

Майер Р.В.
КИБЕРНЕТИЧЕСКОЕ МОДЕЛИРОВАНИЕ
ВЗАИМОДЕЙСТВИЯ МЕЖДУ УЧЕНИКОМ И УЧИТЕЛЕМ

Как сказал У. Р. Эшби, кибернетика – «наука о том, как надо управлять очень сложной системой, чтобы в итоге она вела себя желательным для нас образом». Основная задача кибернетической педагогики состоит в выявлении принципов и способов эффективного управления учебным процессом, при котором минимальные затраты времени (усилий, денег) позволяют достичь требуемого уровня знаний учащихся. Решение этой проблемы требует построения абстрактной кибернетической системы учебного процесса, состоящей из множества взаимосвязанных объектов, которые воспринимают, запоминают, перерабатывают и обмениваются информацией. Создание такой качественной модели позволяет осуществить математическое моделирование, а затем перейти к имитационной модели на ЭВМ.

1. Кибернетическая система учебного процесса

Построим кибернетическую систему учебного процесса. Она должна включать в себя абстрактные модели учителя, учеников и их родителей, способных воспринимать, запоминать, перерабатывать и обмениваться информацией. С целью упрощения рассуждений можно абстрагироваться от стохастического характера поведения перечисленных выше объектов и считать их детерминированными автоматами с большим числом внутренних состояний. Итак, учитель в простейшем случае моделируется автоматом, задаваемым двойкой $\langle P, A \rangle$, где P — программа курса, A — алгоритм работы. Программа курса характеризуется списком из N вопросов (тем), их сложностью S_i и временем их изучения t_i . Модель ученика задается четверкой $\langle \alpha, \gamma, U, Z \rangle$, где α — коэффициент научения, γ — коэффициент забывания ученика, U — уровень притязаний ученика из интервала $[0; 1]$, пропорциональный оценке, которую он стремится получить в конце обучения, $Z = \{Z_1, Z_2, \dots, Z_m\}$ — знания ученика. Будем считать, что Z_i — уровень знаний i -ой темы, который лежит в интервале $[0; 1]$ и равен вероятности правильного выполнения теста по данной теме. Модель родителя — воображаемый автомат, задаваемый двойкой $\langle W, V \rangle$, где W — уровень притязаний родителя, V — возможность родителя оказать психологическое воздействие на своего ребенка и повысить его мотивацию к обучению W .

В процессе обучения учитель воздействует на учеников, сообщая им учебную информацию и осуществляя текущий контроль (вопросы, тестирования). Учащиеся также воздействуют на учителя, сообщая, что им понятно или не понятно, задавая вопросы и выполняя задания текущего тестирования. Так возникает **первый замкнутый контур управления**. Учитель, видя реакцию учеников, может очень быстро на нее реагировать: отвечать на вопросы, обращать внимание учащихся на их ошибки, помогать им их исправлять.

В конце изучения темы учитель проводит контрольную работу, результаты которой также позволяют оценить уровень знаний учащихся и выбрать дальнейшую стратегию обучения: либо приступить к изучению новой темы, либо повторить изучение тех вопросов, которые были усвоены недостаточно хорошо. Это **второй замкнутый контур управления**. Он содержит элемент задержки, который как бы задерживает сигнал от учащегося на некоторое время τ_1 .

