

Разработка методических материалов в среде системы Mathematica

Сосновский Н.Н.

Важным элементом процесса преподавания математики является разработка методических материалов для студентов. Автор исходит из того, что процесс освоения математики должен быть активным.

Студент должен выполнить обязательный минимум упражнений для того, чтобы достигнуть определенного уровня понимания материала и закрепить его в навыках.

Предлагаемые упражнения бывают двух типов.

Первый тип упражнений - это достаточно простые упражнения (задачи), цель которых проверить понимание излагаемого материала и помочь студентам приобрести необходимые навыки решения типовых задач.

Второй тип упражнений – это упражнения, требующие искусства для их выполнения. Такие упражнения предназначены для углубленного понимания математики и могут быть осилены не всеми студентами. Разработка таких упражнений не может быть осуществлена стандартными способами и требует творческого подхода от преподавателя.

Что же касается упражнений первого типа, то, напротив, разработка их может и должна быть автоматизирована, так как здесь требуется приготовить большое количество задач одного типа, чтобы обеспечить выдачу каждому студенту индивидуального задания, по результатам проверки которого можно сделать вывод о степени усвоения материала.

Так как это индивидуальные практические задания, то их много, и, следовательно, необходимо обеспечить наличие ответов для упрощения проверки и оценки результатов.

Сказанное говорит о том, что преподавателю необходимо иметь своего рода генератор задач для обеспечения формирования индивидуальных практических заданий по излагаемому материалу.

Автор имеет опыт разработки таких генераторов для различных задач: по алгебре, дискретной математике, теории вероятностей и математической статистике, - в среде системы Математика. В данном докладе рассматривается генератор задач дискретной математики.

Задания и ответы выводятся в отдельные файлы, конвертируются в PDF, и файлы с заданиями доводятся до студентов по электронной почте. Генератор обеспечивает формирование индивидуального задания каждому студенту группы. Студент выбирает вариант по своему номеру в списке группы, который я также довожу до студентов по электронной почте.

Все задачи задания подробно разбираются на практических занятиях.

Кроме того, я посылаю студентам методический материал, в котором на конкретных примерах показывается решение задач.

Сейчас я хочу показать на конкретных задачах, как устроен генератор и почему он обеспечивает необходимое разнообразие.

На рис.1 приводится пример вариантов заданий по дискретной математике, формируемых таким генератором.

Показаны два варианта задач и ответов к ним.

Варианты задач по дискретной математике

<p>Вар.№01 (3111)</p> <p>1) Решить диофантово уравнение $4876x + 1909y = -115$.</p> <p>Ответ записать в виде: $x=x_0+k \cdot t, y=y_0+h \cdot t$; x_0-наименьшее положительное число.</p> <p>2) Представить $\sqrt{140}$ в виде периодической цепной дроби.</p> <p>3) Найти представление рационального числа $\frac{734}{327}$ непрерывной дробью.</p> <p>4) Вычислить $\frac{29}{26}$ в кольце вычетов по модулю 57.</p> <p>5) Решить уравнение, записанное в 5-ичной системе счисления: $1200 + 24x = 4021$. Решение записать в десятичной и 5 - ичной системах.</p> <p>6) Найти наименьшее натуральное число, удовлетворяющее условиям $x \equiv 4 \pmod{23}, x \equiv 3 \pmod{17}, x \equiv 7 \pmod{41}$.</p> <p>7) Найти остаток от деления $77^{13^{74}}$ на 90.</p>	<p>Вар.№02 (3111)</p> <p>1) Решить диофантово уравнение $1749x + 1177y = 77$.</p> <p>Ответ записать в виде: $x=x_0+k \cdot t, y=y_0+h \cdot t$; x_0-наименьшее положительное число.</p> <p>2) Представить $\sqrt{176}$ в виде периодической цепной дроби.</p> <p>3) Найти представление рационального числа $\frac{965}{434}$ непрерывной дробью.</p> <p>4) Вычислить $\frac{45}{11}$ в кольце вычетов по модулю 94.</p> <p>5) Решить уравнение, записанное в 8-ичной системе счисления: $371 + 16x = 1073$. Решение записать в десятичной и 8 - ичной системах.</p> <p>6) Найти наименьшее натуральное число, удовлетворяющее условиям $x \equiv 22 \pmod{11}, x \equiv 26 \pmod{23}, x \equiv 11 \pmod{17}$.</p> <p>7) Найти остаток от деления $5^{17^{78}}$ на 86.</p>
<p>Вар.№01 (3111)</p> <p>1) $x = 45 - 83t$ $y = -115 + 212t$</p> <p>2) $\{11, \{1, 4, 1, 22\}\}$</p> <p>3) $\{2, 4, 11, 2, 3\}$</p> <p>4) 34</p> <p>5) $\{\{24, 10\}, \{44, 5\}\}$</p> <p>6) 2672</p> <p>7) 77</p>	<p>Вар.№02 (3111)</p> <p>1) $x = 31 - 107t$ $y = -46 + 159t$</p> <p>2) $\{13, \{3, 1, 3, 26\}\}$</p> <p>3) $\{2, 4, 2, 9, 5\}$</p> <p>4) 81</p> <p>5) $\{\{23, 10\}, \{27, 8\}\}$</p> <p>6) 946</p> <p>7) 5</p>

