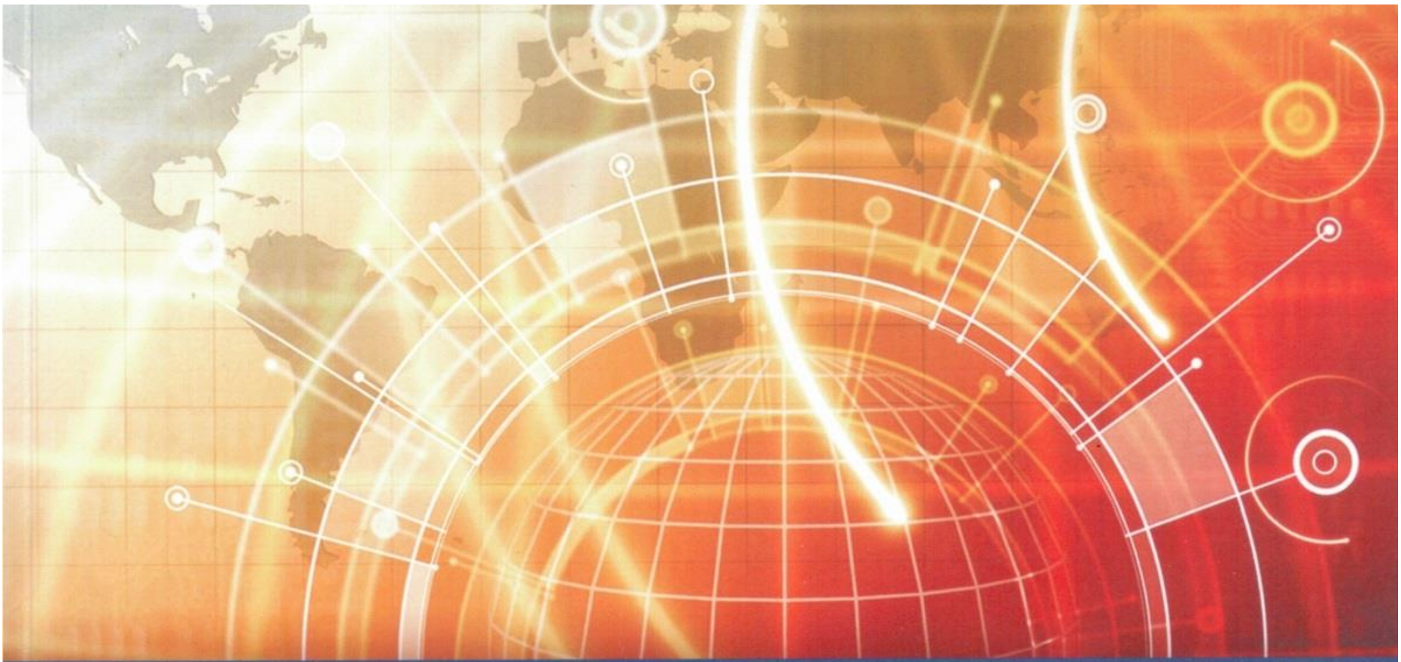

ИНФОРМАТИЗАЦИЯ ОБРАЗОВАНИЯ И НАУКИ

Государственная политика в области информатизации образования и науки

Информационные технологии

Телекоммуникации

Системы защиты информации

Автоматизация и управление технологическими процессами и производствами

Системный анализ, управление и обработка информации

Управление в социальных и экономических системах

Министерство образования и науки Российской Федерации

Informika

Федеральное государственное автономное учреждение «Государственный научно-исследовательский институт информационных технологий и телекоммуникаций»

№4 [32]

ОКТЯБРЬ 2016

Научно-методический журнал
«Информатизация образования и науки»
№ 4(32) / 2016

Учредитель:
Федеральное государственное автономное учреждение «Государственный научно-исследовательский институт информационных технологий и телекоммуникаций» (ФГАУ ГНИИ ИТТ «Информика») Министерства образования и науки Российской Федерации

Тел. 8 (495) 969-26-17 доб. 1112

Журнал включен в Перечень ведущих рецензируемых научных журналов и изданий ВАК

Тираж журнала
500 экз.

Зарегистрирован в Федеральной службе по надзору в сфере связи, информационных технологий и массовых коммуникаций (Свидетельство о регистрации средства массовой информации ПИ № ФС77-48849 от 7 марта 2012 г.)

Подписной индекс 32788 в каталоге «Газеты. Журналы» ОАО Агентства «РОСПЕЧАТЬ»

Отпечатано в типографии ФГАУ ГНИИ ИТТ «Информика»
Адрес: 125009, Москва, Брюсов пер., д. 21

По вопросам редакционной подписки обращаться по адресу:
125315, Москва, ул. Часовая, д. 21/Б, ком. 31

СОДЕРЖАНИЕ

СИСТЕМНЫЙ АНАЛИЗ, УПРАВЛЕНИЕ И ОБРАБОТКА ИНФОРМАЦИИ

МЕТОДИКА РЕШЕНИЯ МАТЕМАТИЧЕСКИХ ЗАДАЧ С ПОМОЩЬЮ ПРОГРАММИРОВАНИЯ И КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ

METHODOLOGY OF SOLVING MATHEMATICAL PROBLEMS WITH PROGRAMMING AND COMPUTER MODELING

Симаков Егор Евгеньевич / Egor E. Simakov,
учитель информатики и математики МБОУ Лицей №1 г. Южно-Сахалинска / Teacher of Informatics and Mathematics, Liceum № 1, Yuzhno-Sakhalinsk,
s-im1a@yandex.ru

Аннотация

Статья посвящена рассмотрению метапредметного подхода к преподаванию математики с использованием средств ИКТ, а также методике применения приемов программирования и вычислительных экспериментов в профильном обучении для решения математических и прикладных задач. В статье описываются основные принципы применения данной методики при изучении отдельных тем математики, приводятся примеры алгоритмов и их реализация в различных программных средах.