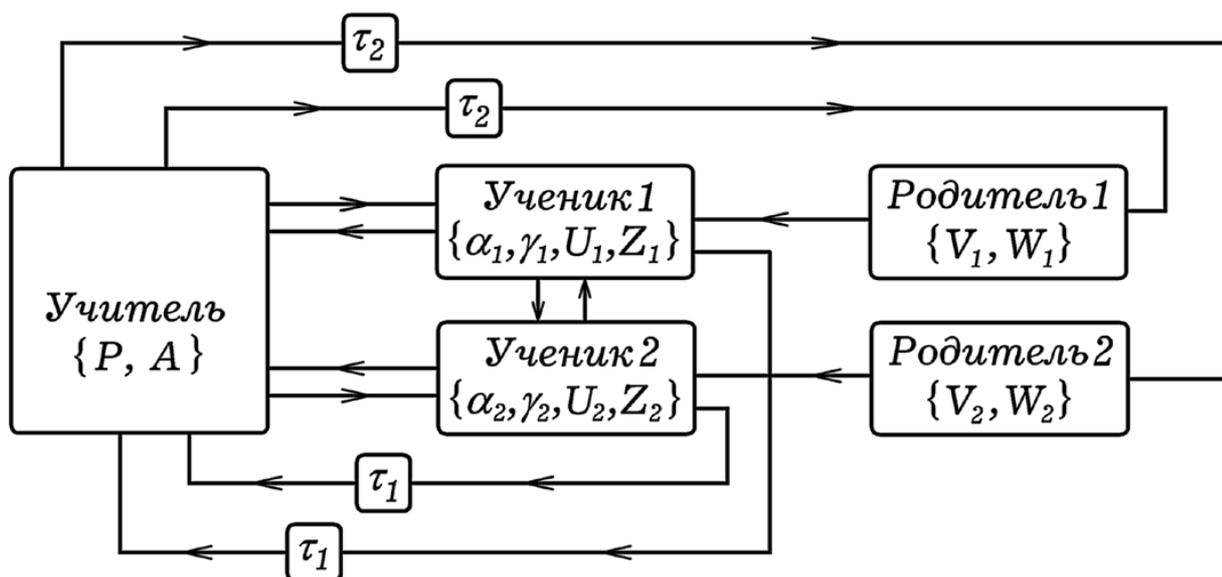


Рис. 1. Учебный процесс как кибернетическая система.

В случае, когда учитель видит, что учащийся плохо работает, он сообщает об этом родителям. Если успехи ребенка их не устраивают и тот имеет возможность воздействовать на ребенка, то он повышает мотивацию к обучению, увеличивая параметр U . Это **третий замкнутый контур управления**. Он также содержит элемент задержки на время τ_2 .

Можно усложнить систему, введя в нее новые элементы, например, директора школы, который контролирует работу учителя и результаты обучения,

сопоставляя их с требуемым уровнем. При этом получится четвертый замкнутый контур управления (на рис.1 он не изображен).

Надо понимать, что в ряде случаев кибернетическая система управления претерпевает изменения. Например, в роли учителя может выступать компьютер с обучающей программой, подключенный к какому-нибудь образовательному ресурсу. Роль родителей, повышающих мотивацию учащегося, может играть учитель, который проводит с ними воспитательную беседу, убеждает в необходимости более ответственного отношения к учебе и т.д.

2. Компьютерная модель процесса обучения

Допустим, обучаемый должен механически запомнить последовательность выполнения каких-либо действий, например, научиться считать от 0 до 9 на русском или иностранном языке, выучить алфавит, последовательность каких-то не связанных друг с другом слов, чисел и т.д. В принципе под эту ситуацию может попасть и случай, когда запоминание не механическое, элементы усваиваемой информации (операции, действия) как-то связаны между собой, но степень связи мы пренебрегаем, либо считаем, что в среднем она несколько повышает коэффициент усвоения информации.

Пусть учащийся пытается усвоить выполнения определенной последовательности операций $O_0 \rightarrow O_1 \rightarrow O_2 \rightarrow \dots \rightarrow O_N$, приводящей к решению некоторой учебной задачи. При этом процесс обучения состоит из двух этапов. На первом этапе обучаемый 5–10 раз выполняет последовательность операций $O_0 \rightarrow O_1 \rightarrow O_2 \rightarrow \dots \rightarrow O_N$ вместе с учителем (компьютером, учебником), например, вслух читает алфавит. Каждый раз, когда он совершает правильный переход от операции O_i к O_{i+1} учащийся учится с коэффициентом научения α_1 . Это означает, что сначала соответствующая вероятность $P_{i,i+1}$ увеличивается $\alpha_1(1 - P_{i,i+1})$, после чего производится нормирование: вероятности всех переходов $P_{i,j}$ пересчитываются так, чтобы их сумма была бы точно равна 1. Для нахождения нормированных вероятностей используется формула:

