РОССИЙСКАЯ АКАДЕМИЯ ОБРАЗОВАНИЯ

Учреждение Российской
академии образования
«Институт педагогических
исследований одаренности детей»

ПЕДАГОГИЧЕСКИЕ ЗАМЕТКИ

Научный журнал

Основан в октябре 2008 года

Том 4

Выпуск 1

2011

УЧРЕЖДЕНИЕ РОССИЙСКОЙ АКАДЕМИИ ОБРАЗОВАНИЯ
ИНСТИТУТ ПЕДАГОГИЧЕСКИХ ИССЛЕДОВАНИЙ
ОДАРЕННОСТИ ДЕТЕЙ

ПЕДАГОГИЧЕСКИЕ ЗАМЕТКИ
Научный журнал. 2011. Т. 4, вып. 1

Свидетельство о регистрации средства массовой
информации ПИ № ФС77-36663 от 01 июля 2009г.

РЕДАКЦИОННАЯ КОЛЛЕГИЯ

Главный редактор
академик РАО *А. А. Никитин*

Заместители главного редактора
к.ф.-м.н. *А. С. Марковичев*
к.э.н. *О. А. Никитина*

Ответственный секретарь
к.п.н. *Ю. В. Михеев*

Члены редколлегии:
академик РАО *Ю. В. Сенько*
чл.-корр. РАО *И. М. Бобко*
чл.-корр. РАО *А. Ж. Жафяров*
чл.-корр. РАО *В. Я. Синенко*
к.п.н. *Г. А. Сапрыкина*

Оригинал-макет
Л. А. Дегтерева, Е. Н. Разинков

Адрес редколлегии:
630098, г. Новосибирск,
ул. Приморская, д. 22
Телефон: (383) 345-80-21
E-mail: edusoft@ngs.ru

Подписано в печать 18.03.2011.
Бумага офсетная №1. Формат 30 х 42/2.
Гарнитура Times New Roman.
Печать офсетная. Усл. печ. л. 51.
Тираж 500 экз. Заказ № 04-11.

Издательство ИПИО РАО
г. Новосибирск, ул. Приморская, д.22

© ИПИО РАО, 2011

Социализация детей и молодежи в мировом научном пространстве

УДК 376.545:371.385.5

Михеев Юрий Викторович, Никитин Александр Александрович
Учреждение Российской академии образования
«Институт педагогических исследований одаренности детей»
Россия, г. Новосибирск, ул. Приморская, д. 22, телефон: (383) 345-80-21
edusoft@ngs.ru

Алешин Владислав Дмитриевич, Пащенко Михаил Георгиевич
Специализированный учебно-научный центр
Новосибирского государственного университета
Россия, г. Новосибирск, ул. Ляпунова, д. 3 (ул. Пирогова, д. 11), телефон (383) 330-30-11

В статье рассматриваются некоторые направления ранней подготовки молодежи к научной деятельности с учетом опыта работы физико-математической школы при НГУ и СУНЦ НГУ. Приведен ряд задач по математике, которые можно использовать в общеобразовательной школе на специализированном уровне обучения.

Ключевые слова: специализация, математика, образование, специализированное обучение, исследовательская деятельность, индивидуализация обучения, одаренные дети.

В последнее время все сильнее осознается важность проблемы социализации молодежи. Поэтому неслучайно в средней общеобразовательной школе в последнее десятилетие все шире внедряется профильный подход к обучению, что позволяет решать проблему социализации молодежи с учетом профессиональных интересов. При этом одним важных направлений в профессиональной ориентации молодежи является подготовка определенной части учащихся общеобразовательных учреждений к исследовательской деятельности в области науки или техники.

Важность ранней подготовки молодежи к научной деятельности ведущие ученые осознавали во все времена. Интеллект является достоянием нации. Одаренные дети есть во всех социальных слоях населения и во всех регионах страны. Раскрыть таланты, помочь им раскрыться и подготовить их к овладению вершинами профессиональной деятельности – это стратегическая задача для страны. Поэтому неслучайно с середины 30-х годов XX века в нашей стране началось регулярное проведение школьных олимпиад по математике, а с середины 60-х годов и по другим естественнонаучным дисциплинам.

Более 50 лет назад, когда создавался Новосибирский научный центр, широко известный как Академгородок Новосибирска, в качестве одной из главных была поставлена задача подготовки научных кадров. В результате с 1962 г. была реализована триединая система: «Школа – Университет – Научно-исследовательский институт», в которой начальным звеном стала всемирно известная Новосибирская физико-математическая школа, преобразованная в 1989 г. в Специализированный учебно-научный центр Новосибирского государственного университета (СУНЦ НГУ).

Похожие физико-математические школы и СУНЦ были созданы в Москве, Киеве, Санкт-Петербурге и Екатеринбурге.

© В. Д. Алешин, Ю. В. Михеев, А. А. Никитин, М. Г. Пащенко, 2011

Может показаться, что создание специализированных физико-математических школ за счет отбора из других школ наиболее сильных учеников отрицательно скажется на уровне школьного образования в целом. Однако практика показывает, что это не совсем так. Возможности даже лучших средних общеобразовательных учреждений по обучению одаренных в области математики и естественнонаучных дисциплин значительно уступают возможностям, в том числе и кадровым, которые может предоставить крупный научный центр или ведущий университет. И в этом плане одаренные дети, поступая в СУНЦ, получают большие возможности для реализации своего потенциала.

Под влиянием этого процесса были созданы также и другие школы с углубленным изучением математики и естественнонаучных дисциплин. В частности, при методической поддержке СУНЦ НГУ созданы и функционируют физико-математические школы в г. Омске, г. Петропавловске-Камчатском, и т. д. Важной особенностью преподавания в СУНЦ НГУ является то, что одна из главных целей обучения – это развитие исследовательских навыков. В результате повышается доля самостоятельности учащихся в учебной деятельности, что достигается за счет индивидуализации обучения. Наиболее глубокой формой реализации принципа индивидуализации обучения является организация работы одного или небольшой группы учащихся под научным руководством преподавателя или научного сотрудника одного из научно-исследовательских институтов Академгородка. При такой форме организуется некий коллектив, в котором основной ролью руководителя является поиск направлений исследования, необходимой учебной литературы, анализ текущих результатов, получаемых учащимися, корректировка формулировок и постановок вновь возникающих задач. Учащиеся при такой форме деятельности самостоятельно знакомятся с необходимой литературой, предпринимают определенные начальные шаги, обсуждают текущие достижения с руководителем, и в случае необходимости получают от руководителя рекомендации по продолжению направлений исследования.

Понятно, что школьникам сложно решать передовые проблемы науки, если это и удается, то в исключительных случаях. Однако решать доступные учащимся школы, интересные и содержательные проблемы вполне возможно. Материал для такой деятельности можно черпать из литературных источников, получать как вспомогательные направления серьезных научных исследований по основной деятельности руководителя, а также как исследование некоторых новых, иногда неожиданных подходов к достаточно известным вещам.

Практика индивидуальной работы с учащимися и их индивидуальной подготовки по тому или иному предмету широко распространена в СУНЦ НГУ. Итогом этой деятельности, как правило, является резкое повышение интереса детей к соответствующей сфере знаний и часто становится определяющим фактором при выборе своей будущей профессии.

Поясним подробнее принципы организации индивидуальной подготовки учащихся по математике на нескольких примерах.

Идея одного исследования возникла в процессе общения одного из преподавателей с учащимися своей группы из СУНЦ НГУ по поводу комплексных и гиперкомплексных чисел. Дело в том, что при первом же знакомстве, например, с комплексными числами часто возникают ощущения их неприятия. Привычка к натуральным, целым, рациональным и действительным числам вырабатывается на протяжении многих детских и школьных лет. В отличие от этого комплексные числа чаще всего вводятся сразу, либо как упорядоченные пары (a, b) действительных чисел, либо как записи вида $a + bi$, где i – особый символ, удовлетворяющий равенству $i^2 = -1$, либо как квадратные матрицы второго порядка особого вида $\begin{pmatrix} a & b \\ -b & a \end{pmatrix}$, где a, b – действительные числа. При таком подходе практическую значимость комплексных чисел на этом начальном этапе вообще трудно пояснять. Поэтому сразу и возникает недопонимание. В частности, при изучении данного материала возник вопрос – возможны ли другие формы представления комплексных чисел, в том числе и с помощью матриц второго порядка. Данная проблема показалась интересной учащимся, и один из них прямо у доски выяснил, что если за единицу принимать матрицу $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, то существует двухпараметрическое семейство матриц, дающих в квадрате элемент (–1). Например, если считать, что матрица $\begin{pmatrix} 3 & 5 \\ -2 & -5 \end{pmatrix}$ соответствует мнимой единице i, то множество матриц вида $\begin{pmatrix} x+3y & 5y \\ -2y & x-3y \end{pmatrix}$, где x, y – действительные числа, с установленными для матриц операциями сложения и умножения обладает всеми свойствами множества комплексных чисел.

Далее учащиеся по совету своего преподавателя продолжили исследование по представлению разными способами кватернионов в виде матриц. Для этого им пришлось ознакомиться с кватернионами по литературе, указанной руководителем, после чего вполне самостоятельно проводили исследование в двух направлениях.

Вариант 1. Представление кватернионов в виде матриц второго порядка с элементами из множества комплексных чисел.

Вариант 2. Представление кватернионов в виде матриц четвертого порядка с элементами из множества действительных чисел.

На каждом из этих направлений были получены новые интересные результаты, с которыми два автора успешно выступили на Международной студенческой научной конференции НГУ (школьная секция).

Идея другого исследования возникла на основе обобщения следующей задачи из задачника журнала «Квант»: «Имеется лента из клеток размером 1 на m. На ней одна клетка закрашена, на другой поставлена фишка. Бросается игральная кость, фишка перемещается к закрашенной клетке на число клеток, выпавшее на игральной кости. Если фишка не достигла закрашенной клетки, но не проскочила её, то игральная кость бросается снова. Процесс прекращается, когда фишка попала в закрашенную клетку или проскочила её. При каком начальном положении фишки вероятность попадания в закрашенную клетку будет максимальной?».