Abstract

The article considers meta-subject approach to teaching mathematics using ICT tools, as well as the method of application programming techniques and computational experiments in profile training to solve mathematical and applied problems. The article describes the basic principles on the use of this technique in the study of selected topics of mathematics. Examples of algorithms and their implementation in a variety of software environments are presented.

Ключевые слова: метапредметный подход, прикладное программирование, профильное обучение, математическое моделирование.

Keywords: meta-subject approach, application programming, specialized education, mathematical modeling.

Введение

Математическое образование вносит свой вклад в формирование общей культуры человека. Необходимым компонентом общей культуры в ее современном толковании является общее знакомство с методами познания действительности, что включает понимание диалектической взаимосвязи математики и действительности, представление о предмете и методе математики, его отличиях от методов естественных и гуманитарных наук, об особенностях применения математики для решения научных и прикладных задач. Изучение математики способствует всестороннему развитию личности, пониманию красоты и изящества математических рассуждений, восприятию геометрических форм, усвоению идеи симметрии. Изучение математики развивает воображение, пространственные представления.

Роль математической подготовки в общем образовании современного человека ставит следующие цели обучения математике:

1) овладение конкретными математическими знаниями, необходимыми для применения в практической деятельности, для изучения смежных дисциплин, для продолжения образования;

2) интеллектуальное развитие учащихся, формирование качеств мышления, характерных для математической деятельности и необходимых для продуктивной жизни в обществе;

3) формирование представлений об идеях и методах математики, о математике как форме описания и методе познания действительности;

4) формирование представле-

ний о математике как части общечеловеческой культуры, понимания значимости математики для общественного прогресса.

Основная задача обучения математике в школе заключается в обеспечении прочного и сознательного овладения учащимися системой математических знаний и умений, необходимых в повседневной жизни и профессиональной деятельности каждому человеку, достаточных для изучения смежных дисциплин и продолжения образования.

Наряду с решением основной задачи расширенное и углубленное изучение математики предусматривает формирование у учащихся устойчивого интереса к предмету, выявление и развитие их математических способностей, ориентацию на профессии, существенным образом связанные с математикой, подготовку к обучению в вузе.

Основы методики решения математических задач с помощью программирования и компьютерного моделирования

Предлагаемая автором методика решения математических задач с использованием программирования направлена на углубление знаний учащихся по теории вероятности, стереометрии, математическому анализу; развитие логического мышления, а также на удовлетворение познавательных интересов и развитие способностей учащихся.

Изучение основ алгоритмизации и программирования способствует решению таких задач математического образования, как:

– работа над формированием сознательного и прочного овладения системой математических знаний, умений, навыков;

– систематизация, расширение и углубление знаний по алгебре и началам математического анализа, стереометрии, теории вероятностей;

– развитие математических способностей учащихся;

– иллюстрация межпредметных связей математики и информатики;

– вовлечение учащихся в самостоятельную исследовательскую деятельность.

В Концепции математического образования [1] обозначены основные принципы обучения математике. Среди них выделено внедрение средств информатики и ИКТ в преподавание математики. Введение в школьную программу информатики дало возможность снять многие возникающие в процессе обучения познавательные трудности, вызвать интерес у учащихся к физическим и математическим задачам, связанный с возможностью их решения новыми, нестандартными методами. Помимо этого, комплексный подход к обучению естественно-математическим предметам на основе информатики позволяет решить и проблемы обучения самой информатике. Например, предлагаемые в учебниках информатики задачи зачастую не имеют реальной практической ценности, выглядят формальными и не вызывают интереса у школьников. Использование же компьютера по его прямому назначению (для решения практических задач, для выполнения громоздких, малоинтересных вычислений, для обработки большого объема информации и др.) усиливает практическую направленность как физики, математики, так и информатики; отражает современные методы исследования в этих отраслях научного знания, способствует устойчивому интересу учащихся к изучаемым предметам.

Методика решения математических задач с помощью программирования реализует взаимосвязь между предметами математики и информатики. Основная идея, положенная в основу методики, – изучение вариантов решения математических, комбинаторных, стереометрических задач с помощью компьютера. В школьном курсе математика и информатика рассматриваются как две отдельные дисциплины. Предлагаемая методика призвана показать учащимся непрерывную связь этих предметов. Она дает возможность применять коллективные, групповые и индивидуальные формы работы.

В ходе обучения решению математических задач через компьютерное моделирование учащиеся получают сведения о тесной связи с информатикой различных направлений математики.

Комбинаторика и теория вероятностей. В математике некоторые задачи можно решить только путем перебора огромного количества вариантов и сложных вычислительных операций. Существует большой класс комбинаторных задач, решение которых стало возможно лишь с появлением электронных вычислительных машин. Комбинаторные задачи имеют огромное практическое применение при решении прикладных задач. Их решение способствует повышению математической и алгоритмической культуры. Комбинаторные задачи представляют богатый материал для изучения основных конструкций, методов и приемов программирования, позволяют показать возможности новых компьютерных технологий при решении практических математических задач. Задачи дискретной математики, к которым относятся многие задачи практического программирования и большинство олимпиадных задач по информатике, часто сводятся к перебору различных комбинаторных конфигураций объектов и выбору среди них наилучшего, с точки зрения условия той или иной задачи. Поэтому знание алгоритмов генерации наиболее распространенных комбинаторных конфигураций является необходимым условием успешного решения задач в целом.