$$P'_{i,j} = \frac{P_{i,j}}{P_{i,0} + P_{i,1} + P_{i,2} + \dots + P_{i,9}}.$$

На втором этапе учащийся работает с учителем: ученик по памяти пытается воспроизвести запоминаемую последовательность операций, а учитель как-то реагирует на ответы учащегося, поощряет правильные, наказывает или исправляет неверные и т.д. На рис. 1 представлен алгоритм функционирования системы "учитель – учащийся". В случае правильного ответа учащегося учитель поощряет его (говорит "Да" или молчит), при этом школьник обучается с коэффициентом научения a_2 . В случае ошибочного действия $O_i \rightarrow O_k$, $k \neq i+1$ учитель выбирает одну из следующих четырех стратегий реагирования.

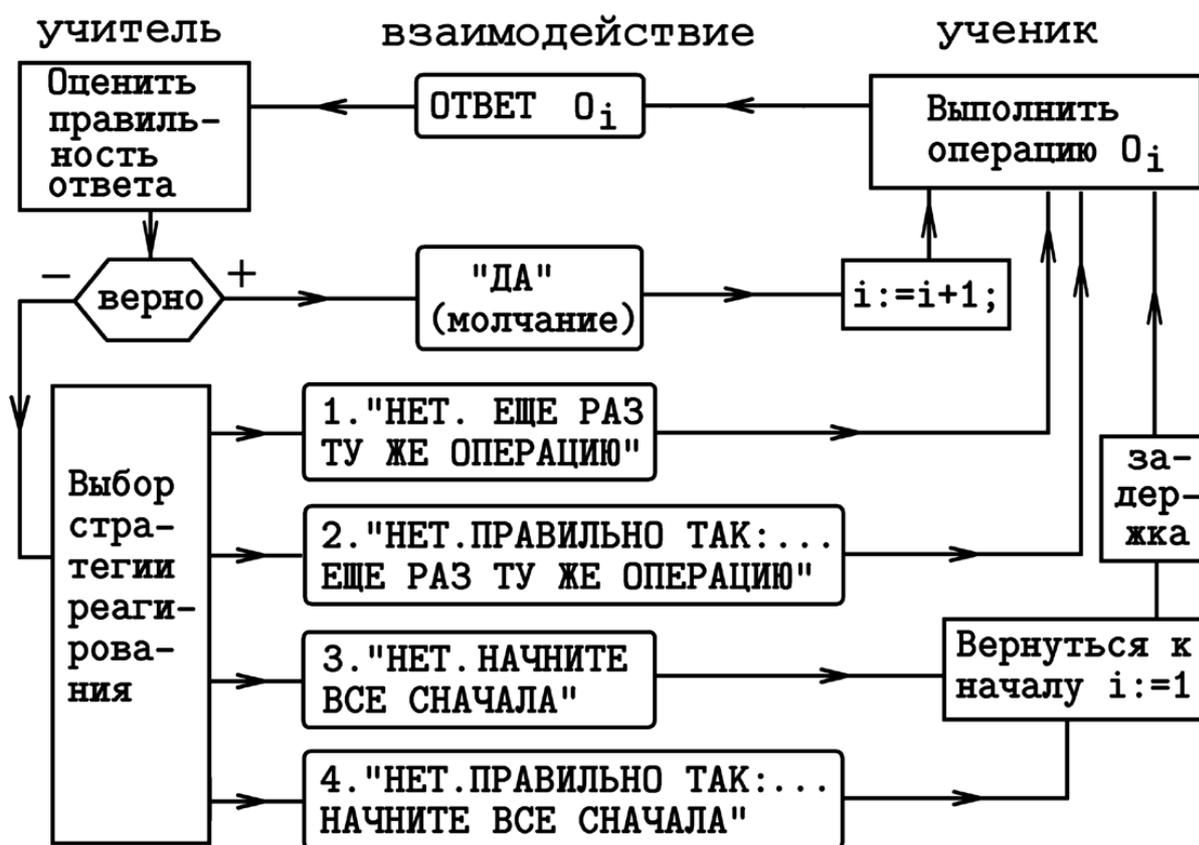


Рис. 2. Взаимодействие между учителем и учащимся.