Рассмотрим генератор по частям.

Первая часть представлена на рис.2

```
(«Процедура формирования m порядковых целых чисел в текстовом формате»)
intstr[m_] := (sn = Table[0, {m}]; Do[(sn[[i]] = IntegerString[i] ; If[i < 10, sn[[i]] = StringJoin["0", sn[[i]]]],
    {i, 1, m}]);
intstr[100];
(«Процедура формирования строки из n пробелов»)
sp[n_] := (spn := ""; Do[spn = StringJoin[spn, " "], {i, 1, n}]);
sp[35];
doma1 = Table[Prime[n], {n, 13, 20}];
domb1 = Table[Prime[n], {n, 21, 30}];
domd = Table[Prime[n], {n, 5, 9}];
domQ = {119, 125, 128, 140, 155, 167, 173, 176, 185, 194, 215, 218, 248, 254, 266, 287, 293, 299, 305,
    320, 335, 338, 359, 365, 368, 377, 392, 395, 398};
domR = {

$$\frac{219}{76}, \frac{652}{329}, \frac{993}{338}, \frac{634}{433}, \frac{476}{169}, \frac{821}{328}, \frac{250}{137}, \frac{745}{318}, \frac{102}{53}, \frac{683}{324}, \frac{928}{455}, \frac{443}{193}, \frac{931}{370}, \frac{135}{74}, \frac{220}{83}, \frac{799}{412}, \frac{90}{53}, \frac{638}{415},$$


$$\frac{109}{53}, \frac{719}{324}, \frac{817}{328}, \frac{327}{226}, \frac{179}{73}, \frac{751}{306}, \frac{659}{309}, \frac{355}{218}, \frac{436}{171}, \frac{945}{326}, \frac{401}{192}, \frac{359}{216}, \frac{323}{154}, \frac{216}{133}, \frac{629}{302}, \frac{668}{411}, \frac{127}{73}, \frac{695}{332},$$


$$\frac{191}{87}, \frac{166}{67}, \frac{650}{347}, \frac{79}{29}, \frac{37}{17}, \frac{945}{317}, \frac{974}{451}, \frac{215}{134}, \frac{706}{309}, \frac{606}{337}, \frac{178}{77}, \frac{410}{187}, \frac{962}{441}, \frac{841}{314}, \frac{99}{43}, \frac{727}{332}, \frac{83}{32}, \frac{189}{74}, \frac{761}{313},$$


$$\frac{734}{327}, \frac{949}{334}, \frac{466}{209}, \frac{827}{399}, \frac{848}{433}, \frac{958}{411}, \frac{741}{332}, \frac{97}{57}, \frac{228}{83}, \frac{73}{45}, \frac{907}{317}, \frac{759}{359}, \frac{965}{434}, \frac{305}{106}, \frac{363}{218}, \frac{219}{131}, \frac{165}{73}}$$

};
```

Рис. 2

Сначала мы видим две процедуры: `intstr[m_]` и `sp[n_]`.

Первая процедура предназначена для формирования чисел определенной длины в текстовом формате. В данном случае это необходимо, чтобы получать номера билетов в виде текстовой строки длины 2 для чисел от 1 до 99, т.е. третий билет будет иметь номер 03.

Вторая процедура формирует строку пробелов заданной длины, в данном случае 35.

Далее идут операторы, создающие таблицы данных, используемые генератором в процессе работы. Так, `doma1`, `domb1` и `domd`, - это три непересекающихся диапазона простых чисел.

Таблица `domQ` содержит целые числа, разложения которых в цепную дробь имеют подходящую длину, в данном случае от 4 до 6.

В таблицу `domR` отобраны несократимые рациональные дроби, обладающие подобным свойством.

Отбор был сделан предварительно с помощью операторов цикла, использующих в качестве условия указанное ограничение на длину.

Заметим, в качестве примера, что формирование диапазонов чисел происходит с помощью команды `Table[Prime[n],{n,13,20}]`, которая в данном случае создает диапазон простых чисел с номерами от 13 до 20.