Задачи стереометрии. Пространственное воображение учащихся, необходимое для решения задач геометрии, на начальном этапе обучения находится на низком уровне. Основой процесса формирования и развития пространственного мышления является практическая работа учеников с пространственными объектами, манипулирование ими, изменение их положения в пространстве. Правильное выполнение чертежа имеет большое значение для отыскания плана решения задач и наоборот, неверное выполнение чертежа часто не позволяет «увидеть» нужные соотношения. В ходе решения задач изучаются следующие методы построения сечений в многогранниках:

- метод следов;
- метод внутреннего проектирования;
- метод деления n-угольной призмы (пирамиды) на треугольные;
- метод параллельных прямых, параллельный перенос.

При этом используется построение сечений многогранника с помощью компьютерных программ, имеющее ряд преимуществ перед обычным способом построения: активизирует самостоятельную деятельность учащихся, способствует рефлексии действий учащихся, предотвращает на ранних этапах неверное решение задач и экономит учебное время. Все это способствует развитию у учащихся пространственных представлений; формированию понятий математической модели; раскрытию прикладных возможностей геометрии. Его развитие облегчается применением компьютерных программ. Как показывает ЕГЭ по математике, процент выполнения практических заданий по стереометрии остается низким.

Алгебра и начала математического анализа. Изучению функций и построению их графиков отводится важное место в программе школьного курса обучения математике. Здесь закладываются основы аналитического мышления, развивается логика, формируются математическая интуиция и навыки уверенного владения методами графического решения уравнений и неравенств. Исследование сложных функциональных зависимостей, включающих модули, тригонометрические функции, степени с рациональным показателем, логарифмические и показательные функции и др., а также построение графиков подобных функций затруднительно для учащихся. И здесь на помощь приходят компьютерные программы. Особое внимание уделяется заданиям, формирующим математическое мышление, способствующим обучению школьников графическому языку. В число таких упражнений, кроме задач на геометрические преобразования графиков, входят задания на построение графиков функций элементарными методами, упражнения по переводу со словесного описания поведения функций на графический язык и обратно.

Такое распределение практических занятий предполагает постепенное и наиболее качественное усвоение учащимися фундаментальных понятий комбинаторики, стереометрии, алгебры и начал математического анализа. Возможность перейти к автоматизированным действиям на компьютере позволяет более полно и быстро разобрать большее количество заданий. Данная методика является средством развития мышления учащихся, формирования приемов умственной деятельности, кроме этого поддерживается на достаточно высоком уровне познавательный интерес учащихся и к математике, и к информатике, идет укрепление межпредметных связей.

Этапы решения математических задач с помощью программирования и компьютерного моделирования

Процесс моделирования включает три элемента:

- субъект (исследователь);
- объект исследования;
- модель, определяющую отношения познающего субъекта и познаваемого объекта.

Процесс разработки моделей и их исследование с помощью компьютера включает несколько этапов.

1. Создание математической модели задачи:

1) описательная модель (выделение существенных свойств);

2) формальная модель (запись модели с помощью формального языка (формул, уравнений)).

2. Создание компьютерной модели (создание проекта на языке программирования или построение модели с использованием приложений):

1) условного образа объекта, описанного с помощью взаимосвязанных компьютерных таблиц, блок-схем, диаграмм, графиков и т.д. и отображающий структуру и взаимосвязи между элементами объекта;

2) отдельной программы, позволяющей с помощью вычислений и визуализации результатов имитировать процессы функционирования объекта при условии воздействия различных факторов.

3. Проведение компьютерного эксперимента над математической моделью объекта исследования, который состоит в том что, по одним параметрам модели вычисляются другие и на этой основе делаются выводы о свойствах изучаемого объекта.

4. Анализ результатов и корректировка модели.

5. Визуализация моделей.

Моделирование – циклический процесс. Это означает, что за первым циклом может последовать второй, третий и т.д. При этом знания об исследуемом объекте расширяются и уточняются, а исходная модель постепенно совершенствуется. Недостатки, обнаруженные после первого цикла моделирования, обусловленные малым знанием объекта или ошибками в построении модели, можно исправить в последующих циклах.

Примеры решения математических задач с помощью программирования и компьютерного моделирования

Пример 1. Составить программу для реализации алгоритма «Решето Эратосфена».

Данный алгоритм используется для нахождения всех простых чисел до некоторого целого числа N. Он осуществляет, своего рода, фильтрацию всех чисел, за исключением простых. По мере перебора составные числа отбрасываются, и в результате остается последовательность простых чисел. Алгоритм состоит из следующих шагов:

- выписать подряд все целые числа от 2 до N;
- первое простое число равно 2 (оно фиксируется);

– вычеркнуть все числа, большие 2 и кратные 2;

– первое из оставшихся чисел 3 является простым (зафиксировать);

– вычеркнуть все числа, большие 3 и кратные 3 и т.д.

Представим данный алгоритм в виде блок-схемы.

