

DOI: 10.32517/2221-1993-2024-23-5-23-29

Н. И. Заводчикова*Ярославский государственный педагогический университет им. К. Д. Ушинского, г. Ярославль, Россия;
Средняя школа № 76, г. Ярославль, Россия***И. А. Быкова***Ярославский государственный педагогический университет им. К. Д. Ушинского, г. Ярославль, Россия;
Средняя школа № 26, г. Ярославль, Россия*

ПСИХОЛОГИЧЕСКИЕ АСПЕКТЫ УСТАНОВЛЕНИЯ ВНУТРИПРЕДМЕТНЫХ СВЯЗЕЙ МЕЖДУ ТЕМОЙ «СИСТЕМЫ СЧИСЛЕНИЯ» И РАЗДЕЛОМ «АЛГОРИТМЫ И ПРОГРАММИРОВАНИЕ»

Аннотация

Реализация внутрипредметных связей позволяет частично снизить остроту противоречия между информационной насыщенностью курса информатики и существующими сроками обучения. Для определения внутрипредметных связей целесообразно использовать ассоциативный подход, при этом под ассоциацией понимается форма организации психических процессов, которую они принимают на конечной стадии своего формирования. Так как локальные ассоциации имеют тенденцию к «затуханию», необходимо их включение в системы ассоциаций разного уровня. Оправданным является включение частно-системных ассоциаций, сформированных при изучении темы «Системы счисления», в структуру ассоциаций, формирующихся при обучении программированию. В статье приведены примеры заданий для различных тем раздела «Алгоритмы и программирование», предполагающие оперирование с числами в различных системах счисления. Критерием сформированности разнообразных ассоциативных связей могут служить способы решения школьниками заданий на анализ готовой программы.

Ключевые слова: внутрипредметные связи, ассоциация, система счисления, обучение программированию.

Контактная информация

Заводчикова Надежда Ивановна, канд. пед. наук, доцент,

доцент кафедры теории и методики обучения информатике, Ярославский государственный педагогический университет им. К. Д. Ушинского, г. Ярославль, Россия; *адрес:* 150000, Россия, г. Ярославль, ул. Республиканская, д. 108/1;
учитель информатики, средняя школа № 76, г. Ярославль, Россия; *адрес:* 150031, Россия, г. Ярославль, ул. Автозаводская, д. 69д;
e-mail: zaw-nadejda@yandex.ru

Быкова Ирина Альбертовна,

старший преподаватель кафедры теории и методики обучения информатике, Ярославский государственный педагогический университет им. К. Д. Ушинского, г. Ярославль, Россия; *адрес:* 150000, Россия, г. Ярославль, ул. Республиканская, д. 108/1;
учитель информатики, средняя школа № 26, г. Ярославль, Россия; *адрес:* 150045, Россия, г. Ярославль, ул. Блюхера, д. 74;
e-mail: i.bukova@yandex.ru

N. I. Zavodchikova

Yaroslavl State Pedagogical University named after K. D. Ushinsky, Yaroslavl, Russia;
School 76, Yaroslavl, Russia

I. A. Bikova

Yaroslavl State Pedagogical University named after K. D. Ushinsky, Yaroslavl, Russia;
School 26, Yaroslavl, Russia

PSYCHOLOGICAL ASPECTS FOR ESTABLISHING INTRA-SUBJECT RELATIONS BETWEEN THE THEME "NUMBER SYSTEMS" AND THE SECTION "ALGORITHMS AND PROGRAMMING"

Abstract

Implementing intra-subject relations allows to partially reduce the severity in contradiction between the information richness of the course of informatics and present duration of study. The article proposes to use an associative approach to determine intra-subject relations, when association refers to the form of mental processes organization that they take at the final stage of their formation. Since local associations tend to "fade", it is necessary to include them in association systems of different levels. It is justified to include private-system associations formed during the study of the theme "Number systems" into the structure of associations formed during programming training. The article contains examples of tasks for various themes in the section "Algorithms and programming", involving operation with numbers in various number systems. The criterion for the formation of various associative relations can be methods for students to solve sums for analyzing the finished program.

Keywords: intra-subject relations, association, number system, teaching programming.

1. Введение

Противоречие между информационной насыщенностью содержания школьного курса информатики и существующими сроками обучения не раз отмечалось в современной научно-методической литературе [2]. Одним из средств, позволяющим частично решить указанную проблему, является реализация внутрипредметных связей [5, 10, 14].