Лектор одного из лекционных потоков СУНЦ НГУ сообщил об этой задаче учащимся, указал, с какой математической литературой полезно познакомиться, и сказал, что в тех случаях, когда некоторый процесс представляется в виде бесконечной числовой последовательности, то интересно ставить вопрос о сходимости и вычислении предела.

Данная задача заинтересовала одного ученика, и он за некоторое время успешно справился со всеми проблемами, которые возникали по пути к окончательной цели. В итоге удалось показать, что при неограниченном росте числа клеток вероятность попадания в закрашенную клетку стремится к 2/7. С этим результатом автор также успешно выступил на Международной студенческой научной конференции.

На двухгодичных потоках СУНЦ НГУ, на которых в течение первого года обучения уже приобретаются серьезные знания в области математического анализа, можно предлагать учащимся достаточно сложные проблемы. Ограничимся двумя интересными задачами, с которыми некоторым из учащихся удавалось справляться.

В первой задаче предлагалось рассмотреть один «экзотический» числовой ряд

$$\frac{1}{3}+\frac{1}{7}+\frac{1}{8}+\frac{1}{15}+\frac{1}{24}+\frac{1}{26}+\frac{1}{31}+\frac{1}{35}+\frac{1}{48}+\ldots,$$

в котором без повторений рассматривается сумма всех слагаемых вида $\frac{1}{n-1}$, где n является некоторой точной степенью (квадратом, кубом, четвертой, пятой степенью и т. д.) натурального числа, большего единицы, и установить, что сумма ряда равна 1.

Во второй задаче для каждого действительного x рассматривается последовательность $a_n(x) = a_n$, которая задается рекуррентно: $a_1 = \sin x$, $a_{n+1} = \sin a_n$ при каждом $n \geq 1$, а затем последовательность $b_n = \sqrt{n}\cdot a_n$. После этого ставится вопрос, чему равен предел последовательности b_n при $n \to \infty$. Интересно отметить, что при каждом $0 < x < \pi$ этот предел равен $\sqrt{3}$.

Приведенные примеры исследований старшеклассников в области математики демонстрируют, что при целеустремленном отношении к работе учащимся вполне по силам достаточно серьезные математические результаты.

Конструкции дискретных математических объектов в рамках парадигмы функционально-декларативного программирования

УДК 372.851:004.43

Власов Владимир Николаевич
Специализированный учебно-научный центр
Новосибирского государственного университета

Россия, г. Новосибирск, ул. Ляпунова, д. 3 (ул. Пирогова, д. 11), телефон (383) 330-30-11

vlasov@lab.nsu.ru

В статье рассмотрен опыт преподавания спецкурса «Введение в функциональное программирование» кафедры информатики и дискретной математики СУНЦ НГУ в рамках изучения языка программирования общего назначения Haskell с учетом концепций современной дискретной математики: теории моделей, теории категорий, теории алгоритмов и рекурсивных функций, математической логики и др. Целью спецкурса является ознакомление учащихся специализированных школ, физико-математических классов средних школ, одаренных школьников и студентов младших курсов университетов с разнообразными математическими идеями в процессе конструирования алгебраических и других дискретных объектов самими обучающимися.

Ключевые слова: обучение, математика, информатика, функциональное программирование, Haskell, алгебраические системы и модели, категории, вычислимость, математическая логика, теория множеств.

Введение

Изучение математики и информатики в школе построено на нескольких установившихся в обществе концептуальных парадигмах, определяемых целями такого обучения и спектром социально-культурных традиций, всестороннее обсуждение которых выходит за рамки предлагаемой статьи. Однако отметим, что курс математики в средней школе, к сожалению, далеко не всегда предполагает использование приобретенных знаний и навыков обучаемым при дальнейшем образовании в вузе, а вузовская программа обучения по математике вносит огромное количество абстрактных понятий, о существовании которых вчерашний школьник даже не предполагал – так возникает разрыв процесса обучения при переходе от школьной к вузовской математике. К сожалению, такой разрыв почти неизбежен, так как трудности усвоения школьником идей современной высшей математики многократно усиливаются без соответствующего предварительного ознакомления, например, с алгебраическими объектами и без опыта обращения с ними. Для школьника важен опыт манипуляции с объектом, прежде чем он будет понимать описывающие его теоремы и решать задачи. Во многом именно поэтому школьная математика в старших классах сконцентрирована на геометрии, арифметике и элементарной алгебре и основах математического анализа – разделах, где объекты понятны учащимся уже со средних классов.

С другой стороны, парадигмы изучения информатики в школе, обусловленные необходимостью «компьютерного всеобуча», запросами IT-индустрии, масс-медиа, собственной логикой Computer Science и т. п., далеко не всегда связаны взаимной поддержкой с изучением математики в старших классах. В лучшем случае школьники понимают задачи программирования как последний этап математического исследования и моделирования. Они обычно учатся программировать уже *решенную* математическую проблему (например, вычисление площади подграфика функции или вычисление корней квадратного уравнения), фокусируясь на технологических аспектах.

Однако, уроки информатики могут дать для обучения основам современной математи-

© В. Н. Власов, 2011

ки гораздо больше. Разумеется, в последние годы очень много авторских коллективов направляют усилия на использование IT-технологий в образовании вообще, а на официальном сайте Министерства образования РФ в портале цифровых образовательных ресурсов можно найти несколько тысяч разнообразных программных продуктов, электронных учебников, презентаций, интерактивных интернет сайтов для использования в математическом образовании.

Среди них особо интересен проект «Электронный учебно-методический комплекс (ЭУМК)» в двух вариантах: «Стереометрия» и «ИнДи» [1; 2]. Отметим ряд интересных идей, выделяющий этот проект:

– в комплексе «Стереометрия» обучаемый имеет возможность наглядной манипуляции стереометрическими объектами, что дополняет пространственное воображение и позволяет с помощью прилагаемого инструментария видеть отдельные элементы геометрических конструкций, сечения, проекции;

– есть возможность проведения дополнительных построений и исследований, не включенных явно в логику решения учителем, но проводимых школьником самостоятельно, и есть возможность автоматизированной проверки таких построений;

– в комплексе «ИнДи» реализована свобода произвольных алгебраических преобразований, при этом также есть возможность автоматизированной проверки эквивалентности результатов таких действий.

В данном комплексе заложена идея свободы манипулирования математическими объектами для лучшего понимания задачи и ее решения. Однако, в настоящем варианте указанный комплекс предлагает обучение и манипуляцию с уже знакомыми школьнику объектами: пространственными телами, планиметрическими фигурами, элементарными функциями, алгебраическими преобразованиями, вычислением производных, особо фокусируясь на автоматизации сопровождения экспериментов школьников и процесса решения. Любое включение новых математических объектов, хотя и возможно, но затруднительно и, по-видимому, не являлось целью разработчиков данного программно-методического комплекса.

Но для цели сближения школьной и вузовской математики, а также для цели более тесной интеграции уроков математики и программирования, необходимы идеи и инструменты, достаточно мощные и гибкие для того, чтобы можно было включать многочисленные современные логические, алгоритмические, алгебраические, теоретико-множественные и теоретико-категориальные конструкции.

Предлагается новая идея, суть которой заключается не в проектировании автоматизированного программного комплекса, а в изучении современного языка программирования, который предоставляет программисту-обучаемому возможность создавать известные математические объекты как структурированные данные, функции и иные программные сущности, построенные на базе уже существующих модулей, или создать их заново. Все это возможно реализовать исключительно в рамках синтаксиса и семантики одного языка программирования общего назначения Haskell, причем, силами либо самих школьников или студентов, либо совместно с преподавателем в процессе урока. Таким образом, создавая, например, алгебраический тип данных, изоморфный кольцу вычетов или полю комплексных чисел, школьники легко осваивают сложнейшие математические абстракции, приобретая опыт манипуляции такими данными, и в то же время изучая один из самых современных, интересных и перспективных языков программирования.

Цель данной статьи состоит в том, чтобы познакомить преподавателей математики и информатики, а также студентов и школьников с тем, что может дать современный язык функционального программирования общего назначения Haskell для изучения, приобретения нового опыта и даже для более глубокого освоения некоторых концепций высшей математики. Перечислим, какие идеи могут быть проиллюстрированы и усвоены:

– теория множеств (в том числе и построение бесконечных множеств);

– отношения (в том числе эквивалентность и конгруэнтность), порядки, порядковые типы и операции на них;

– разнообразные алгебраические системы и модели (группы и кольца, поле рациональных и комплексных чисел, гауссовы целые комплексные числа);

– выполнимость на алгебраических системах, конструктивные алгебраические системы;

– рекурсивные структуры (например, деревья и списки) и рекурсивные функции, лямбда-исчисление Черча и другие формализации вычислимости;

– теория категорий, понятие функторов, естественных преобразований и монад;

– разнообразные подходы в изучении математической логики (булевы операции, вывод формул, комбинаторная логика);

– КС-грамматики и грамматический вывод, конечные автоматы и регулярные выражения;

– и многое, многое другое.

Как происходит знакомство школьников с такими концепциями? На спецкурсе «Введение в функциональное программирование» учащиеся параллельно с изучением языка сами создают образцы указанных выше структур, обучаясь и программированию высокого уровня, и основам современной высшей математики. С каждым шагом обучения дается информация о новой математической идее или алгебраической структуре, подходящей для изученного раздела языка, на сайте спецкурса[1] дается подробный пример одной из возможных реализаций аналогичной структуры. Далее в статье будет подробно рассмотрено, как это делается на занятиях в некоторых конкретных примерах.