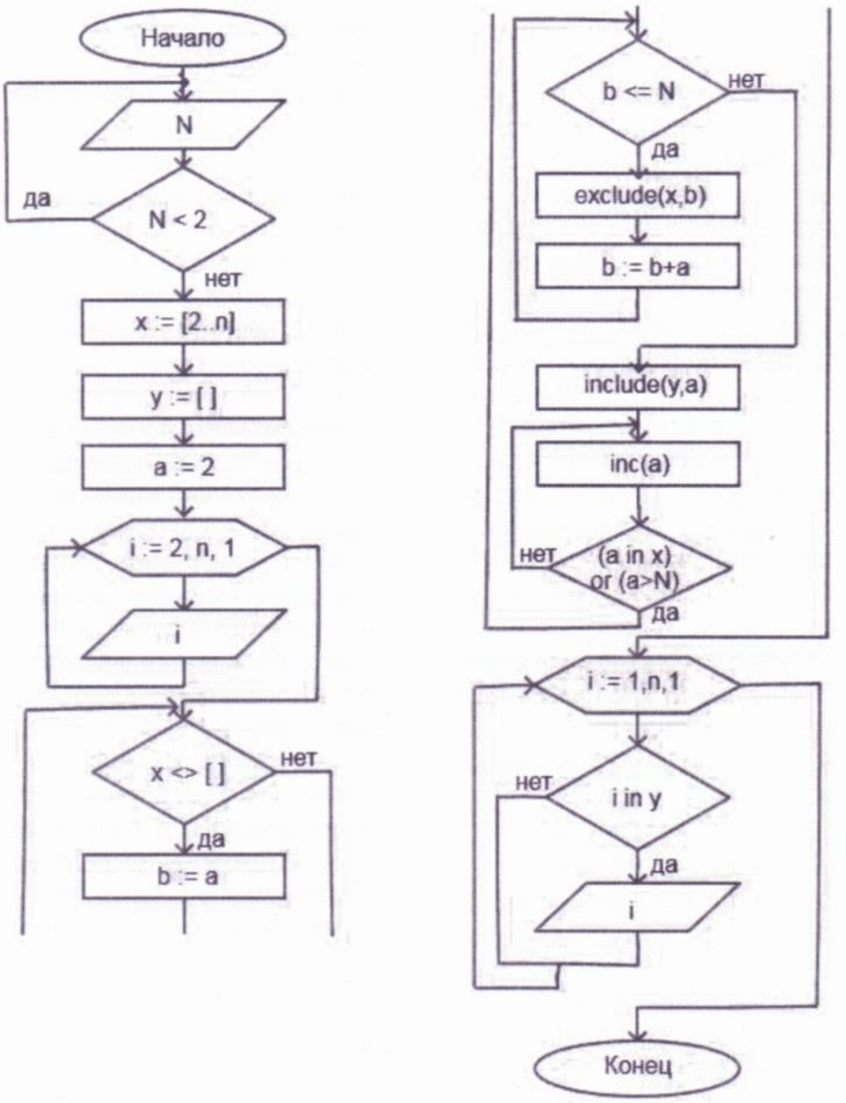

Рис. 1. Блок-схема алгоритма «Решето Эратосфена»

Далее приведен листинг программы на языке Pascal.

```
type
mn=set of 1..999;                 // Множество целых чисел до 999
var n: integer;
a,b,i: word;
x,y: mn;
begin
repeat                            // Ввод значения N>=2 с проверкой
```

```
writeln('Введите N');
readln(n);
until (n>=2);
x:=[2..n];    y:=[];  a:=2;               // X - множество всех целых чисел до N
                                          // Y - множестов простых чисел до N
writeln('Натуральные числа от 2 до N');
for i:=2 to n do                          // Выписываем все целые числа до N
write(i,' ');
writeln();
while (x<>[]) do begin                 // Пока в X остаются числа, выполняем алгоритм
b:=a;
while (b<=n) do begin                   // Вычеркиваем составные числа из X
exclude(x,b);                        // согласно алгоритму
b:=b+a;
end;
include(y,a);                             // Добавляем простое число в Y
repeat
inc(a);
until (a in x) or (a>n);
end;
writeln('Простые числа до N:');
for i:=1 to n do                            // Выводим простые числа до N
if (i in y) then write(i,' ');
end.
```

Пример 2. Решить систему нелинейных уравнений с заданной точностью:

$$\begin{cases} x - y = 3, \\ x^2 + y^2 = 25 - 2 \cdot x \cdot y. \end{cases}$$

Алгоритм решения данной системы разработан с помощью метода простых итераций, который заключается в следующем: пусть задано уравнение f(x) = 0.

Заменим его на уравнение $x = \varphi(x)$, которое получается из данного путем эквивалентных преобразований. Это уравнение определяется на некотором множестве E. Если значение f(x) тоже принадлежит E, то можно построить итерационную последовательность значений функции $\varphi(x)$ с начальным значением $x_0 \in E$.

Если эта последовательность сходится, то ее предел является единственным корнем уравнения $x = \varphi(x)$. Таким образом, исходное уравнение будет решено. В итоге $x = x_0 - f(x)/M$, где M – максимальное значение $f'(x)$ на [a;b]. $x_0 = (a+b)/2$. Процесс продолжается до тех пор, пока значение $|x_0 - x|$ остается бо́льшим точности или не достигнуто максимальное количество операций.

Реализуем описанный алгоритм в среде Delphi на языке Object Pascal. Интерфейс программы приведен на рисунке 2. Для определения приближенных значений корней (начальных приближений) используем инструмент Chart. Точность решения можно задать отдельно в виде десятичной дроби. В результате получаем ответ в виде пары (x;y), а также количество итераций, необходимых для решения.

