтому должна оцениваться точность получаемых результатов по известным формулам.

**Вычисление определённого интеграла.** Точность интегрирования будет определяться количеством точек, в которых известно значение функции и выбором конкретного метода. Если функция задана аналитически, то за счёт уменьшения шага дискретизации возможно достижение заданной точности. При этом усложнение метода интегрирования приведет в общем случае к повышению точности результата (при одинаковом количестве точек). Если значения функции получены экспериментально, то повышение точности будет связано лишь с изменением метода интегрирования. В этом случае нетрудно оценить влияние точности исходных данных на значение интеграла. В качестве примера рассмотрим вычисление определённого интеграла

$$S = \int_0^1 (\sqrt{x} - x^2) dx,$$

теоретическое значение которого равно 1/3. При вычислении воспользуемся методом трапеций:

$$\int_a^b f(x) dx \approx \left( \sum_{i=1}^{n-1} f(x_i) + \frac{f(x_0) + f(x_n)}{2} \right) \Delta x$$

Сформируем таблицу исходных данных, состоящую из двух столбцов. В первом должны содержаться значения аргумента в диапазоне от нижнего до верхнего пределов интегрирования, изменяемые с заданным шагом. Во втором – соответствующие значения подынтегральной функции. Для отыскания приближённого значения интеграла в ячейку C2 введём формулу трапеций для конкретных исходных данных (рис. 8).

	A	B	C
1	0	=КОРЕНЬ(A1)-A1*A1	0.01
2	=A1*\$C\$1	=КОРЕНЬ(A2)-A2*A2	=C1*((B1+B101)/2+СУММ(B2:B100))

Рис. 8. Формулы для вычисления значений аргумента, функции и интеграла

Результат расчета по формуле представлен в ячейке C2 на рис. 9

	A	B	C
1	0	0	0.01
2	0.01	0.0999	0.333112947

Рис. 9. Приближённое значение интеграла

Если бы нам не требовалась такая высокая точность, можно было выбрать меньшую дискретность. Например, для значения шага 0.05 интеграл равняется значению 0.330696561, а для шага 0.1 – 0.325509342. Наоборот, при необходимости увеличения точности можно было бы воспользоваться, например, методом Симпсона. Тогда значение интеграла (при шаге 0.01) составило бы 0.333252149. Ясно, что выбор точности вычисления интеграла для практических случаев напрямую зависит от точности исходных данных. Нетрудно оценить эту зависимость, изменяя исходные данные. Подчеркнём, что все эти расчёты не занимают много времени, быстро осваиваются студентами, которые получают устойчивые навыки алгоритмизации задач для достижения требуемой точности результатов. Полезно давать графическое представление отыскиваемого определённого интеграла. С этой целью в рассматриваемом примере для тех же значений аргумента нужно вычислить значения

функций  $y_1 = \sqrt{x}$  и  $y_2 = x^2$  (поместив их в соответствующие столбцы) и воспользоваться Мастером диаграмм для построения этих функций. Вычисляемая площадь заключена между изображенными на рис. 10 графиками функций  $y_1$  и  $y_2$ .

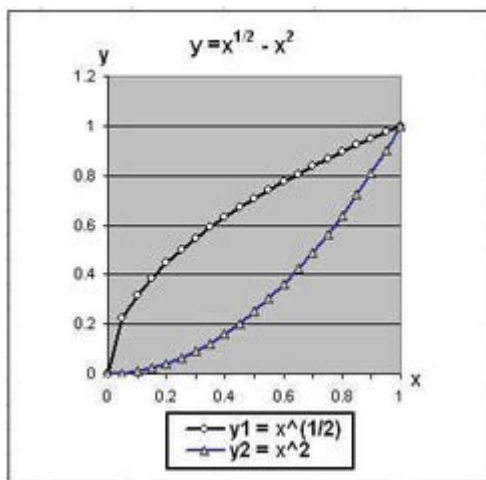


Рис. 10. Вычисляемая площадь (ограничена построенными графиками)

Заметим, что сложность подынтегральной функции практически не влияет на затраты времени при вычислении интеграла, так как соответствующую формулу требуется ввести всего один раз. Практика показывает, что студенты весьма быстро овладевают компьютерными механизмами вычисления определённых интегралов, оставляя больше времени на анализ сути рассматриваемых задач.

## 2.2. ЗАДАЧИ МАТРИЧНОЙ АЛГЕБРЫ

При изучении в вузах матричной алгебры много времени уделяется рассмотрению вспомогательных операций на пути решения одной из основных задач – решение системы линейных алгебраических уравнений (СЛАУ). Например, от студентов требуется освоение таких вспомогательных операций как вычисление определителя и отыскание обратной матрицы. Суть необходимых математических преобразований становится ясной довольно скоро, но их практическое применение при вычислении определителей и получении обратных матриц занимает немало времени и ограничивается задачами небольшой размерности. Ясно, что решение задач реальной размерности вручную весьма трудоёмко и малопродуктивно. Применение компьютерных программ смещает акцент в сторону правильной подготовки данных, умению пользоваться стандартными функциями и анализировать получаемые результаты. Естественно, что решению задач на компьютере должно предшествовать строгое определение всех выполняемых матричных операций с указанием роли каждой из них.

Операции сложения и вычитания матриц выполняются в Calc с помощью обычных арифметических операций над блоками; для умножения матриц, их транспонирования, вычисления определителя, обратной матрицы, решения СЛАУ используются встроенные функции MMULT, TRANSPOSE, MDETERM, MINVERSE. Возможности этих функций и интерфейс аналогичны тем, что применяются в Excel [4]. Наиболее подготовленные студенты могут разрабатывать и интегрировать в OpenOffice.org свои собственные компоненты UNO (Universal Network Objects - универсальные сетевые объекты). Эти новые компоненты, называемые дополнениями, расширяют функциональность OpenOffice.org. Их можно добавлять в меню и на панели инструментов.

### 3. Заключение

Применение современных информационных технологий в процессах преподавания математики способствует углубленному усвоению студентами теоретических основ и формированию навыков решения практических задач. На примере OpenOffice.org 2.4.0 Calc показано, как освоение математических методов можно эффективно сочетать с формированием навыков решения задач с помощью программ массового применения. Студенты овладевают способами алгоритмизации процессов решения задач, связанными с дискретизацией исходных данных, навыками решения задач реальной размерности, подготовки и контроля данных, анализа точности получаемых результатов. Рассмотренные технологические приёмы особенно эффективно способствуют освоению методов решения типовых задач будущими экономистами и специалистами в области организационного управления.

### Литература

1. ИЛЬИН В.Д., КУРОВ Б.Н. *Сравнение алгоритмов распределения нагрузок с учетом изменения состояния энергосистемы при реализации решений* // Электричество. –1972. – № 9. – С.9-14.
2. ИЛЬИН В.Д., КУРОВ Б.Н., ХРУЩЕВ С.Н. *Математическое обеспечение одного класса систем автоматизации научных исследований* // УСиМ. – 1980. – № 4. – С.99–102.
3. БАРЫШНИКОВ В.Н., ИЛЬИН В.Д., КУРОВ Б.Н., СЁМИК В.П. *Обработка результатов эксперимента в системе ДИЭКС.* // УСиМ. – 1984. – № 1. – С.62–65
4. СДВИЖКОВ О.А. *Математика в Excel 2003.* – М.: СОЛОН-ПРЕСС, 2005. 192 С.