Под *внутрипредметными связями* можно понимать множество пар элементов знания, в которых первый компонент пары используется при изучении второго компонента [10], всевозможное отношение взаимной зависимости, обусловленности, общности между объектами одного учебного предмета [8]. Ю. А. Самарин рассматривает внутрипредметные и межпредметные связи как некоторый комплекс ассоциативных связей и выделяет разные уровни ассоциаций: локальные, частно-системные, внутрисистемные (внутрипредметные), межсистемные (межпредметные) [13].

Представители существующих направлений в психологии по-разному решают вопрос о роли ассоциаций в мыслительной деятельности человека [4]. Согласно Большому психологическому словарю, ассоциация — это возникающая в опыте индивида закономерная связь между двумя содержаниями сознания, которая выражается в том, что появление в сознании одного из содержаний влечет за собой и появление другого [1].

А. И. Назаров в своей статье [11] отмечает, что данное определение требует уточнения. Опираясь на развитие идей «опосредования» мыслительной деятельности в работах Л. С. Выготского, А. Н. Леонтьева, П. Я. Гальперина и других, он говорит, что переход от развернутого внешнего действия к свернутому, автоматизированному есть переход от действия, представленного максимальным количеством опосредований, к действию с минимальным количеством опосредований, которое и представляет собой ассоциацию.

Вслед за А. И. Назаровым под *ассоциациями* будем понимать естественную (натуральную) форму организации психических процессов, которую они приобретают на конечной стадии своего формирования [11].

Изучение отдельных порций учебного материала (параграфа, темы курса) приводит к формированию ограниченных систем ассоциаций, которые играют свою роль в мыслительном акте: на их базе происходят отбор, сопоставление связей, входящих в эту систему, выделение ключевых и второстепенных связей; актуализация необходимых ассоциаций находится под контролем смыслового содержания текущего действия, благодаря чему неуместные ассоциации тормозятся, а в сознание попадает та ассоциация, которая уместна в данной ситуации [13].

Естественным является вопрос об условиях формирования и подкрепления ассоциаций в процессе обучения. Как отмечали А. Н. Леонтьев, Т. В. Розанова, ассоциации формируются в составе деятельности индивида [9]. При изучении отдельных тем курса происходит формирование локальных и частно-системных ассоциаций. Предотвращение «отмирания» накопленных ассоциативных структур может осуществляться за счет концентрического построения курса (актуализация и закрепление частно-системных ассоциаций) и за счет

активного использования систем ассоциаций одной темы при изучении другой (создание внутрисистемных ассоциаций).

В содержании школьного курса информатики выделяют четыре раздела: «Цифровая грамотность», «Теоретические основы информатики», «Алгоритмы и программирование», «Информационные технологии». Изолированное изучение этих разделов, как правило, приводит к «затуханию» ассоциативных связей. Например, в нашем исследовании [7] мы отмечали, что использование разрядной сетки для опосредования действий с числами в различных системах счисления оказывает положительное влияние на осознанность действий учащихся и дает возможность быстрой актуализации созданных ассоциативных связей, однако, не влияет на длительность сохранности навыка у средних и слабых учащихся без дополнительного подкрепления.

Рассмотрим, каким образом можно частно-системные ассоциации, сформированные при изучении темы «Системы счисления», включить в структуру формирующихся ассоциаций раздела «Алгоритмы и программирование».

На уровне задачного и иллюстративного материала можно наблюдать установление внутрипредметных связей раздела «Алгоритмы и программирование» и темы «Системы счисления» в учебниках по информатике [3, 12]. Эти связи являются двусторонними и могут быть пролонгированными во времени, что позволит лучше усвоить как алгоритмы действий с числами в различных системах счисления, так и развить навыки программирования.

Ниже приведены примеры заданий, позволяющих постоянно актуализировать ассоциации, сформированные при изучении темы «Системы счисления», в процессе изучения различных тем раздела «Алгоритмы и программирование».

2. Линейные, разветвляющиеся и циклические алгоритмы

Изучение программирования, как правило, начинается со знакомства с **линейными алгоритмами**, где для решения задач используются операторы ввода и вывода, команда присваивания, арифметические операции; вводится понятие линейной записи выражения; учащиеся оперируют с числовыми и строковыми типами данных.