Рис. 2. Интерфейс программы для решения системы нелинейных уравнений

Листинг основных процедур приведен ниже.

```
// Задаем уравнения системы
function TForm1.Functions(N: byte; X_F, Y_F: real): real;
begin
if N=1 then Functions:=3+Y_F;
if N=2 then Functions:=(X_F*X_F+Y_F*Y_F-25)/(-2*X_F);
end;
// Основная процедура, в которой реализован описанный алгоритм
procedure TForm1.BitBtn1Click(Sender: TObject);
var i:integer;
x,y: real;
begin
New_data;                  // Доп. процедура (считываем входные параметры)
i:=1;
x:=Functions(1,x0,y0);
y:=Functions(2,x,y0);
repeat                     // Цикл продолжается, пока не будет достигнута за-
данная
i:=i+1;               // точность или максимальное количество итераций
x0:=x;
y0:=y;
x:=Functions(1,x0,y0);
y:=Functions(2,x,y0);
until (((abs(x-x0)<eps) and (abs(y-y0)<eps)) or (i>10000));
if i>10000 then Application.MessageBox ('Достигнуто максимальное количество ите-
раций!', 'Внимание!');
Edit3.Text:=FloatToStr(x);          // Вывод результатов
Edit4.Text:=FloatToStr(y);
Edit5.Text:=IntToStr(i);
end;
```

Пример 3. Вычислить значение площади фигуры, ограниченной осью OX и графиком функции $f(x) = 6x - x^2$.

Для решения поставленной задачи

необходимо воспользоваться геометрическим смыслом определенного интеграла. Для нахождения значения интеграла рассмотрим некоторые методы численного интегрирования, основная идея которых состоит в замене подынтегральной функции на более простую. При этом для оценки значения интеграла получаются формулы вида $I \approx \sum_{i=1}^{n} w_i f(x_i)$, где n – число точек, в которых вычисляется значение подынтегральной функции, точки x_i – узловые точки, числа w_i – веса узлов. При замене подынтегральной функции на полином нулевой, первой и второй степени получаются соответственно методы прямоугольников, трапеций и парабол (Симпсона). Рассмотрим алгоритм нахождения значения интеграла таблично заданной функции на некотором отрезке [a; b].

Метод прямоугольников

Разделим отрезок [a;b] точками $x_0, x_1, ..., x_n$ на n равных отрезков длиной $\Delta x = \frac{b-a}{n}$; $y_0, y_1, ..., y_n$ – значение функции $f(x)$ в точках $x_0, x_1, ..., x_n$.

Составим интегральные суммы $y_0 \Delta x + y_1 \Delta x + ... + y_{n-1} \Delta x$.

Если $f(x)$ – положительная, возрастающая функция, то данная сумма приближенно равна значению интеграла $\int_a^b f(x)dx \approx \frac{b-a}{n}(y_0 + y_1 + ... + y_{n-1})$ и выражает площадь ступенчатой фигуры, составленной из «входящих» прямоугольников. При этом формула $\int_a^b f(x)dx \approx \frac{b-a}{n}(y_1 + y_2 + ... + y_n)$ выражает площадь ступенчатой фигуры, составленной из «выходящих» прямоугольников.

Чем меньше длина отрезков, на которые делится отрезок [a;b], тем точнее значение, вычисляемое по этой формуле, искомого интеграла.

Более точное значение интеграла можно получить, если взять в качестве опорной точки для нахождения высоты, точку посередине промежутка:

$$\int_a^b f(x)dx \approx \frac{b-a}{n} f\left(x_{i-1} + \frac{h}{2}\right).$$

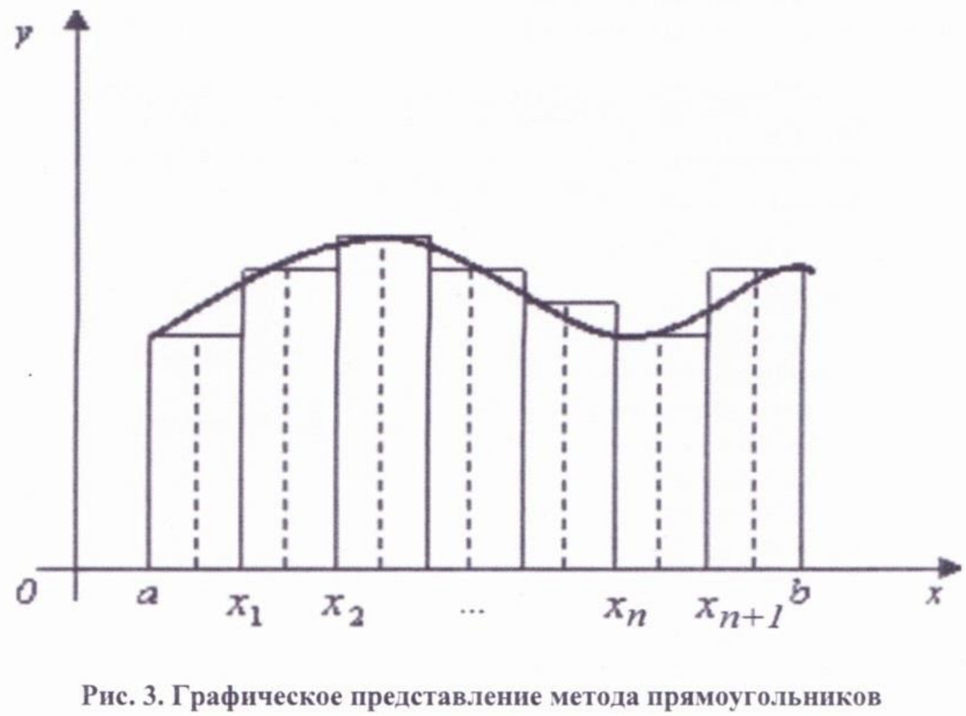

Рис. 3. Графическое представление метода прямоугольников

Метод трапеций

Аналогично предыдущему методу разделим отрезок [a;b] точками $x_0, x_1, \ldots, x_n$ на n произвольных отрезков.