Следующая часть генератора представляет собой процедуру, процедура `przd[vr_,ngr_]`, параметрами которой являются номер варианта и номер группы. Процедура состоит из последовательности программ, генерирующих условия и ответы к задачам, входящим в билет.

Выполним последовательный обзор этих программ с тем, чтобы рассмотреть используемые операторы и оценить сложность программ.

Программа для первой задачи представлена на рис. 3.

Задача 1. Решить диофантово уравнение $4876x + 1909y = -115$

```
(*задача 1*)
a1 = RandomChoice[doma1];
b1 = RandomChoice[domb1];
d = RandomChoice[domd];
a = a1 d RandomInteger[{1, 5}];
b = b1 d RandomChoice[{1, -1}];
c = d RandomInteger[{3, 10}] RandomChoice[{1, -1}];
Clear[x, y];
usl1 = StringForm["`", a x + b y == c];
{{x0, y0}} = {x, y} /. FindInstance[{a x + b y == c, 0 ≤ x ≤ Abs[b/d]}, {x, y}, Integers];
ans1 = StringForm["x = `1`
y = `2`", x0 - b/d t, y0 + a/d t];
```

Рис. 3

Первые строки программы, это генерация с помощью операторов `RandomChoice[]` и `RandomInteger[]` коэффициентов уравнения. Затем, с помощью команд `FindInstance[]` и `StringForm[]` происходит решение уравнения и формирование текстов условия и ответа к задаче.

Примерный вид ожидаемого решения

$$4876x + 1909y = -115$$

r	4876	1909	1058	851	207	23	0
q			2	1	1	4	9
x	1	0	1	-1	2	-9	83
y	0	1	-2	3	-5	23	-212

$$4876*(-9)+1909*23= 23$$

$$-115/23= -5$$

$$4876*(45)+1909*(-115)= -115$$

$$4876*(83)+1909*(-212)= 0$$

$$x=45-83*t$$

$$y=-115+212*t$$

Задача 2. Представить $\sqrt{140}$ в виде периодической цепной дроби

```
(*задача 2*)
q = RandomChoice[domQ];
us12 = StringForm["√", q];
ans2 = ContinuedFraction[√q];
```

Рис. 4

Первый оператор это случайный выбор числа, RandomChoice[]. Вторым и третьим оператором формируют условие и ответ к задаче на разложение корня квадратного из целого числа в непрерывную дробь. Как видим, используются два оператора: StringForm [] и ContinuedFraction [].

Примерный вид решения

$$\sqrt{140} = 11 + \sqrt{140} - 11$$

$$\frac{1}{\sqrt{140} - 11} = \frac{\sqrt{140} + 11}{19} = 1 + \frac{\sqrt{140} - 8}{19}$$

$$\frac{19}{\sqrt{140} - 8} = \frac{\sqrt{140} + 8}{4} = 4 + \frac{\sqrt{140} - 8}{4}$$

$$\frac{4}{\sqrt{140} - 8} = \frac{\sqrt{140} + 8}{19} = 1 + \frac{\sqrt{140} - 11}{19}$$

$$\frac{19}{\sqrt{140} - 11} = \sqrt{140} + 11 = 22 + \sqrt{140} - 11$$

$$\sqrt{140} = (11, (1, 4, 1, 22))$$

$$11 + \sqrt{140} = (22, 1, 4, 1, 22)$$

Задача 3. Найти представление рационального числа $\frac{734}{327}$
непрерывной дробью

```
(*задача 3*)  
us13 = RandomChoice[domR];  
ans3 = ContinuedFraction[us13];
```

Рис. 5

Здесь осуществляется выбор из диапазона рациональных дробей и формирование ответа.

Примерный вид решения

r	734	327	80	7	3	1	0
q			2	4	11	2	3
x	1	0	1	-4	45	-94	327
y	0	1	-2	9	-101	211	-734

						734	327
						654	2
					327	80	
					320	4	
				80	7		
				77	11		
				7	3		
				6	2		
				3	1		
				3	3		
				0			

$$\frac{734}{327} = (2, 4, 11, 2, 3)$$

Задача 4. Вычислить $\frac{29}{26}$ в кольце вычетов по модулю 57

```
(*задача 4*)
m = RandomInteger[{51, 99}];
lm = EulerPhi[m];
t1m = Table[0, {lm}];
s1 = 0; Do[If[GCD[k1, m] == 1, (s1 = s1 + 1; t1m[[s1]] = k1)], {k1, 1, m - 1}];
k = RandomChoice[t1m];
If[k == 1, k = t1m[[2]]];
sol = FindInstance[k x + m y == 1 && 0 < x < m, {x, y}, Integers];
{a} = x /. sol;
b = RandomChoice[t1m];
While[GCD[b, k] != 1 || Mod[b/k, 1] == 0, b = RandomChoice[t1m]];
us14 = b/k;
ans4 = Mod[a b, m];
m4 = m;
```

Рис. 6

Алгоритм к задаче 4 следующий.