Стратегия 1: "Неверно, повторите еще раз ту же операцию". При этом он "наказывает" учащегося с коэффициентом научения a_2 . Это значит, что вероятности неправильного перехода p_{ik} уменьшается на $a_3 p_{ik}$, а затем осуществляется нормирование всех вероятностей p_{ij} ($j = 1, 2, \dots, N$). После этого учащийся снова пытается выполнить операцию

Стратегия 2: "Неверно. Правильно так: O_{i+1} . Повторите еще раз ту же операцию". При этом увеличивается вероятность правильного перехода на $p_{i,i+1}$

на $a_3(1 - p_{i,i+1})$ и нормируются остальные вероятности p_{ij} ($j = 1, 2, \dots, N$). Учащийся продолжает решение задачи с операции O_i .

Стратегия 3: "Неверно. Повторите всю последовательность действий с начала (с операции O_1)". Учащегося наказывают с коэффициентом обучения a_3 . При этом вероятности неправильного перехода p_{ik} уменьшается на $a_3 p_{ik}$, после чего осуществляется нормирование всех вероятностей p_{ij} ($j = 0, 1, \dots, 9$). Затем учащийся начинает решать задачу с самого начала.

Стратегия 4: "Неверно. Правильно так: O_{i+1} . Повторите всю последовательность действий с начала (с операции O_1)". При этом увеличивается вероятность правильного перехода на $p_{i,i+1}$ на $a_3(1 - p_{i,i+1})$ и нормируются остальные вероятности p_{ij} ($j = 0, 1, \dots, 9$). Учащийся возвращается к началу задачи.

Важным вопросом является проблема оценки уровня научения учащегося. В качестве показателей успешности обучения выбраны: 1) уровень знаний (или сформированности навыка), равный среднему арифметическому вероятностей всех правильных переходов $p_{cp} = (p_{01} + p_{12} + p_{23} + \dots + p_{89})/9$. 2) вероятность правильного выполнения всей совокупности операций (решения задачи), равная произведению вероятностей правильных переходов: $P_{зад} = P_{01}P_{12}P_{23}P_{34} \dots P_{89}$.

Компьютерная программа ПР-1, моделирующая анализируемую ситуацию, приведена в приложении. Результаты моделирования представлены в таблице 1 и на рис. 2. В нашем случае всего было 10 операций O_i , им соответствовало 9 правильных переходов: $O_0 \rightarrow O_1 \rightarrow O_2 \rightarrow \dots \rightarrow O_9$. Было задано $\alpha_1 = \alpha_2 = \alpha_3 = 0,1$. Число повторов в предварительном обучении равно $k_1 = 5$. Каждый раз учитель показывает правильную последовательность операций, вероятности правильных переходов при этом возрастают. После предварительного обучения (1 этап) уровень знаний был $p_{cp} = 0,47$, $P_{зад} = 0,001$.

После этого моделировалось обучение путем проб и ошибок (2 этап). Выполнять операции учащийся начинает с операции O_0 , счетчик операций N_o увеличивается на 1. ПЭВМ выбирает случайное число x из интервала $[0; 1]$ и методом выбора по жребию разыгрывается следующий номер операции,

выбираемой учеником. Если ученик совершает правильный переход, то есть $O_i - O_{i-1} = 1$ (после O_4 выбрана O_5), то учитель хвалит учащегося, подтверждая правильность выбора. При этом вероятность правильно совершенного перехода увеличивается на $\Delta p = a_2(1 - p[o[i-1], o[i]])$, а затем нормируются вероятности p_{ij} ($j = 0, 1, \dots, 9$) так, чтобы их сумма была равна 1.