На каждом элементарном отрезке заменим (аппроксимируем) подынтегральную функцию на линейную функцию. Площадь под графиком функции аппроксимируется прямоугольными трапециями.

Сумма площадей полученных трапеций даст приближенное значение интеграла на отрезке [a;b]: $\int_a^b f(x)dx \approx \sum_{i=0}^{n-1} \frac{f(x_i)+f(x_{i+1})}{2}(x_{i+1}-x_i)$.

Метод Симпсона

Если для аппроксимации использовать многочлен второй степени, то на каждом из участков отрезка [a; b] функция заменится на фрагмент параболы. Для аппроксимации можно использовать три точки – концы и середину отрезка. Приближенное значение интеграла выражается формулой:

$$\int_a^b f(x)dx \approx \frac{b-a}{6}\left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right).$$

Кроме методов численного интегрирования площадь рассматриваемой фигуры можно найти с помощью вероятностного метода Монте-Карло, который заключается в следующем:

1) «поместим» целиком полученную фигуру в некоторую прямоугольную область;

2) задавая случайным образом координаты точек, будем помещать их в пределах этой области;

3) отношение числа точек, попавших внутрь фигуры, к общему числу точек примерно равно отношению площади фигуры к площади прямоугольника.

Проведя компьютерный эксперимент с использованием выше перечисленных методов, можно оценить их точность, а также проследить зависимость получаемых результатов от количества отрезков, на которые разбивается интервал (или от количества точек).

Описанные методы реализованы в среде Delphi на языке Object Pascal. Интерфейс программы представлен на рисунке 4. Программа позволяет варьировать некоторые параметры эксперимента, а также визуально проследить за его ходом. Ниже также представлены фрагменты основных процедур программы.

Рис. 4. Интерфейс программы для нахождения значений площади фигуры с помощью интегрирования функции

function f(x:real):real; // *Задаем функцию, график которой ограничивает фигуру*

```
begin
 f:=6*x-sqr(x);
end;
procedure TForm1.BitBtn1Click(Sender: TObject);
var a,b,s,h,Ip,Ip1,Ip2,It:real;
n,n_point,i,x,y,x_point,y_point,k_point,koeff:integer;
t,k:extended;
polygon1: array [1..7] of TPoint;
begin

//Метод прямоугольников
for i:=0 to n-1 do
s:=s+f(a+i*h);
Ip1:=s*h;
edit4.Text:=floattostr(Ip1);
s:=0;    Ip:=0;
for i:=1 to n do
s:=s+f(a+i*h);
Ip2:=s*h;
edit5.Text:=floattostr(Ip2);
Ip:=(Ip1+Ip2)/2;
edit6.Text:=floattostr(Ip);
 //Метод трапеций
S:=(f(a)+f(b))/2;
for i:=1 to n-1 do
s:=s+f(a+i*h);
It:=s*h;
edit7.Text:=FloatToStr(It);

//Метод Симпсона
Ip:=0;    h:=0.5*(b-a)/n;    Ip:=f(a);    t:=a+h;
for i:=1 to 2*n-1 do begin
if i mod 2=0 then k:=2 else k:=4;
Ip:=Ip+k*f(t);     t:=t+h;
end;
Ip:=Ip+f(b);    Ip:=h*Ip/3;
edit8.Text:=FloatToStr(Ip);
//Метод Монте-Карло
koeff:=26;
n_point:=strtoint(edit10.text);
k_point:=0;
randomize;
for i:=1 to n_point do begin
x_point:=random(Image1.Width-7);
y_point:=random(Image1.Height-9)+9;
Image1.Canvas.Pixels[x_point,y_point]:=clBlue;
if (-6*x_point+sqr(x_point))/35 <= y_point then inc(k_point);
end;
```

```
edit9.Text:=floattostrf((Image1.Height-9)*(Image1.Width-
7)*k_point/(sqr(koeff)*n_point), ffFixed,6,4);
end;
```

Пример 4. Завод выпускает четыре вида продукции. Затраты на выпуск каждого вида приведены в таблице 1.

Таблица 1

Пример 4 – затраты на производство

Вид продукции	**Затраты**		
	Металл, кг	**Время работ станков, ч**	**Затраты электроэнергии, квт/ч**
1 вид	1	1	1
2 вид	2	3	2
3 вид	1	1	4
4 вид	2	3	4

На заводе существуют ограничения на использование ресурсов – не более 50 кг металла, 70 ч работы станков и 100 квт/ч электроэнергии в сутки. Необходимо определить, какое количество продукции каждого вида необходимо выпускать заводу для получения максимальной прибыли.

Прибыль с одной единицы продукции каждого вида приведена в таблице 2.

Таблица 2

Пример 4 – прибыль

	Вид продукции			
	1 вид	2 вид	3 вид	4 вид
Прибыль, руб.	2	7	8	5

При условии, что завод может исчерпать все ресурсы, составим соответствующую систему (x_i - количество продукции каждого вида):

$$\begin{cases} x_1 + 2x_2 + x_3 + 2x_4 = 50, \\ x_1 + 3x_2 + x_3 + 3x_4 = 70, \\ x_1 + 2x_2 + 4x_3 + 4x_4 = 100. \end{cases}$$

Зададим функцию, отражающую получаемую прибыль:

$$P = 2x_1 + 7x_2 + 8x_3 + 5x_4 .$$

Также при составлении модели рассматриваемого процесса необходимо учесть, что количество продукции не может быть отрицательной величиной.