Стоит отметить, что помимо уже рассмотренного выше ЭУМК и ссылки на многочисленные менее развитые программные продукты по поддержке математического образования в школе, существуют отдельные профессиональные пакеты, направленные на поддержку труда математиков разных специальностей: «Mapple», «Mathematica», «MathLab» – они имеют развитые вычислительные средства, возможности для символических преобразований, но, как правило, не предназначены для заявленных в данной работе целей и не обладают достаточной гибкостью для конструирования разнообразных структур. Однако, существуют и программные системы, направленные на решение задач, аналогичных обсуждаемым в этой работе. Например, свободно распространяемая система GAP, предназначенная для применения в области вычислительной дискретной математики, в частности, теории групп, колец, теории чисел и т. п. Здесь есть свой довольно развитый паскалеподобный специализированный язык программирования, есть возможность работать в интерактивном режиме, есть громадная библиотека различных функций и библиотека данных по группам, что дает действительно эффективные средства для выдвижения и тестирования научных гипотез и для обучения сту-

1 http://wiki.nsunc.com/haskell

дентов алгебре. Однако, предлагаемый в данной работе язык программирования Haskell в отличие от системы GAP является, во-первых, языком общего назначения, что дает, например, возможность объекты построенные в заготовленных примерах, или построенные самостоятельно школьниками, использовать в любых других программах. Во-вторых, Haskell сам в плане языкового дизайна является экспериментальной площадкой математиков и насыщен идеями современной математики. Так, например, система ввода-вывода организована с использованием терминологии и концепции монад из теории категорий, а дизайн анонимных функций построен на базе лямбда-исчисления Черча. Изучение языка Haskell само по себе является введением в современную математику.

1. Развитие функционального программирования и возможности языка Haskell

История функциональных языков начинается с пятидесятых годов прошлого столетия, когда Дж. Мак-Карти создал Lisp, но восходит к идеям А. Черча, Х. Карри и других математиков, известным с конца 20-х годов XX века как лямбда-исчисление, комбинаторная логика, теория рекурсивных функций и т. п.

В функциональном программировании функции рассматриваются как наиболее общие основные понятия. В традиционном, обычно называемом императивным, программировании понятие функции или процедуры несколько отличается от математического, так как на практике императивные функции могут зависеть не только от аргументов, но и от состояния внешних по отношению к функции переменных, и таким образом, при вызове на разных этапах выполнения алгоритма, можно получить разные данные на выходе. Кроме того, императивные функции могут иметь побочные эффекты и могут изменять состояние внешних переменных. Стратегия при императивном программировании заключается в описании последовательности состояний вычислительного процесса для достижения конечного результата. Стратегия при функциональном программировании заключается в декларативном описании функций, фокусируя внимание программиста не на том, как надо делать, а на том, что надо делать. Достоинствами языков функционального программирования является также относительная краткость, выразительность, надежность, модульность, возможность к параллелизации. Программы на функциональном языке хорошо пригодны для формального анализа и доказательства свойств математических объектов [3. С. 5-10].

Языки программирования функционального типа развивались достаточно интенсивно, был накоплен большой опыт в их дизайне и практическом применении, и когда к концу 80-х гг. прошлого века появилось большое и разнообразное множество таких языков, выявилась нужда в некоторой стандартизации и объединении усилий разработчиков в рамках единой платформы. Такая нужда была осмыслена в 1987 г. на конференции по функциональным языкам программирования и компьютерной архитектуре в Орегоне (FPCA'87), где было решено создать комитет для разработки открытого стандарта, и в результате работы которого в 1999 г. был опубликован «The Haskell 98 Report» [4], который стал основой дальнейшего развития языка и его стандартным описанием. Новый язык получил название Haskell, которое произошло от имени выдающегося математика Хаскелла Карри (Haskell Curry).

Язык создан группой математиков и специалистов-теоретиков в области Computer Science. Собственно этот факт и определяет его специфику: чрезвычайную насыщенность языка современными математическими концепциями. Язык имеет необычный синтаксис, его парадигма далека от привычных представлений большинства программистов относительно языков Си и Java. Если писать на Haskell небольшие программы относительно просто, то дальнейшее изучение и тонкое владение неизбежно предполагает математическое образование. Тем не менее, этот язык, как и парадигма функционального программирования в целом, становится все более популярным в IT-сообществе, идеи функционального программирования проникают во многие языки.

Ключевыми особенностями языка Haskell являются:

– использование функций как основных программных сущностей;

– недопустимость побочных эффектов (чистота языка), для ввода-вывода используется концепция монад;

– наличие функций высших порядков (возможность передавать функции в качестве аргументов другим функциям);

– частичное применение функций и каррирование;

– ленивые (отложенные) вычисления;

– параметрический полиморфизм;

– статическая типизация и автоматическое выведение типов (из теоремы Хиндли-Милнера);

– определяемые пользователем алгебраические типы данных, в том числе рекурсивные и параметризуемые типы данных, возможность создания абстрактных типов данных (инкапсуляция);

– разнообразный набор встроенных примитивных типов данных, включая списки, массивы, целые числа произвольной и фиксированной точности и числа с плавающей точкой.

В упомянутом выше «The Haskell 98 Report» [4. С. 3] приведена следующая общая характеристика данного языка: «Haskell является чисто функциональным языком программирования общего назначения, который включает много последних инноваций в разработке языков программирования... Haskell является и кульминацией, и кристаллизацией многих лет исследования нестрогих функциональных языков».

В последующих главах рассмотрены примеры. Все приведенные определения и вычисления тестировались в среде интерпретатора GHCI, входящего в состав стандартного компилятора «The Glasgow Haskell Compiler».

2. Пошаговый пример конструирования алгебраических систем

На подготовительной лекции дается ряд необходимых общематематических понятий. Перечислим их ниже.

Зададим тройку $\langle \sigma_F, \sigma_P, \sigma_C \rangle$, состоящую из попарно непересекающихся множеств, где множество функциональных символов σ_F определяет имена основных операций, множество предикатных символов σ_P определяет множество основных отношений, множество σ_C – множество выделенных элементов. Зададим отображение $\rho{:}\sigma_F \cup \sigma_P \to \mathbf{N}^+$, определяющее арность или местность функциональных и предикатных символов. Тогда $\sigma = \langle \langle \sigma_F, \sigma_P, \sigma_C \rangle, \rho \rangle$ называется *сигнатурой*. Если множества символов сигнатуры σ конечные, то часто будем писать

$$\sigma \quad = \quad \left\langle \left\langle F_0^{n_0}, \ldots, F_k^{n_k}; P_0^{m_0}, \ldots, P_r^{m_r}; c_0, \ldots, c_s \right\rangle, \rho \right\rangle,$$

где $\sigma_F = \{F_0^{n_0}, \ldots, F_k^{n_k}\}$, $\sigma_P = \{P_0^{m_0}, \ldots, P_r^{m_r}\}$, $\sigma_C = \{c_0, \ldots, c_s\}$.

Определение 1 [5. С. 2]. *Алгебраической системой* (моделью) $\mathbf{A}$ сигнатуры σ называется пара $\mathbf{A} = \langle A, \mathrm{int}_\sigma \rangle$, состоящая из непустого множества A и интерпретации int_σ, которая функциональному символу $f \in \sigma_F$ сопоставляет отображение $\mathrm{int}_\sigma(f)$, действующее из $A^{\rho(f)}$ в A, предикатному символу $P \in \sigma_P$ – подмножество $\mathrm{int}_\sigma(P)$ множества $A^{\rho(P)}$, константному символу $c \in \sigma_P$ – элемент $\mathrm{int}_\sigma(c)$ множества A.

Множество A называется *основным множеством* или *носителем* системы, и обозначается как $|\mathbf{A}|$. Отображение int_σ, называется *интерпретацией* алгебраической системы $\mathbf{A}$. Если из контекста ясно, как определены int_σ и ρ и как заданы сами операции и отношения на $\mathbf{A}$, то эти обозначения опускаем, и чтобы задать систему $\mathbf{A}$ в случае конечной сигнатуры, обычно будем писать:

$$\mathbf{A} \;=\; \left\langle A; F_0^{n_0}, \ldots, F_k^{n_k}; P_0^{m_0}, \ldots, P_r^{m_r}; c_0, \ldots, c_s \right\rangle,$$

или сокращенно $\mathbf{A} = \langle A; \sigma \rangle$.

Введенное определение алгебраической системы чрезвычайно важно как для математики, прежде всего для дискретных ее областей, так и для концепций Computer Science, где оно находит много общего с понятием объектов в объектно-ориентированном программировании или с полиморфными перегружаемыми классами в языках, подобных Haskell.

Необходимые понятия термов, логических формул, значений термов и истинности (выполнимости) утверждений на модели для школьников лучше вводить, пользуясь неформальным языком и на конкретных примерах. Для алгебраических структур, которые будут программно определены далее, важно, чтобы обучающиеся умели проверять истинность различных подходящих утверждений.

Определение 2. Алгебраическая система $\mathbf{A} = \langle A, +, \cdot, 0 \rangle$ называется *кольцом*, если на данной системе выполнены следующие аксиомы:

1). $\forall x \forall y \forall z \quad ((x+y)+z = x+(y+z))$;

2). $\forall x \forall y \quad (x+y = y+x)$;

3). $\forall x \quad (0+x = x+0 = x)$;

4). $\forall x \exists y \quad (x+y = y+x = 0)$;

5). $\forall x \forall y \forall z \quad (x\cdot(y+z) = x\cdot y + x\cdot z)$;

6). $\forall x \forall y \forall z \quad ((x+y)\cdot z = x\cdot y + x\cdot z)$.

Если на кольце задано отношение эквивалентности $\sim$, то говорят, что оно устойчиво относительно операций, если выполняются следующие условия:

$$\forall x_1 \forall x_2 \forall y_1 \forall y_2 \quad ((x_1 \sim y_1 \wedge x_2 \sim y_2) \to (x_1 + x_2 \sim y_1 + y_2) \wedge (x_1 \cdot x_2 \sim y_1 \cdot y_2)).$$

В этом случае говорят об отношении конгруэнтности, при котором возможно задать фактор-кольцо. Для целых чисел Z таким фактор-кольцом будет кольцо вычетов по некоторому модулю $Z/\equiv_m$. С другой стороны, можно задать изоморфную алгебру вычетов **K** для множества некоторых объектов, например $\{\overline{0}, \overline{1}, \overline{2}, \overline{3}, \overline{4}\}$, явным указанием таблицы операций (т. е. интерпретацией сигнатурных символов на данном множестве):

+	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

·	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

Эта алгебра будет связана с фактор-кольцом теоремой о гомоморфизме, где эпиморфизм из целых чисел Z задается естественным образом по остатку от деления на m.