Случайно выбираем целое число m , у которого достаточно большая система приведенных вычетов.

Далее формируем эту таблицу приведенных вычетов tlm и выбираем из нее случайное число k , не равное 1.

Затем решаем сравнение $ax \equiv 1 \pmod{m}$. Запоминаем решение, число a .

Далее выбираем взаимно простое с k число b , для которого b/k не является целым.

В условии пишем рациональную дробь b/k , в ответ пишем остаток от деления числа ab на m .

Ожидаемое решение

$$\frac{29}{26} \equiv x(57) \quad 26x + 57y = 29$$

r	26	57	26	5	1	0
q			0	2	5	5
x	1	0	1	-2	11	-57
y	0	1	0	1	-5	26

$$26 \cdot 11 + 57 \cdot (-5) = 1$$

$$26 \cdot (11 \cdot 29) + 57 \cdot (-5 \cdot 29) \equiv 29$$

$$\text{ОСТАТ}(11 \cdot 29; 57) = 34$$

Задача 5. Решить уравнение записанное в 5 –ой системе счисления:
 $1200 + 24x = 4021$. Решение записать в десятичной и 5 – ой системах.

```
(*задача 5*)  
a = RandomInteger[{5, 19}];  
b = RandomInteger[{101, 299}];  
u = RandomInteger[{11, 29}];  
c = a u + b;  
q = RandomChoice[{2, 3, 4, 5, 6, 7, 8, 9}];  
aq = IntegerString[a, q];  
bq = IntegerString[b, q];  
uq = IntegerString[u, q];  
cq = IntegerString[c, q];  
us15 = StringForm["`", aq x + bq == cq];  
ans5 = {{u, 10}, {uq, q}};
```

Рис. 7

Алгоритм.

Выбираем случайно целые числа a, b, u и вычисляем число c : $c = au + b$.
Наугад выбираем основание системы счисления q из списка $\{2, 3, 4, 5, 6, 7, 8, 9\}$.
Переводим затем числа a, b, u и c в q - ичную систему счисления и формируем условие и ответ к задаче.

Задача 6. Найти наименьшее натуральное число, удовлетворяющее условиям: $x = 4 \pmod{23}$, $x = 3 \pmod{17}$, $x = 7 \pmod{41}$

```
(*задача 6*)
c = RandomSample[Range[0, 40], 3];
domr = {11, 13, 17, 19, 23, 29, 31, 37, 41, 43};
domr1 = {10, 14, 15, 21, 25, 35, 49};
e = RandomChoice[domr1];
AppendTo[domr, e];
m = RandomSample[domr, 3];
usl6 = StringForm["x ≡ `` mod ``", x ≡ `` mod ``", x ≡ `` mod ``", c[[1]], m[[1]], c[[2]], m[[2]],
  c[[3]], m[[3]]];
ans6 = ChineseRemainder[c, m];
```

Рис. 8

Алгоритм

Выбираем 3 целых числа без повторения из диапазона $\{0,1,\dots,40\}$.

Формируем диапазон для выбора трех попарно простых чисел – модулей сравнений m_1, m_2, m_3 . . Делаем случайную выборку без повторения 3 чисел.

Формируем условие и ответ к задаче.

Задача 7. Найти остаток от деления $77^{13^{74}}$ на 90.

```
(*задача 7*)|
m = RandomInteger[{31, 99}];
lm = EulerPhi[m];
t1m = Table[0, {k, 1, lm}];
s1 = 0; Do[If[GCD[k1, m] == 1, (s1 = s1 + 1; t1m[[s1]] = k1)], {k1, 1, m - 1}];
na = RandomInteger[{2, lm}];
a = t1m[[na]];
lm1 = EulerPhi[lm];
t1m1 = Table[0, {k, 1, lm1}];
s1 = 0; Do[If[GCD[k1, lm] == 1, (s1 = s1 + 1; t1m1[[s1]] = k1)], {k1, 1, lm - 1}];
nb = RandomInteger[{2, lm1}];
b = t1m1[[nb]];
tlmc = Table[If[GCD[k, a] == 1 && GCD[k, b] == 1, k, 0], {k, 1, 125}];
tlmc = Select[tlmc, # > 0 &];
lc = Length[tlmc];
nc = RandomInteger[{2, lc}];
c = tlmc[[nc]];
b1 = PowerMod[b, c, lm];
xm = PowerMod[a, b1, m];

us17 = StringForm["`1`^`2`^`3`", a, b, c];
ans7 = xm;
```

Рис. 9

Алгоритм.