Приведем примеры заданий, которые позволяют на этом этапе актуализировать ассоциации, связанные с алгоритмами перевода натурального десятичного числа в систему счисления с основанием q и обратно.

1. Введите основание системы счисления q , переведите число 111_q в десятичную систему счисления и выведите его на экран.
2. Введите натуральное число x , выведите последнюю цифру числа x в троичной системе счисления.
3. Введите натуральное число x и основание системы счисления q . Выведите последнюю цифру числа x в системе счисления с основанием q .
4. Введите основания систем счисления p и q . Выведите минимальное целое число, которое в p -ричной и q -ричной системах счисления оканчивается на 0.

5. Введите строку, содержащую запись некоторого числа в двоичной системе счисления. Выведите строку, являющуюся результатом умножения этого числа на 2, в двоичной системе счисления.
6. Введите число N — количество знаков в двоичном представлении некоторого числа. Выведите количество знаков, которое будет иметь это число в четверичной, восьмеричной и шестнадцатеричной системах счисления.

Задания на разветвляющиеся алгоритмы также могут быть направлены на нахождение последних цифр числа в системе счисления с заданным основанием и на оценку количества цифр в числе.

1. Введите натуральное число x .
 - 1) Выведите «Yes», если число x в двоичной системе счисления оканчивается на «000», и «No» — в противном случае.
 - 2) Выведите «Yes», если число x в троичной системе счисления оканчивается на «21», и «No» — в противном случае.
 - 3) Выведите «Yes», если число x в пятеричной системе счисления оканчивается на две одинаковые цифры, и «No» — в противном случае.
 - 4) Выведите «Yes», если число x в пятеричной системе счисления является трехзначным, и «No» — в противном случае.
 - 5) Выведите «Yes», если число x в пятеричной системе счисления является трехзначным и оканчивается на 4, и «No» — в противном случае.
2. Введите натуральное число x и основание системы счисления q .
 - 1) Выведите «Yes», если введенное с клавиатуры число x в системе счисления с основанием q оканчивается на единицу, и «No» — в противном случае.
 - 2) Выведите «Yes», если введенное с клавиатуры число x в системе счисления с основанием q является двузначным, и «No» — в противном случае.

Возможно, описанные выше задания не стоит предлагать всему классу, чтобы не отвлекать школьников от основного материала, подлежащего усвоению, а рекомендовать только тем учащимся, которые уже владеют навыками разработки линейных и разветвляющихся алгоритмов.

При знакомстве школьников с **циклом по условию** традиционными являются задачи на нахождение суммы и количества цифр числа, обладающих некоторым свойством, в различных системах счисления. Разработка программ для решения указанных задач позволяет учащимся выйти на новый уровень обобщения изученных ранее алгоритмов.

При изучении **цикла с параметром** целесообразно использовать задания на обработку последовательности чисел, а также задания на генерацию чисел с заданным свойством, например:

1. Найдите количество чисел, меньших 1000, которые в семеричной системе счисления оканчиваются на 25.
2. Выведите максимальное и минимальное десятичные числа, которые в двоичной системе счисления имеют N знаков, K из которых — единицы.

На этом этапе обучения программированию, как правило, используют задания, предназначенные для подготовки к решению заданий 14 и 17 ЕГЭ по информатике [6], что также способствует актуализации и развитию ассоциативных структур, сформированных при изучении темы «Системы счисления».

3. Вспомогательные алгоритмы

Использование вспомогательных алгоритмов для оперирования с цифрами числа в различных системах счисления позволяет сделать решение следующих задач более структурированным, а также способствует сворачиванию уже сформированных ассоциативных структур, связывающих алгоритмы работы с цифрами числа в различных системах счисления и фрагменты программ, их реализующих. Как правило, на этом этапе обучения учащиеся могут написать программу без внешней опоры на алгоритм перевода, то есть с минимальным количеством опосредований.