Вводная лекция помогает не только дать необходимые определения и теоремы, но выявить общую идею алгебраических систем, а также на примере кольца вычетов понять, откуда и для чего возникают подобные понятия, и как они в данном случае связаны с идеей объединения «схожих» элементов в класс эквивалентности и конструкцией фактор-алгебры.

Далее, после лекции вводится понятие типа данных в Haskell в различных существующих формах, таких как простой тип, алгебраический тип, полиморфный тип [6]. Объясняется, какие синтаксические особенности существуют в данном языке по отношению к другим языкам. На предварительных примерах происходит обучение, после чего вместе с преподавателем по шагам описывается новый тип данных, собственно представляющий кольцо вычетов по модулю 5. В данном примере это может быть представлено следующим образом.

```
data MyAlg = O | I | II | III | IV  deriving (Eq, Read, Show)

-- вспомогательные функции
next :: MyAlg ->  MyAlg
next O   = I
next I   = II
next II  = III
next III = IV
next IV  = O

prev :: MyAlg ->  MyAlg
prev O   = IV
prev I   = O
prev II  = I
prev III = II
prev IV  = III

-- операция сложения, задаем рекурсивно
(<>) ::  MyAlg -> MyAlg -> MyAlg
O <> x = x
n <>x = next  (prev n <> x)
-- операция умножения, задаем рекурсивно
(><) ::  MyAlg -> MyAlg -> MyAlg
O >< _ = O
n >< x = (prev n >< x) <> x

-- эпиморфизм из целых в вычеты (классы экв.)
toMyAlg :: Integer -> MyAlg
toMyAlg 0 = O
toMyAlg 1 = I
toMyAlg 2 = II
```

```
toMyAlg 3 = III
toMyAlg 4 = IV
toMyAlg x = toMyAlg (x `mod` 5)

-- определяем канонического представителя
fromMyAlg :: MyAlg -> Integer
fromMyAlg O   = 0
fromMyAlg I   = 1
fromMyAlg II  = 2
fromMyAlg III = 3
fromMyAlg IV  = 4

-- интерпретация сигнатурных символов (в терминах Haskell: перегрузка
-- операций класса Num
instance Num MyAlg where
  fromInteger = toMyAlg
  x + y =  x <> y
  x * y =  x >< y
-- вводим отрицание (рекурсивно) и др.
  negate O = O
  negate n = prev (negate (prev n))
  signum O = O
  signum _ = I
  abs = id
```

Таким образом, создав полноценный алгебраический тип `MyAlg`, мы фактически построили алгебраическую систему $\mathbf{K} = \langle\{\mathtt{O},\mathtt{I},\mathtt{II},\mathtt{III},\mathtt{IV}\};+,*\rangle$, а функции + и * из сигнатуры интерпретировали через рекурсивно заданные на данном конечном множестве функции <> и ><. Функция `fromInteger` (так же как и `toMyAlg`) в данном контексте осуществляет эпиморфизм из множества целых чисел на множество $\{\mathtt{O},\mathtt{I},\mathtt{II},\mathtt{III},\mathtt{IV}\}$, изоморфное фактор-кольцу вычетов по модулю 5. Функция `fromMyAlg` определяет канонического представителя для каждого класса эквивалентности. Операция нахождения противоположного элемента задана как отрицание `negate` тоже рекурсивно.

После этого, учащиеся могут выполнить разнообразные простые вычисления с данным типом в режиме интерпретатора или даже использовать его для написания дальнейших программ, например:

```
II + II = IV;
II * II = IV;
II + III = O;
II * III = I.
```

Затем непосредственными вычислениями на конкретных примерах проверяются свойства кольца, например, дистрибутивность (вводим и проверяем в среде интерпретатора ghci):

```
 >III * (II + I) == IV
True
>(III * II) + (III * I) == IV
True
```

Здесь же на уроке рассказывается о том, как можно реализовать полное математическое доказательство данного и других свойств.

После того как пример полностью разобран, учащимся дается задание создать собственные примеры алгебраических типов данных, которые будут (в некотором приближении) воплощениями таких алгебраических систем, как, например, булевы алгебры, комплексные числа или векторные пространства.

Интересным развитием идеи приведенного примера является возможность конструктивизации данной алгебраической системы.

Определение 3 [5. С. 174]. Пусть задана алгебраическая система $\mathbf{A}$ конечной сигнатуры σ $\mathbf{A} = \langle A, \sigma \rangle$, где

$$\sigma \quad = \quad \left\langle F_0^{n_0}, \ldots, F_k^{n_k}; P_0^{m_0}, \ldots, P_r^{m_r}; c_0, \ldots, c_s \right\rangle,$$

и задана некоторая нумерация $\nu{:}\mathbf{N} \to A$ такая, что множества $\left\{ \langle n, m \rangle | \nu n = \nu m \right\}$ и $\nu^{-1}(P_i)$ рекурсивны, а для любого $i \le k$ существуют рекурсивные функции f_i такие, что

$$\forall m_1 \ldots \forall m_{n_i} \quad \nu\, f_i(m_1, \ldots, m_{n_i}) = F(\nu m_1, \ldots, \nu m_{n_i})_i.$$

Тогда такая алгебраическая система $\mathbf{A}$ называется *конструктивной*.

Рассматривая сужение введенной выше функции `fromInteger` на множество натуральных чисел, можно показать что алгебраический тип `MyAlg` представляет собой конструктивную алгебраическую систему с нумерацией `fromInteger`. В самом деле, равенству по номерам соответствует сравнение по модулю 5, а необходимые рекурсивные функции f_i для сложения и умножения в алгебре это и есть сами сложение и умножение, то есть

```
> fromInteger(2+5) == fromInteger 2 <> fromInteger 5
True
```

Таким образом, учащимся будут продемонстрированы не только важнейшие понятия теории моделей, но будет возможность в рамках фактически одного решения (программного описания алгебраического типа) связать эти понятия с идеями вычислимости на алгебраических системах.

3. Пример построения композиции алгебраических систем

Следующие примеры позволяют проиллюстрировать такую важную особенность языка Haskell как полиморфизм. Процесс построения ведется в два этапа. На первом этапе мы из конкретных двух линейных порядков (конечного и бесконечного) строим новый линейный порядок, являющийся их суммой. На втором этапе мы создаем конструктор и методы, позволяющие из произвольных двух линейных порядков, представимых средствами Haskell, создать их сумму.

Определение 4. Назовем линейно-упорядоченное множество $\mathbf{L} = \langle L; \le \rangle$ *суммой* $\mathbf{L}_1 + \mathbf{L}_2$ *двух линейно-упорядоченных множеств* $\mathbf{L_1} = \langle L_1; \le \rangle$ и $\mathbf{L_2} = \langle L_2; \le \rangle$, если носитель модели состоит из множества $L = (L_1 \times 1) \cup (1 \times L_2)$, где $1 = \{e\}$, $e \notin L_1, L_2$, а предикат $\le$ интерпретируется таким образом:

$$x \le y \Leftrightarrow (x_2 = y_1 = e) \vee (x_2 = y_2 = e \wedge x_1 \le y_1) \vee (x_1 = y_1 = e \wedge x_2 \le y_2).$$

Если обозначим $\mathrm{Fst}\, L_1 = L_1 \times 1$, а $\mathrm{Snd}\, L_2 = 1 \times L_2$, и $\mathrm{Fst}\, x = (x, e), \mathrm{Snd}\, y = (e, y)$, то определение 4 перепишется в более удобном для дальнейшего виде:

$$u \le v \Leftrightarrow (u = \mathrm{Fst}\, x \wedge v = \mathrm{Snd}\, y) \vee (u = \mathrm{Fst}\, x \wedge v = \mathrm{Fst}\, y \wedge x \le y) \vee (u = \mathrm{Snd}\, x \wedge v = \mathrm{Snd}\, y \wedge x \le y)$$

Для нашей задачи рассмотрим два линейно-упорядоченных множества $\mathbf{L}_1 = Z$ с естественным линейным порядком (т. е. порядковый тип π) и конечное линейно-упорядоченное множество $\mathbf{L}_2 = \{A, B, C, D\}$, и необходимо построить $\mathbf{L} = \mathbf{L}_1 + \mathbf{L}_2$. Тогда решение на Haskell будет иметь следующий вид:

```
type L1 = Integer
data L2 = A | B | C | D deriving (Eq, Ord)
data L = Fst L1 | Snd L2 deriving Eq
instance Ord L where
  Fst _ <= Snd _           = True
  Fst x <= Fst y  | x <= y = True
  Snd x <= Snd y  | x <= y = True
  _ <= _                   = False
```

После этого учащиеся в качестве упражнения могут проверить, например, действительно ли верно, что $\mathrm{Fst}\ 13 \le \mathrm{Snd}\ A$.

Рассмотрим обобщение проблемы. Пусть необходимо построить $\mathbf{L} = \mathbf{L}_1 + \mathbf{L}_2$ для произвольных линейно-упорядоченных множеств $\mathbf{L}_1$ и $\mathbf{L}_2$. Тогда это реализуется в виде следующей программы.

```
data (Ord l1, Ord l2) =>  L l1 l2 = Fst l1 | Snd l2 deriving Eq
instance (Ord l1, Ord l2) => Ord (L l1 l2) where
  Fst _ <= Snd _           = True
  Fst x <= Fst y  | x <= y = True
  Snd x <= Snd y  | x <= y = True
  _ <= _                   = False
```

Решение демонстрирует построение полиморфного алгебраического типа данных, который изоморфен сумме произвольных линейно-упорядоченных множеств $\mathbf{L}_1$ и $\mathbf{L}_2$, заданных алгебраическими типами данных `l1` и `l2`, и которые воплощают порядковый класс `Ord`.

