

DOI: 10.32517/2221-1993-2023-22-2-79-86

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

ПОЭТАПНОЕ ФОРМИРОВАНИЕ УМСТВЕННЫХ ДЕЙСТВИЙ ПРИ ОБУЧЕНИИ ШКОЛЬНИКОВ ПРОГРАММИРОВАНИЮ

Аннотация

Использование при обучении школьников программированию набора заданий, разработанных в соответствии с основными положениями теории поэтапного формирования умственных действий, позволяет повысить качество обучения. На первом этапе открытия алгоритма целесообразно предложить учащимся набор заданий и вопросов, которые будут способствовать самостоятельному выделению ориентировочной основы действия. Задания на заполнение трассировочной таблицы для конкретных числовых данных соответствуют этапу материализованного действия. На этапе формирования действия во внешней речи можно предложить школьникам упражнения на работу с формулировкой алгоритма; для этого можно использовать задачи с пропусками в вербальном описании алгоритма, задания на установление соответствия вербального описания алгоритма и строк кода программы. Переводу действия во внешнюю речь «про себя» будут способствовать задания на поиск ошибки в решении задачи, заполнение пропуска в коде программы. Для автоматизации описываемого умственного действия необходимо выполнение школьниками достаточного количества стандартных задач на написание программы. Для дальнейшего развития навыка применения алгоритма необходимо использовать задачи, в которых данный алгоритм является одним из этапов решения. В статье предлагаются задания, которые оформлены в виде коллекции в сервисе создания интерактивных упражнений LearningApps.

Ключевые слова: программирование, теория поэтапного формирования умственных действий, ориентировочная основа действия, набор задач.

1. Введение

Вопросы методики обучения программированию на разных этапах обучения обсуждаются в методической литературе не один десяток лет [1, 2]. В [6] отмечено, что управление познавательной деятельностью учащихся при изучении программирования может осуществлять-

ся по конечному продукту, при этом путь, ведущий ученика к получению этого продукта, остается неизвестен для учителя. Более эффективным на начальных этапах обучения является управление, опирающееся на циклическую обратную связь, предполагающую получение сведений о процессе получения конечного продукта. Для реализации пошагового контроля решения

Контактная информация

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 общеобразовательная школа № 26, г. Ярославль, Россия; *адрес:* 150045, Россия, г. Ярославль, ул. Труфанова, д. 16;
e-mail: i.bukova@yandex.ru

I. A. Bikova

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

STEP-BY-STEP FORMATION OF MENTAL ACTIONS WHEN TEACHING PROGRAMMING FOR SCHOOLCHILDREN

Abstract

The use of a set of tasks developed in accordance with the basic provisions of the theory of step-by-step formation of mental actions in teaching programming to schoolchildren makes it possible to improve the quality of learning. At the first stage of the discovery of the algorithm, it is advisable to offer students a set of tasks and questions that will contribute to the independent identification of the orientation basis of the action. The tasks for filling in the trace table for specific numeric data correspond to the stage of the materialized action. At the stage of forming an action in external speech, students can be offered exercises to work with the formulation of the algorithm; for this, you can use tasks "with omissions" in the verbal description of the algorithm, tasks to establish the correspondence of the verbal description of the algorithm and the lines of the program code. The translation of the action into external speech "to yourself" will be facilitated by tasks to find an error in solving the problem, filling in a gap in the program code. To automate the described mental action, it is necessary for schoolchildren to perform a sufficient number of standard tasks for writing a program. To further develop the skill of applying the algorithm, it is necessary to use tasks in which this algorithm is one of the solution stages.

Keywords: programming, theory of step-by-step formation of mental actions, orientation basis of action, set of tasks.



Рис. 1

задач на программирование можно использовать серии взаимосвязанных задач (укрупненные упражнения), реализованные в цифровой обучающей среде.

Использование укрупненных упражнений позволяет организовать работу учащихся по самостоятельному открытию теоретических фактов и алгоритмов [4]. При обучении программированию укрупненные упражнения могут быть построены в соответствии с основными положениями теории поэтапного формирования умственных действий. В исследованиях П. Я. Гальперина [5] и Н. Ф. Талызиной [11, 12] отмечено, что усвоение учебного материала будет происходить более успешно, если организовать планомерную поэтапную деятельность учащихся по составлению схемы ориентировочной основы действия, формированию действия в материальном (материализованном) виде, формированию внешнеречевого действия и действия во внешней речи «про себя», автоматизации действия во внутренней речи.