1. Определите количество принадлежащих отрезку $[1000; 2000]$ натуральных чисел, у которых сумма цифр в шестеричной системе счисления больше 7, а в пятеричной — меньше 8.
2. Определите количество принадлежащих отрезку $[1000; 2000]$ натуральных чисел, у которых старшая цифра в троичной системе счисления равна 2.
3. Определите минимальное и максимальное принадлежащие отрезку $[1000; 2000]$ натуральные числа, у которых старшая цифра в троичной или семеричной системе счисления равна 2.
4. Определите среднее арифметическое принадлежащих отрезку $[251763; 514827]$ натуральных чисел, которые в троичной системе счисления содержат 7 двоек.
5. Определите максимальное из принадлежащих отрезку $[1000; 2000]$ натуральных чисел, у которых минимальная цифра в шестеричной системе счисления больше минимальной цифры в восьмеричной системе счисления, но меньше минимальной цифры в пятеричной системе счисления

4. Массивы (списки), строки

Строки и списки дают инструмент для хранения цифр числа в произвольной системе счисления, что способствует актуализации ассоциаций, связанных с принципами построения числа в позиционных системах счисления.

Существующий банк задач по этой теме можно дополнить следующими заданиями:

1. Выведите максимальное и минимальное десятичные числа, которые в троичной системе счисления имеют N знаков, K из которых — двойки.
2. Введите строку S , содержащую цифры и прописные латинские буквы, и число q — основание системы счисления. Выведите «Yes», если строка S — число, представленное в q -ричной системе счисления, и «No» — в противном случае.
3. Введите строку S , содержащую запись числа в двоичной системе счисления. Выведите строку, являющуюся записью этого числа в четверичной системе счисления.

4. Введите строку S , содержащую запись числа в девятеричной системе счисления. Выведите строку, являющуюся записью этого числа в троичной системе счисления.

Эти задания можно переформулировать, заменив используемую структуру данных для хранения значений цифр числа на список.

5. Задания на анализ готовой программы

Особый интерес представляют задачи на «чтение» готового алгоритма. Они могут быть использованы в качестве разминки в начале урока, предложены не всему классу, а только сильным ученикам как дополнительное задание. Предполагается, что учащиеся выполняют их без использования компьютера.

Общая формулировка задания имеет вид:

Что будет выведено на экран? Сформулируйте задачу, для решения которой предназначен данный фрагмент программы.

Внешняя речь на данном этапе служит инструментом обобщения — помогает учащимся перейти от оперирования с конкретными данными к обобщенному представлению о совершаемых действиях.

В таблице 1 приведены примеры заданий, рассмотрение которых целесообразно при изучении темы «Вспомогательные алгоритмы».

При изучении рекурсивных алгоритмов задание на опознание знакомого алгоритма перевода, представленного в рекурсивной форме, положительно скажется на создании вариативных ассоциативных связей, что будет способствовать более глубокому усвоению материала (табл. 2).

При изучении алгоритмов обработки строк можно предложить задания на работу с родственными системами счисления (табл. 3).

В таблице 4 представлены задания, в которых предлагается выполнить анализ программ, использующих списки.

В последних двух заданиях школьники могут дать различные ответы. Варианты ответов приведены в табли-

Таблица 1

Примеры заданий на анализ программы при изучении темы «Вспомогательные алгоритмы»

Задание	Примерный ответ
<pre>def sum_digit(x,p): s = 0 while x > 0: digit = x%p s += digit x = x // p return s counter = 0 for i in range(10, 20): if sum_digit(i, 5)>5: counter += 1 print(counter)</pre>	<p><i>Будет выведено:</i> 3</p> <p><i>Задача:</i> Найти количество чисел в диапазоне от 10 до 19, у которых сумма цифр в пятеричной системе счисления больше пяти.</p>
<pre>def count_digit(x,p): k = 0 while x > 0: k += 1 x = x // p return k s = 0 for i in range(2, 10): if count_digit(32, i) > 3: print(i)</pre>	<p><i>Будет выведено:</i> 2 3</p> <p><i>Задача:</i> Найти основания систем счисления, в которых число 32 имеет более трех знаков.</p>
<pre>def st_digit(x, p): while x > 0: k = x % p x = x // p return k for i in range(2, 16): if st_digit(15, i) == 1: print(i)</pre>	<p><i>Будет выведено:</i> 2 3 8 9 10 11 12 13 14 15</p> <p><i>Задача:</i> Вывести основания систем счисления, в которых число 15 начинается с единицы.</p>

Таблица 2

Примеры заданий на анализ программы при изучении темы «Рекурсия»