В качестве дополнительных упражнений учащиеся могут на этом или на самостоятельно построенных примерах проверить различные свойства линейных и частичных порядков, например, что наименьший элемент является минимальным, или что наименьший элемент является единственным. В качестве более сложного упражнения можно предложить программно построить такой частичный порядок, у которого есть минимальный элемент, но отсутствует наименьший. Возможно программирование и других операций над порядковыми типами (например, умножение), построение монотонных отображений, проверка их свойств.

4. Категории и функторы

Раздел «Категории и функторы» современной математики, традиционно считающийся областью исключительно «высшей математики», тем не менее, в языке Haskell стал одной из основ языка, без знания которой сложно даже организовать консольный вывод или поддержку состояний во время исполнения программы. Эта особенность отпугивает многих программистов и создает дополнительные трудности при изучении языка новичками без достаточного математического образования. Однако помимо цели дизайнерского решения для сохранения функциональной чистоты языка наличие аппарата теории категорий дает бóльшую выразительность и гибкость языку, повышает уровень модульности программ. В нашем же случае учащиеся фактически ставятся в необходимость изучать элементы теории категорий для понимания и более глубокого использования синтаксических конструкций языка, что является безусловным плюсом, так как нет надобности в дополнительной мотивации. Разумеется, опубликовано уже множество статей, которые пытаются интерпретировать такие понятия теории категорий как функторы, монады, катаморфизмы в терминах и идеях, знакомых работающим программистам [6-9].

В сильно типизированном языке, которым является Haskell, статическая система типов определяет связь между значениями (семантически это результат вычисления выражений) и типами (интуитивно понимаемые как множества значений), и еще в стадии компиляции гарантирует, что программист не допустит ошибок неверной типизации – ошибок, которые так распространены при программировании на слабо типизированных языках.

Вывод типов пользовательских функций осуществляется в большинстве случаев автоматически на основе теоремы Хиндли-Милнера, описывая типы синтаксическими средствами языка [4; 6]. Тогда можно рассмотреть категорию $Hask$, объектами которой будут всевозможные типы языка Haskell, как встроенные, так и определяемые пользователем, а морфизмами – функции на типах [7]. Не нарушая общности (в частности, за счет особой техники, называемой в языке Haskell *каррированием*), мы можем ограничится рассмотрением только одноместных функций. Стандартная функция композиции '.' будет соответствовать композиции морфизмов $\circ_{Hask}$ в категории $Hask$. Тождественная полиморфная функция `id` будет соответствовать единичному морфизму. Тем самым ассоциативность и аксиома единицы тривиальным образом будут выполнены.

Определение 5 [7. С. 7; 10. С. 20]. *Категорией* (малой) будем называть такую систему $\mathbf{C} = \langle O_C, M_C, \circ_C \rangle$, состоящую из множества объектов O_C, множества морфизмов M_C, операции композиции морфизмов $\circ_C$ так, что выполняются следующие аксиомы:

1) каждому морфизму f из M_C соответствует пара объектов, называемых область $\mathrm{dom} f$ и кообласть $\mathrm{codom} f$, что обозначается как $f{:}\mathrm{dom} f \to \mathrm{codom} f$;

2) ассоциативная операция $\circ_C$ каждой паре морфизмов таких, что $\mathrm{dom} f = \mathrm{codom}\, g$, ставит в соответствие морфизм $f \circ g : \mathrm{dom}\, g \to \mathrm{codom} f$;

3) каждому объекту A из M_C соответствует тождественный (единичный) морфизм $\mathrm{id}_A : A \to A$ такой, что для любого морфизма $f : A \to B$ выполнено $\mathrm{id}_B \circ f = f = f \circ \mathrm{id}_A$.

Определение 6 [7. С. 9-11; 10. С. 23-24]. *Функтором* из категории C в категорию D будем называть пару функций $F = \langle F_0, F_1 \rangle$, где $F_0{:}O_C \to O_D, F_1{:}M_C \to M_D$, таких, что:

1) каждому морфизму f из M_C такому, что $f : A \to B,$ соответствует морфизм $F_1(f){:}F_O(A) \to F_O(B)$;

2) $F_1(\mathrm{id}_A) = \mathrm{id}_{F_O}(A)$;

3) $F_1 f \circ_C g() = F_1(f) \circ_D F_1(g)$.

Отображение типов (т. е., объектов категории) делается средствами конструкторов типов либо встроенных, либо задаваемых пользователем. Отображение функций (т. е., морфизмов) делается с помощью следующего, входящего в базовую комплектацию Prelude, определения класса:

```
class Functor f  where
  fmap :: (a -> b) -> f a -> f b,
```

которое перегружает функцию `fmap` для каждого определения функтора.

Такую структуру можно рассматривать также как список. Списки – важная и необходимая часть функциональных языков, начиная от Lisp. В языке Haskell мы можем рассматривать списки с различных точек зрения, в частном, рассмотрим эту структуру как функтор. Для каждого объекта a категории $Hask$ существует объект $[a]$, который представляет собой список элементов, каждый из которых имеет тип a. Отображение морфизмов сделаем, перегрузив функцию `fmap`:

```
fmap f [] = []
fmap f (x:xs) = (f x) : (fmap f xs)
```

Таким образом, мы задали

$$F_0 A = [A], \quad F_1 = \mathrm{fmap}\text{, где } A \text{ – произвольный тип.}$$

Можно доказать, что полученные конструкции действительно задают функтор. Для этого необходимо доказать, что:

$$F_1 \mathrm{id}_A = \mathrm{id}_{F_0 A} \text{ для произвольного типа } A,$$

$$F_1(g \circ f) = F_1 g \circ F_1 f \text{ для произвольных подходящих функций } g, f.$$

Так как список является рекурсивно сконструированным типом, то доказательство мы можем сделать с помощью математической индукции.

I. База индукции (применение к пустому списку)

Сохранение единицы:

```
fmap id [] == [] == id [] {- здесь и ниже псевдо-код на Haskell -}
```

Сохранение композиции:

```
fmap (g . f) [] == [] == fmap g [] == fmap g (fmap f [])
  == (fmap g . fmap f) [].
```

II. Индукционный шаг.

Сохранение единицы:

```
fmap id (x:xs) == (id x) : (fmap id xs) ==
  (id x) : (id xs) == x:xs == id (x:xs)
```

Сохранение композиции:

```
fmap (g . f) (x:xs) == ((g . f) x) : (fmap (g . f) xs) ==
  (g(f(x)) : (fmap g(fmap f xs)) == fmap g ((f x):(fmap f xs))==
  fmap g (fmap f xs) == (fmap g . fmap f) xs
```

Таким образом, наше утверждение доказано.

Несмотря на то, что доказательство несложное, его необходимо разобрать на занятии подробно, так как даже студенты университетов обычно мало знакомы с основами теории категорий.

Заключение

В статье рассмотрены три примера на применение языка к построению теоретико-категорных и алгебраических объектов. Более сложные и содержательные примеры есть как в определениях базового модуля Prelude, поставляемого стандартно вместе с компилятором GHC и настоятельно рекомендуемых к изучению, так и в разнообразной литературе на английском языке. Разумеется, более интересно и важно учащимся самим научиться строить подобные конструкции. Ряд из них уже выполнен силами школьников СУНЦ НГУ на спецкурсе «Введение в функциональное программирование». Важно, что при подобном подходе идеи и конструкции современной математики перестают быть трансцендентными абстракциями для учащихся, но становятся практическим инструментом как в становлении математиков, так и программистов. Тем самым уменьшается барьер между средней школой и вузом, по крайней мере, школьники будут более подготовлены и их математический багаж будет более разнообразен. Конечно, при обучении математически одаренных детей, учащихся с сильной мотивацией и опытом в области программирования, обучение в данном ключе проходит достаточно легко и является прекрасным дополнением к стандартной и факультативной математике. Тем не менее, так как именно прогресс IT-технологий вызвал к жизни идеи функционально-декларативного программирования, то в настоящий момент идет достаточно интенсивное внедрение этих идей как в дизайн новых или уже существующих языков программирования, так и в широкую практику работы программистов, а поэтому знакомство с раскрытой темой полезно для всех интересующихся программированием и математикой.

Литература

1. Никитин, А. А., Алешин, В. Д., Демидко, А. Ю., Михеев, Ю. В., Пащенко М. Г. Информационные и коммуникационные технологии при обучении началам анализа // Тр. Конф. «ИТО-2010».
2. Алешин, В. Д., Демидко, А. Ю., Михеев, Ю. В., Никитин, А. А. Применение информационных и коммуникационных технологий при изучении школьного курса стереометрии // Вестн. Новосиб. гос. ун-та. Серия: Педагогика. – 2007. Т. 8, вып. 2. – С. 2-14.
3. Филд, А., Харрисон, П. Функциональное программирование: Пер. с англ. – М.: Мир, 1993.
4. Haskell 98 Language and Libraries: The Revised Report / Ed. Simon Peyton Jones. Cambridge University Press, 2003. (русский перевод: http://www.haskell.ru)
5. Гончаров, С. С. Счетные булевы алгебры разрешимость. – Новосибирск: Научная книга, 1996.
6. Хьюдак, П., Петерсон, Дж., Фасел, Дж. Мягкое введение в Haskell. (пер. с англ.) URL: http://rsdn.ru/article/haskell/haskell_part1.xml
7. Klinger, S. The Haskell Programmer's Guide to the IO Monad – Don't Panic. Technical report, December 2005, no. 05-54, 33 pp., Centre for Telematics and Information Technology (CTIT).
8. Кирпичев, Е. Монады // RSDN Magazine. 2008. No. 3. С. 21-38. URL: http://rsdn.ru/article/funcprog/monad.xml
9. Meijer, E., Fokkinga, M., Paterson, R. Functional Programming with Bananas, Lenses, Envelopes, and Barbed Wire // Functional programming languages and computer architecture: 5th ACM Conference. Proc. / Ed. by J. Hughes. – Berlin-Heidelberg: Springer-Verlag, 1991.
10. Маклейн, С. Категории для работающего математика / Пер. с англ. под ред. В. А. Артамонова. – М.: Физматлит, 2004.