Выбираем случайное целое число m из диапазона $\{39, \dots, 99\}$.

Строим приведенную систему вычетов tlm для этого числа.

Выбираем случайное целое число a из tlm , не равное 1.

Строим приведенную систему вычетов $tlm1$ для этого числа.

Выбираем случайное целое число b из $tlm1$, не равное 1.

Строим таблицу $tlmc$ чисел, попарно простых с a и b .

С помощью операторов $PowerMod[b,c,lm]$ и $PowerMod[a,b1,m]$ вычисляем значение искомой степени.

Формируем условие и ответ к задаче.

В данной задаче ответ 77, т.е. $77^{13^{74}} \equiv 77 \pmod{90}$

Но для задачи `Найти остаток от деления $35^{19^{74}}$ на 66.` Ответ: 17

Формат текста условия

(*формирование билета и ответа к нему*)

```
us1 = StringForm[
  "Var. №`1``2``3`
1) Решить диофантово уравнение
`4`.
Ответ записать в виде:  $x=x_0+k \cdot t, y=y_0+h \cdot t$ ;
 $x_0$  - наименьшее положительное число.
2) Представить `5` в виде периодической цепной
доби.|
3) Найти представление рационального числа `6`
непрерывной дробью.
4) Вычислить `7` в кольце вычетов по модулю `8`.
5) Решить уравнение, записанное в `9`-ичной системе
счисления: `10`.
Решение записать в десятичной и `9` - ичной
системах.
6) Найти наименьшее натуральное число,
удовлетворяющее условиям
`11`.
7) Найти остаток от деления `12` на `13`.", sn[[vr]], ngr, spn, us11, us12, us13, us14,
m4, q, us15, us16, us17, m];
```

Рис. 10

Формат текста ответов

```
ans = StringForm[
  "Вар.№`1``2``3`
  1) `4`
  2) `5`
  3) `6`
  4) `7`
  5) `8`
  6) `9`
  7) `10`" , sn[[vr]], ngr, spn, ans1, ans2, ans3, ans4, ans5, ans6, ans7]
];
```

Рис. 11

Программа формирования файлов билетов и ответов к ним
prn[nvr_,ngr_], параметрами которой являются число вариантов и номер группы

```
prn[nvr_, ngr_] :=  
(If[Mod[nvr, 2] == 1, nv = nvr + 1, nv = nvr];  
cnd = Table[0, {str, 1, nv/2}, {nzd, 1, 2}, {tip, 1, 2}];  
Do[{k = Mod[2 s - 1, 99, 1]; przd[k, ngr];  
cnd[[s, 1, 1]] = usl; cnd[[s, 1, 2]] = ans;  
k = Mod[2 s, 99, 1]; przd[k, ngr];  
cnd[[s, 2, 1]] = usl; cnd[[s, 2, 2]] = ans;}, {s, 1, nv/2}];  
Do[Print[Grid[{{cnd[[s, 1, 1]], cnd[[s, 2, 1]]}}, Frame → All, Spacings → {1, 1}], {s, 1, nv/2}];  
Do[Print[Grid[{{cnd[[s, 1, 2]], cnd[[s, 2, 2]]}}, Frame → All, Spacings → {1, 1}], {s, 1, nv/2}];)
```

Программа оформлена в виде процедуры с двумя параметрами: число вариантов и номер группы.

Число вариантов округляется до ближайшего сверху целого числа.

Строится трехмерная таблица `spd`, первоначально пустая. Каждый элемент этой таблицы соответствует паре задач, печатаемых на листе рядом. Он состоит из двух элементов – для первой и второй задачи. Элемент каждой задачи в свою очередь делится на два элемента. В первый из этих элементов записывается текст условия, а во второй – текст ответа к задаче.

Заполнение таблицы `spd` происходит в цикле по числу строк файла, т.е. числу пар задач, печатаемых на листе рядом. Для каждой строки два раза запускается процедура `przd[vr_,ngr_]`, параметрами которой являются номер варианта и номер группы. Каждый вариант состоит из одних и тех же задач, но так как процедура каждой задачи включает генерацию случайных чисел, то задачи одного типа для разных вариантов различаются исходными данными.

Две последние строки программы это циклы формирования (печати) билетов и ответов к ним.