Задание	Примерный ответ
<pre>def f(n): if n > 0: print (n % 3) f(n // 3) f(29)</pre>	<p><i>Будет выведено:</i> 2 0 0 1</p> <p><i>Задача:</i> Выведите в обратном порядке цифры числа в троичной системе счисления.</p>
<pre>def f(n): if n > 0: f(n // 3) print (n % 3, end="") f(29)</pre>	<p><i>Будет выведено:</i> 1002</p> <p><i>Задача:</i> Переведите число в троичную систему счисления.</p>
<pre>def f(n): if n == 0: return 0 else: return n % 3 + f(n // 3) print(f(29))</pre>	<p><i>Будет выведено:</i> 3</p> <p><i>Задача:</i> Найдите сумму цифр числа в троичной системе счисления.</p>

Таблица 3

Примеры заданий на анализ программы при изучении темы «Обработка строк»

Задание	Примерный ответ
<pre>ns2 = "10110101" if len(ns2) % 2 != 0: ns2 = '0' + ns2 ns4 = "" for i in range(0, len(ns2), 2): digit = ns2[i:i+2] if digit == '00': ns4 = ns4 + '0' if digit == '01': ns4 = ns4 + '1' if digit == '10': ns4 = ns4 + '2' if digit == '11': ns4 = ns4 + '3' print(ns4)</pre>	<p><i>Будет выведено:</i> 2311</p> <p><i>Задача:</i> Переведите число из двоичной системы счисления в четверичную.</p>
<pre>ns9 = "2507" ns3 = '' for x in ns9: ns3 = ns3 + str(int(x) // 3) + str(int(x) % 3) if ns3[0] == "0": ns3 = ns3[1:] print(ns3)</pre>	<p><i>Будет выведено:</i> 2120021</p> <p><i>Задача:</i> Переведите число из девятеричной системы счисления в троичную.</p>

це в порядке увеличения длины цепочки ассоциативных связей, построение которых необходимо для решения задачи.

Некоторые исследователи отмечают, что чем длиннее цепочки, представляющие ассоциативные связи, тем шире возможности школьника в решении задач [14].

Мы провели **опрос**, в котором участвовали учителя, студенты и школьники X и XI класса. Было предложено

сформулировать задачу, для решения которой предназначены представленные в последних двух строках таблицы 4 фрагменты программ.

В задании, представленном во второй строке таблицы 4, третий вариант ответа дали только учителя информатики, отдельные студенты и некоторые учащиеся XI класса, дополнительно занимающиеся подготовкой к ЕГЭ по информатике. Несколько

Примеры заданий на анализ программы при изучении темы «Списки»

Задание	Примерный ответ
<pre>def fd(x): while x > 0: f = x % 6 x = x // 6 return f A=[] for x in range(1, 26): if fd(x) == 4: A.append(x) print(A)</pre>	<p><i>Будет выведено:</i> [4, 24, 25]</p> <p><i>Задача:</i> Вывести числа, меньшие 26, которые в шестеричной системе счисления начинаются с 4.</p>
<pre>def sd(x, q): s = 0 while x > 0: s = s + x % q x = x // q return s a = [16,1,15,32,31,64,65,2048] b = [] for x in a: b.append(sd(x, 2)) print(b.count(1))</pre>	<p><i>Будет выведено:</i> 5</p> <p><i>Задача:</i> <i>Вариант ответа 1.</i> Вывести количество чисел, которые в двоичной системе счисления имеют сумму цифр 1. <i>Вариант ответа 2.</i> Вывести количество чисел, которые в двоичной системе счисления содержат одну единицу. <i>Вариант ответа 3.</i> Вывести количество чисел, которые являются степенями двойки.</p>
<pre>def binary(x): s='' while x>0: s=str(x % 2)+s x=x//2 return s A = [2,4,16,32,42,45,48,104] B = [] for x in A: if binary(x)[-2:] == '00': B.append(x) print(max(B))</pre>	<p><i>Будет выведено:</i> 104</p> <p><i>Задача:</i> <i>Вариант ответа 1.</i> Вывести максимальное число в двоичной системе счисления, оканчивающееся на два нуля. <i>Вариант ответа 2.</i> Вывести максимальное число, делящееся на 4.</p>

сильных учащихся X класса предлагали первый и второй варианты ответа; а третий вариант ответа смогли сформулировать только после определения результата работы программы.