Если ученик совершил неправильный переход ($O_i - O_{i-1} \neq 1$), то учитель наказывает ученика. Если при этом он не подсказывает правильный выбор, то вероятность неверно совершенного перехода уменьшается на $\Delta p = a_3 p[o[i-1], o[i]]$, а вероятность всех остальных переходов возрастает на $\Delta p / (N - 1)$. В случае, когда учитель подсказывает правильный ответ, то используется другой алгоритм: вероятность неверно совершенного перехода уменьшается на Δp , а вероятность правильного перехода от O_{i-1} к $O_{i-1} + 1$ увеличивается на Δp . Сумма вероятностей всех переходов остается равной 1.

Таблица 1.

Стратегия учителя	Число ответов, $N_{от}$	Число ошибок, $N_{ош}$	Время обучения, t	$P_{ср}$	$P_{зад}$
Стратегия 1	521	286	807	0,84	0,20
Стратегия 2	649	157	806	0,97	0,78
Стратегия 3	698	105	804	0,93	0,50
Стратегия 4	727	78	805	0,96	0,70

Программа делает 200 циклов (испытаний) и каждый раз вычисляет общее число ответов N_1 , число ошибочных ответов N_2 , общее время обучения $t = (N_1 - N_2)\Delta t_1 + N_2\Delta t_2$, которое не должно превышать заданное значение $t_{\max} = 800$ условных единиц времени. Когда это происходит, программа выходит из цикла, заканчивается данное испытание, результаты выводятся на экран ПЭВМ. В нашем случае $\Delta t_1 = 1$ (усл. ед. вр.), $\Delta t_2 = 2$ усл. ед. вр., то есть на ошибочный ответ и его исправление затрачивается в 2 раза больше времени, чем на правильный ответ.

Из таблицы 1 видно, что наиболее эффективной является стратегия 2 (“Нет. Правильно так. Повторите еще раз ту же операцию”) и стратегия 4 (“Нет. Правильно так. Повторите все с начала”). Эти стратегии поведения учителя предполагают подсказку учащимся правильного выбора операции. При использовании учителем стратегии 4 учащийся дает максимальное количество

ответом при минимальном числе ошибок. Стратегия 1 (“Нет. Повторите еще раз ту же операцию”) является самой неэффективной.

Стратегии 3 и 4, предусматривающие возврат учащегося к началу выполнения всех действий, приводят к тому, что он чаще выполняет первые операции O_1, O_2, O_3, O_4 и реже последние O_6, O_7, O_8, O_9 . Поэтому после второго этапа обучения вероятности переходов $p_{01}, p_{12}, p_{23}, p_{34}$, достаточно высоки, в то время как вероятности $p_{56}, p_{67}, p_{78}, p_{89}$ малы, что приводит к низкой вероятности $p_{зад}$ решения всей задачи. Стратегии 1 и 2 не требуют возврата учащегося к началу задачи, — после ошибки он продолжает выполнять действия с того места, где он совершил ошибку. Поэтому вероятности правильных переходов после второго этапа обучения примерно одинаковы. Стратегия 2 эффективнее стратегии 1 потому, что при ее использовании учитель подсказывает правильный выбор операции.