Для нахождения оптимального плана выпуска продукции воспользуемся системой автоматизированного проектирования MathCAD Prime. Решение произведем с помощью решающего блока. В качестве «решателя» MathCAD позволяет использовать несколько видов функций. Т.к. в задаче требуется найти максимальное значение получаемой прибыли, то воспользуемся функцией Maximize. Затем, подставив полученные значения в функцию, получим значение прибыли.

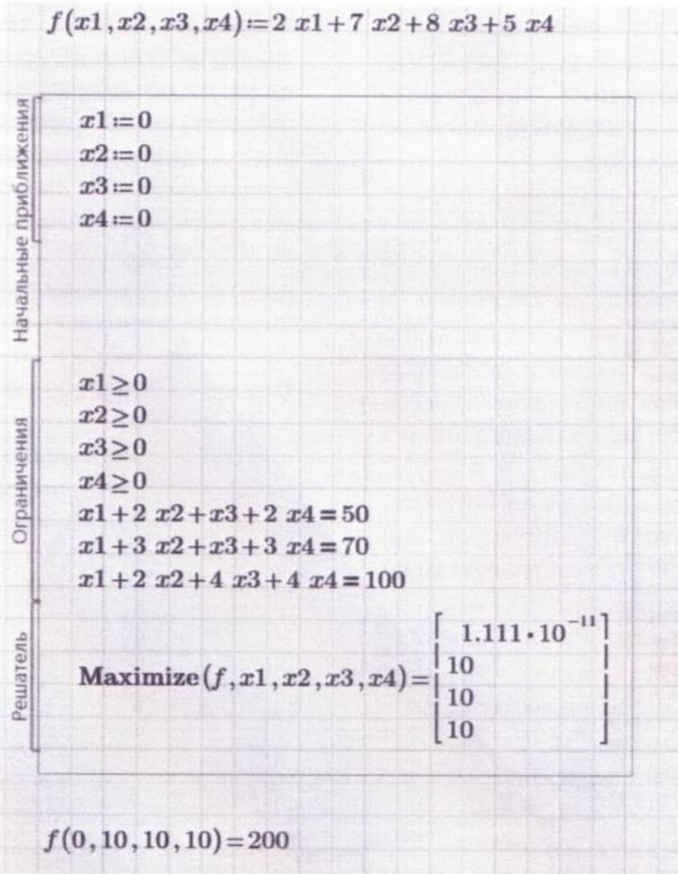

Рис. 5. Решение оптимизационной задачи в MathCAD Prime

Пример 5. Найти площадь поверхности тетраэдра, описанного около шара радиусом R.

Для построения модели и решения задачи необходимо воспользоваться некоторыми геометрическими свойствами:

– т.к. тетраэдр является правильным, то все его грани – равносторонние треугольники;

– радиус шара, вписанного в правильный тетраэдр, может быть найден по формуле $R = \dfrac{a \cdot \sqrt{6}}{12}$, где a – сторона тетраэдра;

– площадь поверхности тетраэдра может быть найдена как сумма площадей его граней: $S = \dfrac{4 \cdot a^2 \cdot \sqrt{3}}{4}$

Построение компьютерной модели проведем в геометрической среде GeoGebra:

1) в режиме *3D Graphics* построим правильный тетраэдр с помощью соответствующего инструмента;

2) определим середину ребра AC (точка E) и проведем отрезки DE и BE;

3) опустим высоту DF из вершины тетраэдра на отрезок BE;

4) проведем биссектрису угла DEB;

5) отметим точку O – точка пересечения биссектрисы и высоты DF;

6) построим шар с центром в точке O и радиусом OF;

7) в режиме *Геометрия* проведем построение сечения тетраэдра плоскостью BDE;

8) построим соответствующий треугольник и опустим в нем высоту DF;

9) аналогичным образом

определим точку O и построим окружность с центром в этой точке и радиусом OF;

10) с помощью инструмента *Ползунок* зададим конкретное значение радиуса шара;

11) зададим формулы для вычисления стороны тетраэдра, площади грани и площади его поверхности, используя значения ползунка;

12) выведем результаты с помощью инструмента *Надпись*.