Рассмотрим, какие задания можно использовать на каждом из этапов при изучении в VIII классе алгоритмов,

предполагающих анализ цифр числа. Алгоритмы реализуются на языке программирования Python.

Представленные в данной статье задания оформлены в виде коллекции в сервисе создания интерактивных упражнений LearningApps (рис. 1):
<https://learningapps.org/view30388593>

2. Примеры заданий для создания схемы ориентировочной основы деятельности

На этапе открытия алгоритма целесообразно предложить учащимся набор заданий и вопросов, которые позволят им самостоятельно выделить ориентировочную основу действия (образ действия, на основе которого происходит управление действием) в обобщенном виде [12]. В работах Н. Ф. Талызиной отмечалось, что действию, сформированному на ориентировочной основе, в которой ориентиры представлены в общем виде, а основа действия составляется обучающимся самостоятельно, присущи большая устойчивость, широта переноса [11, 12].

На этом этапе могут присутствовать задания:

- на актуализацию знаний, необходимых для составления алгоритма;
- на решение задачи для конкретных входных данных;
- на обобщение предыдущих действий с числовыми данными в форме команд языка программирования;
- на соотнесение вербальной формулировки алгоритма и кода программы.

Например, при создании ориентировочной основы действия решения задачи на нахождение суммы цифр десятичного числа можно использовать следующие задания.

Задание 1 (рис. 2).

Найдите значение выражения:

- $56543//10$;
- $56543\%10$;
- $56543//100$;
- $56543\%100$;
- $56543//2$;
- $56543\%2$.

Задание 2.

Напишите команды для нахождения:

- последней цифры числа X ;
- двух последних цифр числа X ;
- трех последних цифр числа X ;
- числа, полученного из числа X отбрасыванием последней цифры.

Задание 3 (рис. 3).

Установите соответствие между командой и ее описанием.

Ответ.

Команда	Описание команды
$X=X//10$	«Отбрасывание» последней цифры числа X
$X//10$	Число X без последней цифры
$X\%10$	Последняя цифра числа X
$s=s+X\%10$	Увеличение значения переменной s на значение последней цифры числа X



Рис. 2

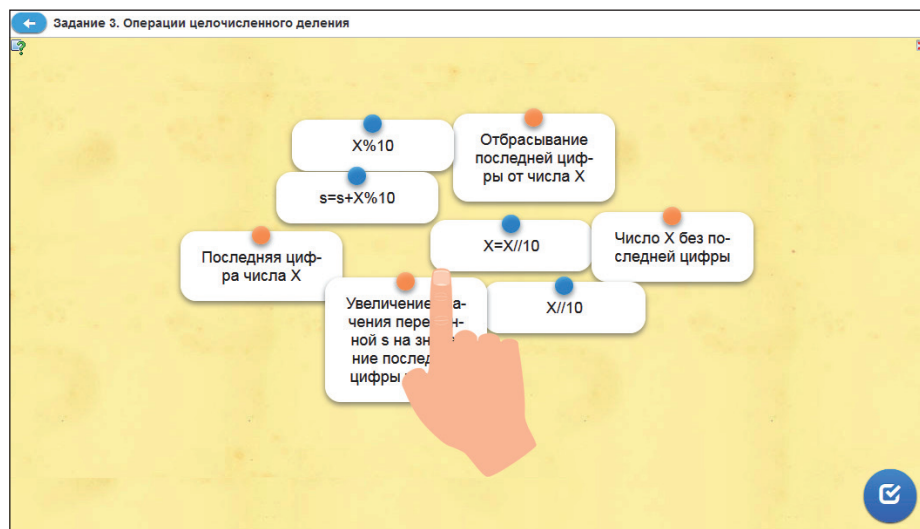


Рис. 3

Задание 4. Трассировка алгоритма нахождения суммы цифр числа

Номер итерации	Значение числа X	Последняя цифра числа X	Текущая сумма S цифр числа
1	324527	7	7
2			
3			
4			
5			
6			
7			
команда			

Задание

Вася при нахождении суммы цифр числа $X=324527$ заполнял приведенную ниже таблицу. Часть информации из таблицы утеряна. Помогите Васе восстановить пропуски в таблице.

В последней строке укажите команды, которые можно использовать

OK

Рис. 4

Задание 4 (рис. 4).