Компетенции школьного курса Планиметрии

УДК 372.851:514.112

Жафяров Акрям Жафярович
ГОУ ВПО «Новосибирский государственный педагогический университет»
Россия, г. Новосибирск, ул. Вилюйская, д. 28, телефон (383) 244-07-31
nauka2003@rambler.ru

В статье рассматривается связь общечеловеческих компетенций и компетенций в области школьного курса Планиметрии. Представлена оригинальная авторская технология повышения компетентности учителей и учащихся по Планиметрии.

Ключевые слова: компетенция, компетентность, педагогическая технология, профильность, планиметрия.

«Наш прогресс как нации полностью
зависит от прогресса в сфере образования»
Дж. Кеннеди

Нашей стране, как и многим другим, нужны качественные кадры во всех областях деятельности, особенно в решении таких важнейших государственных проблем как:

– развитие демократии и построение цивилизованного общества;

– обеспечение роста духовного и материального благосостояния всех граждан России;

– обеспечение конкурентоспособности и обороноспособности страны и т. д.

О дефиците качественных кадров и о важности решения этой проблемы говорили президенты В. В. Путин и Д. А. Медведев.

В связи с этим система образования получила задание Президента страны Д. А. Медведева найти одаренных детей и обеспечить им наилучшее развитие. Это – первый, очень важный шаг подготовки качественных кадров, необходимый для того, чтобы победить в конкурентной борьбе в областях экономики, обороноспособности и повышения благосостояния народа.

Работа с одаренными детьми, так или иначе, в нашей стране велась и раньше. Переход к профильной системе образования дает существенно большие возможности для достижения значительных результатов в работе с указанной категорией детей.

Психолого-педагогическая наука правильно утверждает, что неодаренных детей нет. Просто одни дети имеют природные способности в одной области, а другие – в другой. Но надо помнить, каждая категория одаренных детей имеет свои специфические особенности, поэтому требуется своя специфическая работа с ними, своя педагогическая технология развития индивидуальных способностей.

Чем умнее ребенок, тем более продуманной, прозрачной, ясной и привлекательной должна быть цель изучения того или другого раздела изучаемой дисциплины. Еще лучше, если ребенок сам участвует в формировании этой цели.

Здесь учитель должен брать на себя роль умного и талантливого режиссера с целью решения проблемы перехода от ученика – объекта воздействия, к ученику – субъекту совместного взаимодействия в деле его образования, просвещения и развития.

© *А. Ж. Жафяров, 2011*

Ясно, что реализация указанного взаимодействия является напряженным психологическим и физиологическим трудом как для учителя, так и для учащегося. Если труд не получит своего морального и материального вознаграждения, то не будет ни учителя, ни учащегося. В итоге на ниве системы образования возникнет «серое облако», состоящее из так называемых участников образовательного и просветительского процесса.

Труд этот – государственно значимый, он должен быть морально и материально оплаченным. Особо отметим, что участники должны получить радость от этого образовательного процесса. Прав и сегодня К. С. Станиславский: артист работает на 110 %, если работа приносит ему радость.

Работа на 110 % – это проверенный веками и многими странами метод решения практически любых проблем реальной жизни. Он должен быть принят во всей системе образования, управления образованием и выше.

Из выше сказанного следует, что в учебном процессе должна быть принята не диктаторская, а диалоговая система образования, ниже приводится эта модель (см. рис. 1).

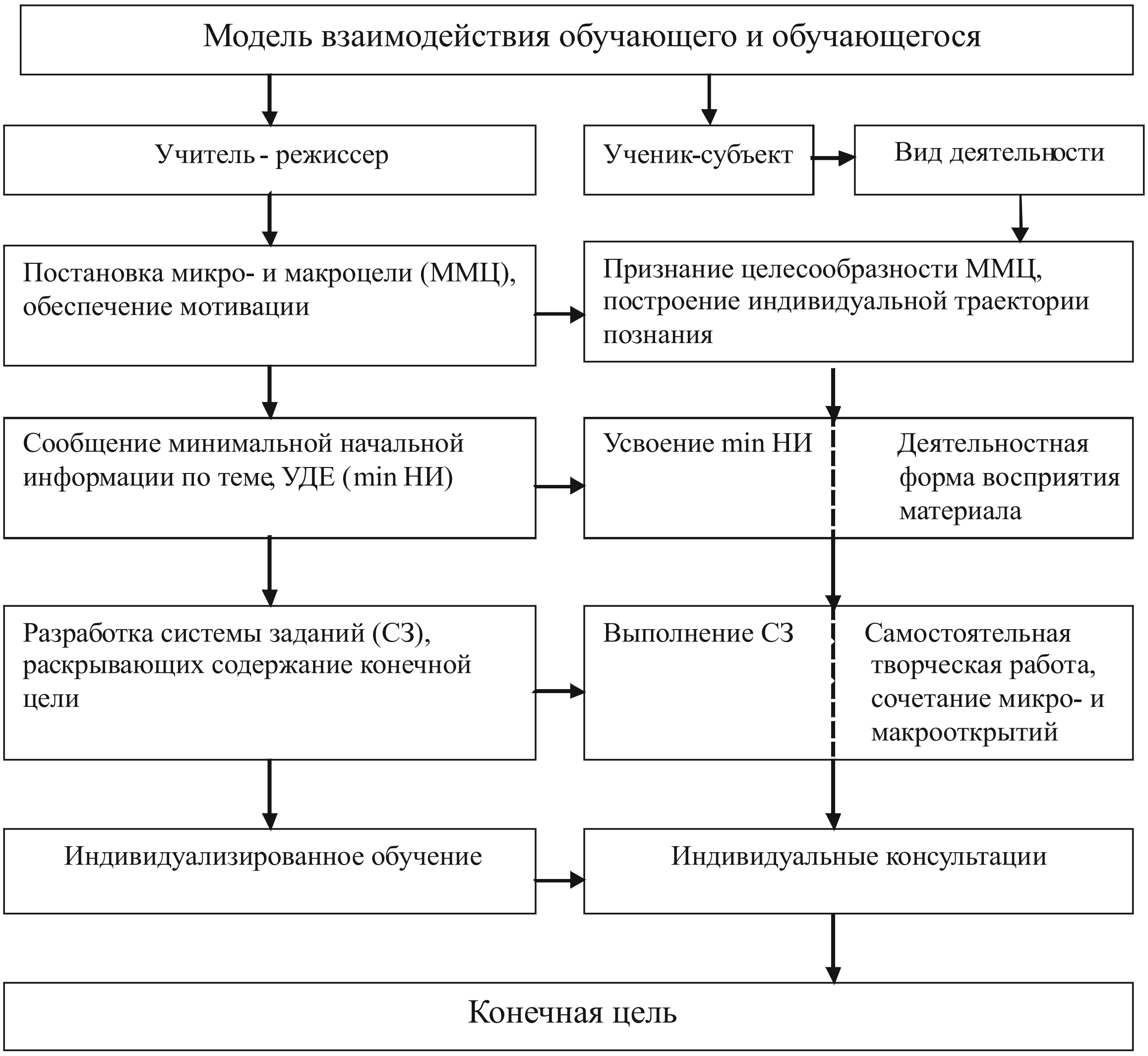


Рис. 1.

В решении подготовки качественных кадров школа играет очень важную роль. Школьная система образования должна обеспечивать решение следующих задач:

1) выявить максимально объективно и тщательно «кто есть кто», т.е. по Канту «вещь в

себе сделать вещью для себя»;

2) наилучшим образом развивать природные склонности и способности учащихся, сформировать творческую личность;

3) сохранить их физическое и психическое здоровье;

4) воспитать в духе патриотизма и признания общечеловеческих ценностей;

5) социально защитить ученика, дав ему возможность получить более высокое образование, а затем на этой базе и достойную профессию.

Эти задачи носят философско-методологический характер, поэтому они не понятны и не затрагивают ученика. Перед учащимися должны быть поставлены созвучные им цели. **Ближайшая цель** – успешная сдача ЕГЭ по математике и подготовка фундаментальной базы, которая способствовала бы успешной учебе в вузе, причем по специальности, соответствующей его склонностям, способностям и интересам.

Стратегическая цель – развитие индивидуальных способностей, формирование компетентной и творческой личности. Ниже приведена модель технологии изучения Планиметрии (рис. 2), реализация которой будет способствовать достижению указанных выше целей.

Рис. 2. Модель технологии изучения Планиметрии

Особо отметим, что в этой модели экспертом выступает сама жизнь. Этапы экспертизы: успешность сдачи ЕГЭ по математике, успешность учебы в вузе и, наконец, сформированность личности.

Коротко опишем компоненты модели технологии изучения Планиметрии.

Многие выдающиеся педагоги (В. П. Беспалько, В. П. Околелов, Ю. Г. Татус, Ф. Янушкевич и др.) придерживаются мнения, что педагогическая технология всегда присутствует в процессе обучения и воспитания. Автор придерживается такого же мнения.

По мнению В. П. Беспалько, Т. И. Шамовой, Т. М. Давыденко технология разрабатывается под конкретный педагогический замысел, в основе ее лежат ценностные ориентации, целевые установки автора или коллектива, имеющие формулу конкретного ожидаемого результата. Массовую разработку и внедрение *педагогических технологий* можно отнести к середине 50-х годов. Это связано с возникновением технологического подхода к обучению вначале в американской, а затем и в европейской школе. В настоящее время, несмотря на то, что понятие педагогической технологии прочно вошло в педагогический лексикон, в его понимании существуют большие разночтения.

За рабочее определение педтехнологии возьмем то, что изложено в трудах таких ученых, как Г. К. Селевко, Л. В. Загрекова и В. В. Николина.

Педагогическая технология – это учебный процесс и сопровождающая его методическая система, обладающая следующими шестью признаками: концептуальность, актуальность, системность, управляемость, эффективность и воспроизводимость.

В последнее время широкое внедрение получили такие важные понятия как компетенция и компетентность. Эти понятия введены с целью приближения образования к реальной жизни. Традиционная система ЗУНов устарела, вызвано это устарение увеличением скорости удвоения научных результатов. Современный компетентный специалист должен обладать по крайней мере тремя параметрами: иметь прекрасные знания, умения и навыки (ЗУНы); быть нравственно воспитанным и признавать общечеловеческие ценности; нацеленным на творчество и инновации. Специалист, владеющий только первым параметром (ЗУНами), является завтрашним «бомжом».