Аналогично при выполнении задания, представленного в третьей строке таблицы 4, только респонденты с большим опытом решения задач по теме «Системы счисления» смогли сформулировать второй вариант ответа.

Очевидно, что решение данных задач в какой-то мере может служить критерием сформированности разнообразных ассоциативных связей как внутри темы «Системы счисления» и раздела «Алгоритмы и программирование», так и между ними. Если учащийся дает верный ответ достаточно быстро и без выполнения программы по шагам, то это может свидетельствовать о сформированности межсистемных ассоциаций и глубоком усвоении материала. Если школьник дает правильный ответ, выполнив алгоритм по шагам для некоторых входных данных, то это означает, что материал усвоен, но ассоциативные структуры не приобрели свернутую форму.

6. Заключение

Установление двусторонних пролонгированных во времени внутрипредметных связей между темой «Системы счисления» и разделом «Алгоритмы и программирование» способствует включению большего числа элементов знаний в общую систему, что позволяет предупредить формализм знаний, сократить затраты учебного времени.

Осознание психологических аспектов реализации внутрипредметных связей позволяет более технологично подходить к проектированию учебного материала, предусмотреть достаточное количество заданий для формирования ассоциаций и интервалы времени, через которые эти ассоциации целесообразно подкреплять.

Список источников

1. Большой психологический словарь / сост. и общ. ред. Б. Мещеряков, В. Зинченко. СПб.: Прайм-ЕВРОЗНАК, 2003. 672 с. EDN: TDNOGT.
2. Босова Л. Л. О новых подходах к изучению школьной информатики в условиях цифровой трансформации общества // Информатика в школе. 2022. № 4. С. 5–14. EDN: DKRLZV. DOI: 10.32517/2221-1993-2022-21-4-5-14/

3. *Босова Л. Л., Босова А. Ю.* Информатика. 8 класс. М.: БИНОМ. Лаборатория знаний, 2021. 176 с. EDN: EVEMHI.
4. *Горячев В. В.* О роли ассоциаций в отечественных исследованиях мышления // Мир науки. Педагогика и психология. 2021. Т. 9. № 6. С. 1–10. EDN: TDQQFG.
5. *Далингер В. А., Симонженков С. Д.* Реализация внутрипредметных связей при решении математических задач посредством когнитивно-визуальной деятельности. Омск: Изд-во ОмГПУ, 2013. 195 с. EDN: SXOPLX.
6. ЕГЭ по информатике (2024) // Преподавание, наука и жизнь: сайт Константина Полякова. <https://kpolyakov.spb.ru/school/ege.htm>
7. *Заводчикова Н. И., Быкова И. А.* О средствах материализации при изучении темы «Системы счисления» // Математика и математическое образование: проблемы, технологии, перспективы. Материалы 42-го Международного научного семинара преподавателей математики и информатики университетов и педагогических вузов (Смоленск, 12–14 октября 2023 года). Смоленск: СмолГУ, 2023. С. 423–426. EDN: IJXRCN.
8. *Костюченко Р. Ю.* Обучение учащихся предельной аналогии при реализации внутрипредметных связей школьного курса геометрии: автореф. дис. ... канд. пед. наук. Омск, 2000. 21 с. EDN: ZKNXNP.
9. *Леонтьев А. Н., Розанова Т. В.* Зависимость образования ассоциативных связей от содержания действия: экспериментальное исследование // Советская педагогика. 1951. № 10. С. 60–77.
10. *Монахов В. М., Гуревич В. Ю.* Об одном методе системного анализа внутрипредметных связей // Математика в школе. 1980. № 2. С. 54–57.
11. *Назаров А. И.* Ассоциации без ассоцианизма // Психологический журнал Международного университета природы, общества и человека «Дубна». 2013. № 2. С. 43–77.
12. *Поляков К. Ю., Еремин Е. А.* Информатика. 10 класс. Базовый и углубленный уровни: учебник. В 2 ч. Ч. 2. М.: БИНОМ. Лаборатория знаний, 2016. 352 с.
13. *Самарин Ю. А.* Об ассоциативной природе умственной деятельности // Вопросы психологии. 1957. № 2. С. 18–30.
14. *Строчилов П. А.* Реализация внутрипредметных связей при обучении физике на основе циклической модели построения содержания учебного курса: дис. ... канд. пед. наук. Волгоград, 2015. 195 с. EDN: BWPZAE.