Вася при нахождении суммы цифр числа $X = 324527$ заполнял приведенную ниже таблицу. Часть информации из таблицы утеряна. Помогите Васе восстановить пропуски в таблице.

Номер итерации	Значение числа X	Последняя цифра числа X	Текущая сумма цифр
1	324527	7	7
2	32452	2	$7 + 2 = 9$
3	3245		$9 + _ = _$
4	324		$14 + _ = _$
5			
6			
7	0		
Команда			

В последней строке укажите команды, которые можно использовать для:

- изменения значения числа X ;
- нахождения последней цифры числа X ;
- нахождения текущего значения суммы цифр числа.

Задание 5 (рис. 5).

Заполните пропуски в описании алгоритма нахождения суммы цифр числа.

Инициализировать переменную суммы цифр числа
 Пока число $_$
 увеличить сумму цифр числа на $_$ цифру числа X ,
 $_$ из числа X последнюю цифру.

Задание 6 (рис. 6).

Установите соответствие между алгоритмом нахождения суммы цифр числа X и строками программы.

Задание 5. Алгоритм нахождения суммы цифр числа

Инициализировать переменную суммы цифр числа

Пока число не равно $_$

увеличить сумму цифр числа на $_$ цифру числа X ,
 $_$ из числа X последнюю цифру

Задание

Заполните пропуски в алгоритме нахождения суммы цифр числа

OK

Рис. 5

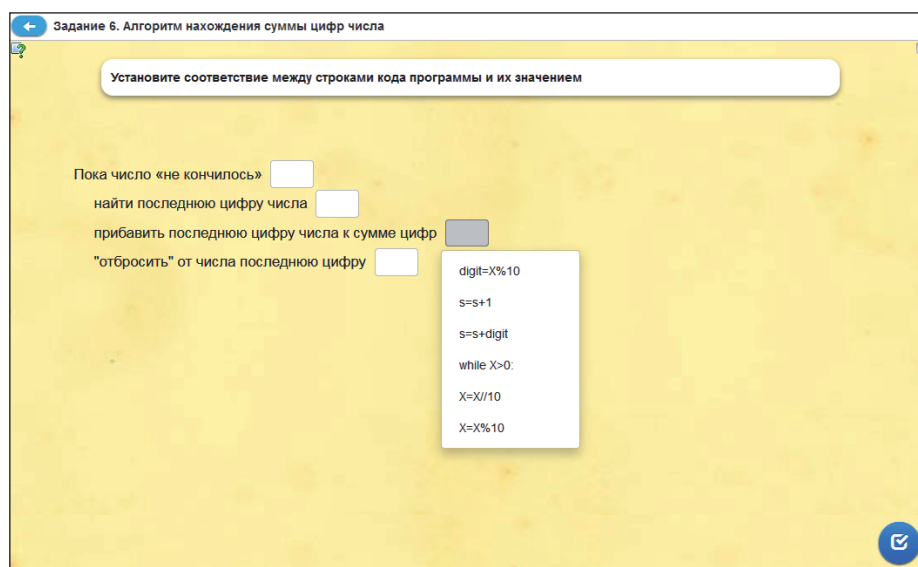


Рис. 6

Ответ.

Описание алгоритма	Программа
Пока число «не кончилось»	<code>while X>0:</code>
Найти последнюю цифру числа	<code>digit=X%10</code>
Прибавить последнюю цифру числа к сумме цифр	<code>s=s+digit</code>
«Удалить» из числа последнюю цифру	<code>X=X//10</code>

3. Примеры заданий для выполнения действия в материализованной форме

После того как схема ориентировочной основы действия учащимися построена, необходимо предложить им несколько упражнений на выполнение действия в материализованной форме. Для этого можно использовать задачи на выполнение алгоритма для конкретных числовых данных. Эти задания могут предполагать за-

полнение трассировочной таблицы или определение результата работы программы.

Приведем пример заданий на выполнение действия в материализованной форме при изучении алгоритма нахождения суммы цифр десятичного числа.

Задание 7 (рис. 7).

