

ПРОГРАММИРОВАНИЕ: ПУТЬ К СЕРДЦУ СТУДЕНТА

Гаев Е.А.

Национальный авиационный университет, Киев, Украина

Сегодняшние студенты совсем иначе воспринимают содержание и методы обучения, чем это было у их сегодняшних учителей. И главное отличие состоит в их настрое на компьютеризацию и мультимедиа при слабом владении базисными науками. Это следует учитывать для приспособления образования под их требования. Путь "к сердцу и уму" студента предлагается через программирование задач, подлежащих освоению. Если ранее дисциплина "Вычислительные методы и программирование" излагалась на старших курсах, то теперь предлагаю вводить алгоритмизацию и программирование на младших, чтобы последующие дисциплины могли рассчитывать на создание специализированных приложений в среде, которая позволяет это делать быстро и комфортно. Ряд учителей и преподавателей ВУЗов привержены этой идее. Из многочисленных языков и сред программирования наиболее подходящим, на наш взгляд, является MATLAB. Данный доклад это предложение аргументирует.

MATLAB сегодня известен преимущественно как средство решения широчайшего круга математических задач. Однако его версии после 2007 г. содержат все основные современные инструменты программирования – операторы организации потока данных *if*, *for*, *while*, *switch* и *try*, возможности GUI-интерфейса, концепции типа данных и объектно-ориентированного программирования (ООП). Но поскольку все это "держится" на богатейшей математической и графической библиотеках, то удастся за годичный курс (примерно 30 лекционных и 70 лабораторных часов) довести новичков до сложных и красивых программ общенаучного и технического содержания.

Предлагается ряд упражнений и серьезных MATLAB-программ, которые опираются лишь на школьный и начальный ВУЗовский курс математики и обосновывают этот подход.

1. Одна из первых задач традиционного программирования предлагает вычислить значение функции по ее ряду Тейлора. MATLAB же позволяет это усвоить на основе "численной игры" с графиками последовательных полиномов.

2. Использование цикла *for* при новом подходе целесообразно начать с анимации вращения какой-либо фигуры на экране. Вообще, простой и естественный язык MATLAB позволяет концентрироваться на алгоритмах, а не на "языковой стороне" проблемы.

3. После рассмотрения ряда классических алгоритмов (поиск наибольшего или наименьшего элемента, сортировка числового или текстового массива и т.п.) следует предложить такой, когда алгоритм не укладывается в линейную схему "Программа-Подпрограмма". Например, построение геометрического фрактала требует вычлнить ряд независимых задач и программу-менеджер, которая их циклически вызывает. Аналогично понимаются рекурсивные алгоритмы.

4. Будут показаны некоторые программы, "завернутые" в красивые графические интерфейсы (GUI). Опыт убеждает, сколь сильно они мотивируют студентов.

5. После такого "курса молодого бойца" студенты в состоянии писать любые программы в каждой из изучаемых дисциплин. Как иллюстрация, показано несколько таковых, созданных студентами. 5.1. Две программы из курса физики: (а) полет с планеты на планету, (б) дисперсия луча света, проходящего через треугольную призму. 5.2. Программы тестирования знаний в произвольной области: тестируемому учащемуся задаются те или иные вопросы в случайном порядке с несколькими вариантами ответов; если дисциплина из области точных наук, то имеется возможность случайного (и более неповторимого) генерирования, например, коэффициентов задачи.

Последний п. 5 и стимулирует новый стиль работы студента. MATLAB идеально подходит для предлагаемого методического подхода.