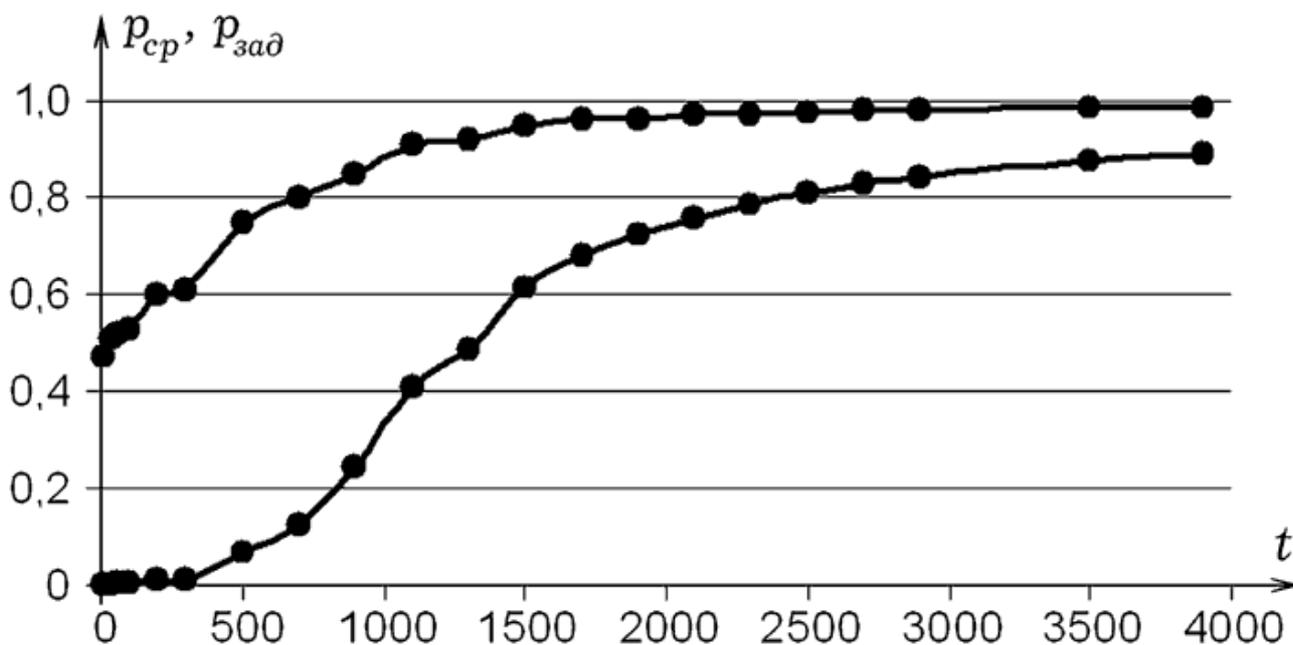


Рис. 3. Типичная зависимость $P_{ср}$ и $P_{зад}$ от времени обучения.

Для изучения зависимости уровня знаний $P_{ср}$ и вероятности $P_{зад}$ решения задачи от времени обучения методом проб и ошибок был проведен вычислительный эксперимент, в котором задавалось время обучения t и определялись средние значения $P_{ср}$ и $P_{зад}$ каждый раз для 100 испытаний. При этом использовалась стратегия 1. Получающиеся графики представлены на рис. 2. Видно, что в процессе обучения кривая научения растет по логистическому закону от 0,47 (уровень после предварительного обучения,) до 1. Вероятность

правильного выполнения задачи $P_{зад}$ сначала невелика, затем также возрастает, стремясь к 1. При использовании других стратегий характер изменения $P_{ср}$ и $P_{зад}$ такой же, скорость возрастания больше.

Приложение

Программа ПР-1

```

uses crt;
const N=10; K_is=100; a1=0.1; a2=0.2; a3=0.2;
slovo:array[1..10]of integer=(0,1,2,3,4,5,6,7,8,9);
var kol_isp,time,N_o,N_pr,N_osh,k,l,i,i1,i2,j: integer;
p: array[0..N,0..N] of real; o: array [-1..N]of integer;
N1,N2,Z1,Z2,P_sr,P_zad,dp,x,sum_p,sum,tt : real;
Label mm;
Procedure Pechat;
begin {raspechatka massiva p}
  For i1:=0 to 9 do begin writeln;
    For i2:=0 to 9 do write(' ',p[i1,i2]:2:3);
  end; writeln; end;
Procedure Normir; {normirovanie p}
begin sum:=0; For j:=0 to N-1 do sum:=sum+p[o[i-1],j];
  For j:=0 to N-1 do p[o[i-1],j]:=p[o[i-1],j]/sum; end;
BEGIN clrscr;
Repeat k:=0;
  For i:=0 to N do For j:=0 to N do p[i,j]:=1/N;
  Repeat inc(k); randomize; i2:=0;
{1. PREDVARITELNOE OBUCHENIE}
  For i:=2 to 10 do begin i1:=slovo[i];
    p[i2,i1]:=p[i2,i1]+a1*(1-p[i2,i1]);
    For j:=0 to N-1 do begin
      If j<>i1 then p[i2,j]:=p[i2,j]-a1*(1-p[i2,i1])/(N-1);
    end;
    i2:=i1; end;
until (k=5)or(Keypressed); {Pechat;}
{2. OBUCHENIE METODOM PROB I OSHIBOK}
l:=0; N_osh:=0; N_o:=0;
Repeat inc(l); o[0]:=0;
For i:=0 to 8 do begin inc(N_o); sum_p:=0;
x:=random(100)/100;
for j:=0 to N-1 do begin sum_p:=sum_p+p[i,j];
If x>sum_p then o[i]:=j+1; end; {uchenik vibiraet opraciu}
If (o[i]-o[i-1]=1)and(i<9) then begin inc(N_pr); {verno}
p[o[i-1],o[i]]:=p[o[i-1],o[i]]+a2*(1-p[o[i-1],o[i]]);
  Normir; end;