Поэтому прогрессивным является введение непрерывной системы образования. Понятия компетенция – компетентность отражают сущность этой системы.

Однако следует отметить, что в толковании этих понятий существуют разночтения. Автор предлагает ввести определения этих понятий следующим образом.

Компетенция в данной области – это название вида деятельности человечества в этой области. Ее сущность состоит в **необходимости** решать относительно конкретные проблемы указанной области.

В качестве пояснения приведем пример одной из компетеций Планиметрии, он имеет номер – КП-4 (**четвертая компетенция Планиметрии**) – это **необходимость** владения **функциональным подходом в геометрии**. Отсюда следует: КП-4 – название вида деятельности человечества в Планиметрии, в данном случае – это необходимость владения функциональным подходом в Планиметрии, т. е. решать Планиметрические задачи на основе таких понятий математического анализа как функция, производная функции, геометрические преобразования и т. д

Компетентностью назовем уровень владения личностью соответствующей компетенцией в данной области деятельности человечества. Основные методы формирования компетентностей указаны в соответствующей модели (см. рис. 2). Важными среди них являются индивидуализация и дифференциация, деятельностный подход и самостоятельная работа обучаемых на основе информационных технологий. Желательно все это направить на творческое развитие ученика.

Более подробно рассмотрим компетентностный подход в Планиметрии. Для достижения поставленной цели предлагается следующий алгоритм:

1) выявить компетенции Планиметрии;

2) изучить связи этих компетенций как с общечеловеческими компетенциями, так и с типичными ошибками, допускаемыми абитуриентами на ЕГЭ; установить причины, порождающие отмеченные ошибки;

3) наметить пути устранения этих причин, в частности кратко излагать теорию (в случае необходимости), привести демонстрационные примеры.

Первая общечеловеческая компетенция – ОЧК-1 состоит в необходимости соблюдения **законов природы**. Природа «живет» уже давно по установившимся причинно-следственным законам. Вмешательство «локальных» специалистов может нарушить связь и многовековую устойчивость в природе. Немало случаев, когда природа наказывает человечество за безграмотность.

Проекцией этой первой ключевой компетенции на Планиметрию является необходимость владения **школьными стандартами.** Надо четко знать определения понятий, формулировки теорем, уметь применять понятия и их свойства для решения учебных и практико-значимых задач. Это и составляет **первую компетенцию Планиметрии** – КП-1. Как говорят, нет ничего практичнее, чем хорошо разработанная теория. В геометрии исторически с древних греков и египтян сложилась научная теория, отражающая многие факты реальной жизни. Действительно, «наука сокращает нам опыты быстро текущей жизни».

Поэтому надо выучить азы **Планиметрии**, другое – не дано. Иначе говоря, надо «жить дружно» с законами природы.

В связи со сказанным представляет интерес вопрос о том, как обстоит дело со знанием стандартов по Планиметрии. В последние годы результаты ЕГЭ говорят о низком уровне подготовленности выпускников по математике вообще, но Планиметрии в частности.

Учащиеся, сдающие ЕГЭ, теряют много баллов из-за элементарного незнания основных теорем геометрии.

Вторая общечеловеческая компетенция – ОЧК-2 – **нацеленность** на инновации и творчество, совершенствование знаний, самоусовершенствование.

В наш стремительный XXI век, когда скорость удвоения научной информации существенно возросла (по информатике удвоение происходит за 1,5 года; по фундаментальной науке, применяемой в военной области или конкурентной экономике – за 3–4 года), просто знающий каноны на сегодняшний день не является квалифицированным специалистом. Он должен постоянно усовершенствовать свои знания (непрерывное образование), причем творчески относясь к этому делу.

В указанном отношении школьная геометрия является благодатной дисциплиной. Сами по себе задачи на построение на плоскости и в пространстве в сочетании с метрикой (вычислениями) способствуют развитию пространственного воображения, конструктивного и логического мышления.

Очень важным подспорьем для развития творчества обучающихся (учащихся, учителей и т. д.) являются метод проектов, конструирование новых задач, решение задач с параметрами, нестандартных задач и т. д.

Вторая компетенция Планиметрии – КП-2 является самой трудной и связана, как уже отмечено выше, с развитием и саморазвитием **творчества**.

Методов развития творчества невозможно даже определить, здесь имеем дело с большой **неопределенностью**. Но многие способы известны, здесь отметим один из важных методов – метод конструирования задач, создания семейства задач, т.е. задач, имеющих «родство».

Развитию творчества, вариативного мышления и фундаментальности знаний способствует и **решение задач с параметрами**.

Этот вид деятельности является очень слабым у большинства учащихся. В школьных учебниках нет системы подготовки учащихся к решению задач с параметрами ни по алгебре, ни по геометрии. Большинство учителей также не владеет этим материалом.

Несмотря на это, в последние годы на ЕГЭ все больше и больше предлагают задачи с параметрами. Именно такие задачи дают большое количество баллов, естественно при правильном решении и обосновании.

Раньше на вступительных экзаменах многие вузы зачисляли абитуриентов в студентов на основе ЗУНов – знаний, умений и навыков. Это был не оптимальный принцип отбора, потому что не учитывался второй интегрированный параметр абитуриента – его **интеллектуальный потенциал и фундаментальность** знаний.

Указанные очень важные особенности абитуриента наилучшим образом раскрывают задачи с параметрами, т. е. задачи с параметрами служат своего рода лакмусовой бумажкой в деле определения качества будущего студента – это **первая** причина увеличения числа задач с параметрами.

Вторая, не менее важная причина, связана с тем, что такие задачи:

а) **лучше содействуют развитию личности ученика**, его индивидуальных склонностей и способностей;

б) учат работать в условиях **небольших и значительных неопределенностей**, чем изобилирует наша сегодняшняя жизнь;

в) развивают **творческое и вариативное мышление** и тем самым способствуют развитию интеллекта и повышению уровня фундаментальности знаний.

Третья общечеловеческая компетенция – ОЧК-3 состоит в необходимости активного участия граждан в построении цивилизованного общества, общества «без двойных стандартов», «без базара» и т. д. Иначе говоря, должна быть достигнута однозначная трактовка законов общества, причинно-следственных заключений и т. д. В математике строгая логика изложения достигается за счет аксиоматического построения теории, соблюдения законов формальной логики, обеспечения равносильности утверждений, что и составляет **третью компетенцию Планиметрии** – КП-3. К сожалению, из программы общеобразовательной школы исключено понятие равносильности математических объектов (уравнений, неравенств, систем, совокупностей).

Третья типичная ошибка абитуриентов в этом и состоит. Решения задач ЕГЭ уровня С должны быть обоснованы. К сожалению, таких работ очень мало, нет причиннодетерминированного изложения. Поэтому многие получают низкие баллы. Сохранение равносильности исключительно важно при решении уравнений, неравенств и т. д. Многие преобразования нарушают равносильность: можно потерять решения, можно и приобрести посторонние решения. В обоих случаях за счет дополнительных исследований приходится восстанавливать равносильность. К этому и не готовы многие выпускники. Во многих случаях причиной возникновения третьей типичной ошибки является то, что понятие равносильности исключено из программы общеобразовательной школы, исключение возможно составляют только спецклассы.

Четвертая общечеловеческая концепция ОЧК-4 состоит в необходимости принимать рискованные, но вместе с тем научно-обоснованные решения в экстремальных ситуациях.

Четвертая компетенция Планиметрии как и всей математики – КП-4 состоит в необходимости решать теоретические и прикладные задачи на экстремум. Функциональный подход в **Планиметрии** (содержащий такие понятия как производная, интеграл, геометрические преобразования) закладывает научные основы принятия оптимальных решений в экстремальных случаях.

Другие компетенции Планиметрии порождены определением предмета математики. По Ф. Энгельсу предметом изучения математики являются пространственные формы и количественные соотношения объектов реального мира.

Поэтому за **пятую компетенцию Планиметрии** – КП-5 возьмем необходимость решать задачи на инциденцию (взаимное расположение) фигур на плоскости. Решение задач на построение относится к этой компетентности.

Указанная компетенция относится к той части определения предмета математики, которая посвящена пространственным формам (в первую очередь).

В реальной жизни прежде чем строить грандиозные здания или создать производственные механизмы, человечество должно продумать, научно обосновать проекты, т.е. решать позиционные и метрические задачи о пространственных формах и количественных соотношениях. Однако, ситуация существенно осложняется, если имеем дело с микро- или макрообъектами.

Мелкие детали надо увеличить, а крупные уменьшить, чтобы на листке бумаги (экране компьютера) можно было решать практикозначимые задачи. Причем надо сделать так, чтобы чертежи, сделанные, например в Москве, можно было использовать в любой другой точке земного шара (наука не имеет ни национальных, ни государственных границ!)

Решить эту проблему может только наука! Человечеством разработана теория изображения плоских и неплоских фигур, вполне разрешающая указанную общечеловеческую проблему. Эти законы (теория и навыки применения) частично имеют место в школьном курсе геометрии, особенно они необходимы для решения геометрических задач ЕГЭ. Поэтому такой вид деятельности (компетенцию) учащихся должны усвоить.

Следующая компетенция связана именно с этой проблемой геометрии – изображением фигур-оригиналов при проектировании (параллельном или центральном) на плоскость.

Шестой компетенцией Планиметрии – КП-6 является изображение плоских фигур. Оно основано на двух теоремах. Сначала введем соответствующее определение.

Пусть множество $F_0 = \pi(F')$ является **проекцией фигуры** F' на плоскость α_0 при параллельном проектировании $\pi : F' \to \alpha_0$ фигуры F' на плоскость α_0. Рассмотрим α – некоторую плоскость. Любая фигура $F \subset \alpha$, подобная фигуре F_0, **называется изображением фигуры F' на плоскости α**.