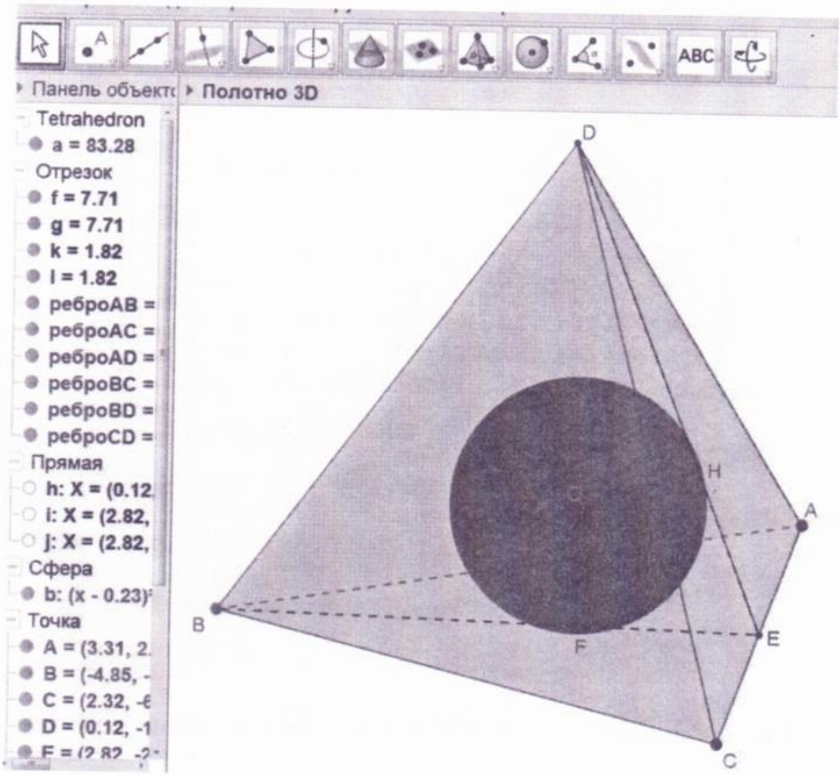

Рис. 6. 3D модель правильного тетраэдра, описанного вокруг шара

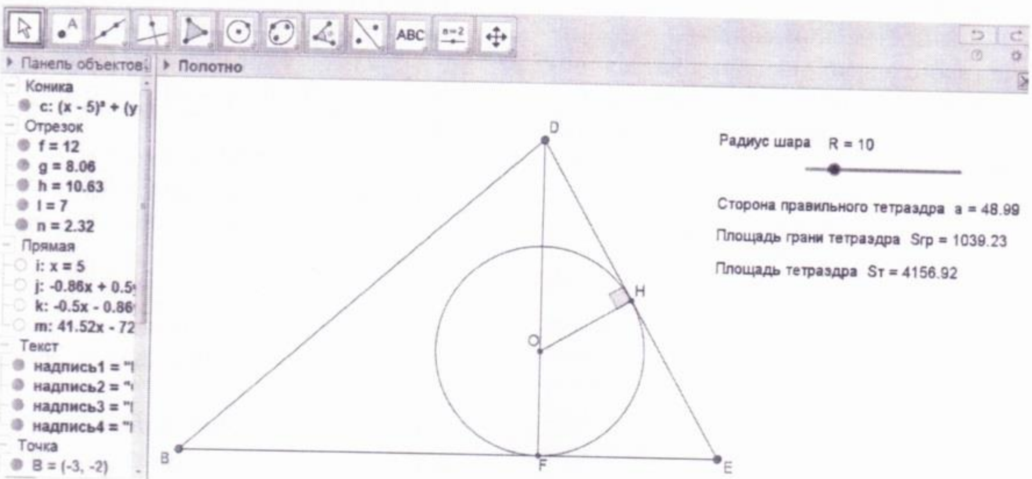

Рис. 7. Решение геометрической задачи в среде GeoGebra

Заключение

Полученные знания, умения и навыки при решении математических задач с помощью программирования и компьютерного моделирования позволяют установить межпредметные связи между предметами естественно-математического цикла, повысить качество выполняемых работ по ГИА и ЕГЭ, применяются при написании исследовательских работ, при решении олимпиадных задач.

В результате обучения учащиеся должны:

1) *знать* основные методы построения компьютерных моделей для:

– решения рациональных, иррациональных, показательных, логарифмических, тригонометрических уравнений, неравенств и их систем;

– построения и преобразования графиков функций;

– решения текстовых, стереометрических и комбинаторных задач;

– решения задач с параметрами;

– понятия производной, первообразной и их применения;

– решения задач планиметрии и стереометрии;

– метода координат и его применения к решению задач;

2) *уметь* использовать изученные среды программирования, системы автоматизированного проектирования для:

– выполнения тождественных преобразований рациональных, логарифмических, тригонометрических и других выражений;

– построения графиков сложных функций и их анализа;

– решения задач, уравнений, неравенств, систем, предусмотренных школьной программой;

– применения аппарата математического анализа к решению задач;

– выполнения пространственных чертежей;

– решения задач комбинаторики, построения графиков функций;

– моделирования реальных ситуаций на языке алгебры, составления уравнения и неравенства по условию задачи, исследования полученных моделей с использованием аппарата алгебры;

– моделирования реальных ситуаций на языке геометрии, исследования построенных моделей с использованием геометрических понятий и теорем;

– анализа реальных числовых данных;

– осуществления практических расчетов по формулам;

– использования оценки и прикидки при практических расчетах;

– описания с помощью функций различных реальных зависимостей между величинами и интерпретацией их графиков;

– извлечения информации, представленной в таблицах, на диаграммах, графиках;

– решения прикладных задач, задач на наибольшее и наименьшее значение, на нахождение скорости и ускорения;

– применения полученных знаний и умений на практике;

– нахождения по возможности оптимальных и рациональных способов решения задач.

Литература

1. Концепция развития российского математического образования (утверждена 24.12.2013 распоряжением Правительства РФ №2506-р) / [Электронный ресурс]. – Режим доступа: http://минобрнауки.рф/documents/3894.

2. Баранова Е.В. Объектно-ориентированное проектирование при обучении современным информационным технологиям / Е.В. Баранова. – С-Пб.: РГПУ им. А.И. Герцена, 2000. – 237 с.

3. Беспалько В.П. Программированное обучение. Дидактические основы / В.П.

Беспалько. – М.: Высшая школа, 1970. – 300 с.

4. Извозчиков В.А. Межпредметные связи и информатика / В.А. Извозчиков, А.М. Слуцкий. – СПб.: 1992. –51 с.

5. Карманов В.Г. Математическое программирование / В.Г. Карманов. – М.: ФизматЛит, 2011. – 264 с.

6. Сулейманов Р.Р. Компьютерное моделирование математических задач. Элективный курс / Р.Р. Сулейманов. – М.: БИНОМ. Лаборатория знаний, 2012. – 381 с.

7. Тишин В.И. Информатика и математика. Решение уравнений / В.И. Тишин. – М.: БИНОМ. Лаборатория знаний, 2013. – 112 с.