```

```

(*If o[i]-o[i-1]<>1 then begin inc(N_osh); {neverno}
  p[o[i-1],o[i-1]+1]:=p[o[i-1],o[i-1]+1]
    +a3*p[o[i-1],o[i-1]+1];
  Normir; goto mm; {o[i]:=o[i-1];} end;
end; mm: {2 i 4 strategii} *)
If o[i]-o[i-1]<>1 then begin inc(N_osh); {neverno}
  p[o[i-1],o[i]]:=p[o[i-1],o[i]]-a3*p[o[i-1],o[i]];
  Normir; {goto mm;} o[i]:=o[i-1]; end;
end; mm: {1 i 3 strategii}
until (N_o+N_osh>800)or(Keypressed); {Pechat;}
time:=N_o+N_osh; inc(kol_isp);
P_sr:=(p[0,1]+p[1,2]+p[2,3]+p[3,4]
  +p[4,5]+p[5,6]+p[6,7]+p[7,8]+p[8,9])/9;
P_zad:=p[0,1]*p[1,2]*p[2,3]*p[3,4]*p[4,5]
  *p[5,6]*p[6,7]*p[7,8]*p[8,9];
writeln(N_o,' Kol_osh ',N_osh,' Vrem ',time,
  ' P_sr ',P_sr,' P_zad ',P_zad);
N1:=N1+N_o; N2:=N2+N_osh; tt:=tt+time; Z1:=Z1+P_sr;
Z2:=Z2+P_zad;
until (Keypressed)or(kol_isp=K_is);
writeln('Kol_otv ',N1/K_is,' Kol_osh ',N2/K_is,
  ' Vremya ',tt/K_is);
writeln(' P_sr ',Z1/K_is,' P_zad ',Z2/K_is); readkey;
END.

```

Литература

1. Аткинсон Р., Бауэр Г., Кротерс Э. Введение в математическую теорию обучения. — М: Мир, 1969. — 486 с.
2. Зыкина А.В. Кибернетика: Конспект лекций. – Омск: Изд-во ОмГТУ, 2005. – 28 с.
3. Леонтьев Л.П., Гохман О.Г. Проблемы управления учебным процессом: Математические модели. — Рига, 1984. — 239 с.
4. Розанова Л.В. Основы кибернетики: конспект лекций – Омск: Изд-во ОмГТУ, 2009. – 60 с.
5. Майер Р.В. Задачи, алгоритмы, программы. [Электронный ресурс] — URL: <http://maier-rv.glazov.net>, <http://komp-model.narod.ru>.
6. Робертс Ф.С. Дискретные математические модели с приложениями к социальным, биологическим и экологическим задачам. – М.: Наука, Гл. ред. физ.-мат. лит., 1986. — 496 с.
7. Харин Ю.С. и др. Основы имитационного и статистического моделирования. Учебное пособие // Харин Ю.С., Малюгин В.И., Кирилица В.П., Лобач В.И., Хацкевич Г.А. — Мн.: Дизайн ПРО, 1997. — 288 с.