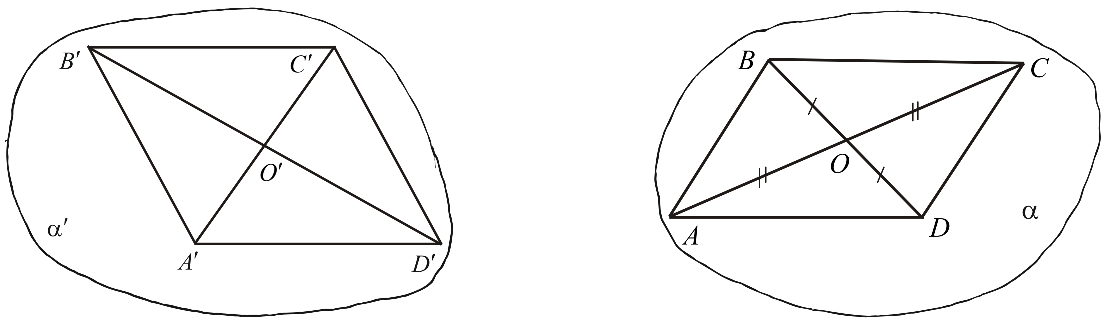
Теорема 1. Любой треугольник ABC плоскости изображений является изображением любого данного треугольника $A'B'C'$ – оригинала.

Теорема 2. Если на плоскости α изображений дан ΔABC, являющийся изображением некоторого $\Delta A'B'C'$ плоскости α', то любая точка плоскости α' однозначно изображается на этой плоскости α.

Используя предыдущие две теоремы, построим изображения изучаемых в школе плоских фигур.

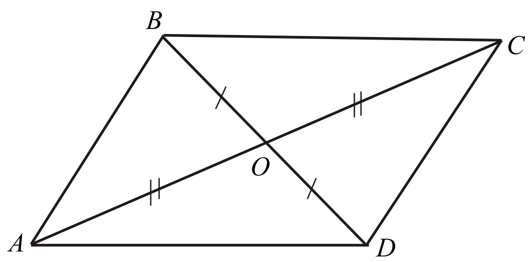
1. Треугольник. Из теоремы 1 следует, что любой треугольник на плоскости изображений α является изображением любого треугольника – оригинала.

2. Параллелограмм. Пусть $A'B'C'D'$ – параллелограмм – оригинал. Построение изображения $ABCD$ на плоскости изображений α можно выполнить двумя способами. Первый



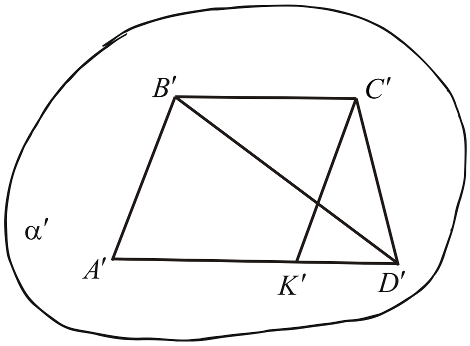
шаг изображения – один и тот же: любой треугольник ABD на плоскости α считаем изображением $\Delta A'B'C'$. Построение изображения точки C' можно провести двумя способами. **Первый** основан на том, что при параллельном проектировании сохраняется параллельность прямых. Поэтому точка C является пересечением двух прямых, проходящих через точки B и D, параллельных соответственно прямым AD и AB.

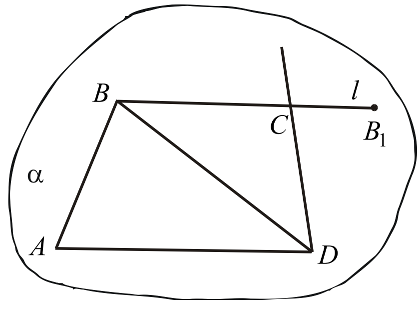
Второй способ является универсальным, им доказана теорема 2. Точку C' соединяем с точкой A'. Пусть $O' = B'D' \cap A'C'$. Изображение O точки O' однозначно определено. Как известно, O' – середина отрезка $B'D'$, поэтому O – середина BD. Далее строим точку C, зная, что O – середина отрезка AC.



3. Трапеция. Пусть $A'B'C'D'$ – трапеция, $ABCD$ – ее изображение.

Алгоритм построения изображения:





1-й шаг. Произвольный треугольник ABD на плоскости изображений α считаем изображением $\Delta A'B'D'$.

2-й шаг. Через точку B проводим прямую l, $l \parallel AD$.

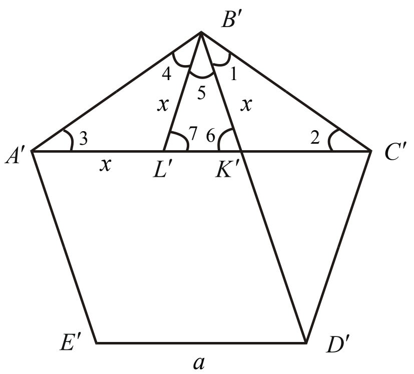
3-й шаг. На луче BB_1, принадлежащем полуплоскости, определяемой прямой AB и точкой D, строим точку C так, чтобы

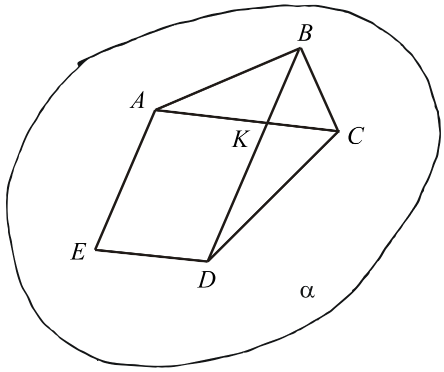
$$\frac{BC}{AD} = \frac{B'C'}{A'D'} = \frac{1}{\beta}.$$

Трапеция $ABCD$ – изображение $A'B'C'D'$.

4. Правильный пятиугольник (трудная, но содержательная задача).

Пусть $A'B'C'D'E'$ – правильный пятиугольник, α – плоскость изображений. Алгоритм построения изображения:





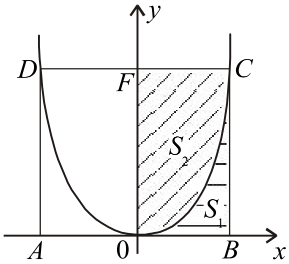
1-й шаг. Произвольный ΔABC на плоскости α считаем изображением $\Delta A'B'C'$.

Если бы построили точку K – изображение точки K', то данный пятиугольник легко изобразить. В самом деле, пусть точка K построена. Тогда точка D, с одной стороны, принадлежит лучу $[BK)$, с другой, $\frac{BK}{KD} = \frac{B'K'}{K'D'} = \beta$, где β – известное число (его можно вычислить). Итак, точку D можно построить. Точка E может быть построена как пересечение двух прямых, проходящих через A и D, параллельных соответственно прямым BD и AC.

Седьмой компетенцией Планиметрии – КП-7 является необходимость владения векторно-координатным методом для решения задач по Планиметрии. Этот метод существенно значим для решения задач стереометрии. Но он применим и в Планиметрии, особенно для решения задач, которые трудно решить чисто геометрическими (синтетическими) методами, таких как задачи о точке пересечения трех высот треугольника, расстоянии от точки до прямой и т. д.

Приведем пример.

Пример. В каком отношении делится площадь квадрата параболой, проходящей через две его соседние вершины и касающейся его стороны в ее середине?



Решение. Пусть $ABCD$ – произвольный квадрат. Выберем систему координат Oxy, где O – середина отрезка AB. Длину отрезка OB примем за единицу длины (за масштаб). Тогда точки O, B, C, D имеют соответственно координаты: O (0; 0), B (1; 0), C (1; 2), D (–1; 2).

Найдем уравнение параболы, проходящей через точки C и D, касающаяся оси Ox в точке O (0; 0). Пусть

$$y = ax^2 + bx + c \tag{1}$$

уравнение искомой параболы. Тогда справедливы следующие равенства:

$$\begin{cases} c = 0, \\ 2 = a + b + c, \\ 2 = a - b + c \end{cases} \Leftrightarrow \begin{cases} c = 0, \\ a = 2, \\ b = 0 \end{cases} \tag{2}$$

Учитывая (1) и (2), получим, что уравнением искомой параболы является: $y = 2x^2$.
Отношение площадей частей, на которые парабола делит квадрат, равно S_1 / S_2, где

$$S_1 = \int_0^1 2x^2 dx, \quad S_2 = S_{OBCF} - S_1 = 2 - S_1.$$

Вычислим интеграл

$$S_1 = \frac{2}{3}x^3 \Big|_0^1 = \frac{2}{3}.$$

Тогда

$$S_2 = 2 - \frac{2}{3} = \frac{4}{3}, \quad S_1 / S_2 = \frac{1}{2}.$$

Ответ: 1:2.

Литература

1. Белоносов, В. С. Задачи вступительных экзаменов по математике / В. С. Белоносов, М. В. Фокин. – Новосибирск: Сиб. унив. изд-во, 2003. – 560 с.
2. Жафяров, А. Ж. Профильное обучение математике старшеклассников: учебно-дидактический комплекс / А. Ж. Жафяров. – Новосибирск: Сиб. унив. изд-во, 2003. – 468 с.
3. Жафяров, А. Ж. Индивидуализация и дифференциация в педагогической теории и практике: анализ отечественного опыта / А. Ж. Жафяров, Е. С. Никитина, М. Е. Федорова. – Новосибирск: Изд. НГПУ, 2004. – 36 с.
4. Жафяров, А. Ж. Элективный курс с электронным обеспечением «Логарифмические и показательные уравнения, неравенства, системы и совокупности с параметрами. ЕГЭ – уровень С» / А. Ж. Жафяров. – Новосибирск: Изд. НГПУ, 2008. – 108 с.
5. Кутасов, А. Д. Пособие по математике для поступающих в вузы / А. Д. Кутасов, Т. С. Пиголкина, В. К. Чехлов, Т. Х. Яковлева. – М.: Наука, 1988. – 720 с.
6. Шамова, Т. И. Управление образовательными системами / Т. И. Шамова, Т. М. Давыденко, Г. Н. Шибанова. – М.: Издательский центр «Академия», 2002. – 384 с.