Запишите результат выполнения каждой из команд на каждой итерации цикла:

	Итерации цикла					
<code>X=5324527</code>						
<code>s=0</code>						
<code>while X>0:</code>	да	да				
<code>digit=X%10</code>	7	2				
<code>s=s+digit</code>	7	9				
<code>X=X//10</code>	32452	3245				

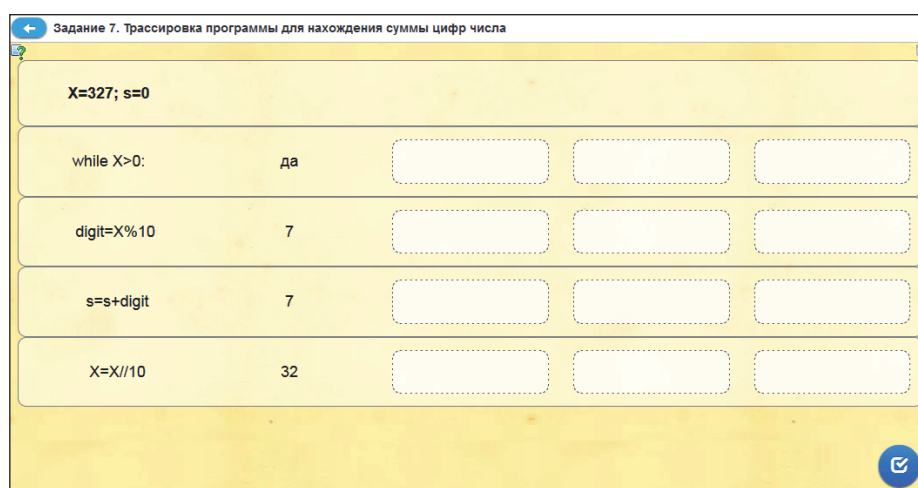


Рис. 7

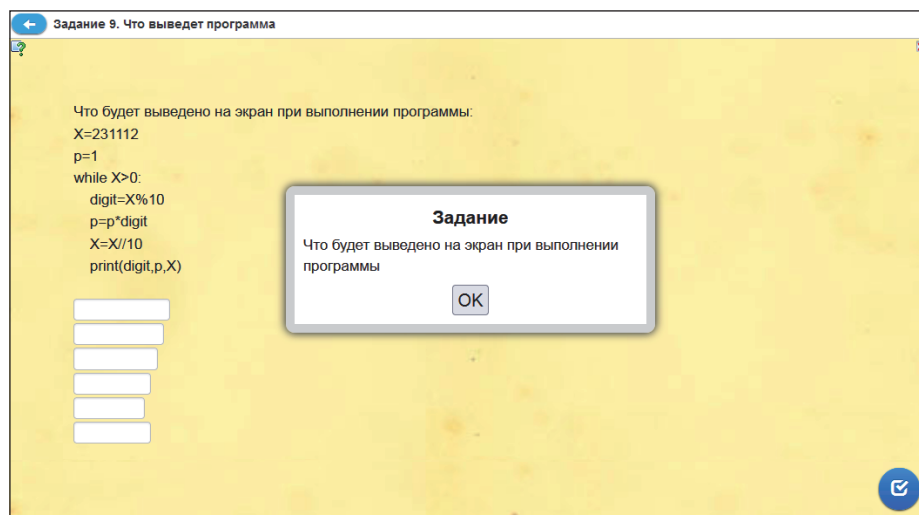


Рис. 8

Задание 8.

Запишите результат выполнения каждой из команд на каждой итерации цикла:

	Итерации цикла						
<code>x=311252</code>							
<code>counter=0</code>							
<code>while X>0:</code>							
<code>digit=X%10</code>							
<code>counter+=1</code>							
<code>X=X//10</code>							

Задание 9 (рис. 8).

Что будет выведено на экран при выполнении приведенной ниже программы?

```
x=2311112
p=1
while X>0:
    digit=X%10
    p=p*10
    X=X//10
    print(digit,p,X)
```

4. Примеры заданий для выполнения действия во внешней и внутренней речи

На этапе формирования действия во внешней речи можно предложить школьникам задачи на работу с формулировкой алгоритма. Для этого можно использовать задачи с пропусками в вербальном описании алгоритма, задания на установление соответствия вербального описания алгоритма и строк кода программы. Переводу действия во внешнюю речь «про себя» будут способствовать задания на поиск ошибки в решении задачи, заполнение пропуска в коде программы.

При изучении алгоритмов работы с цифрами числа на этом этапе можно использовать следующие задания.

Задание 10.

Заполните пропуски в алгоритме нахождения количества цифр числа.

Инициализировать счетчик

Пока число ____
увеличить счетчик цифр числа на ____
____ из числа последнюю цифру

Задание 11 (рис. 9).

Установите соответствие между фрагментами кода программы и описанием алгоритма нахождения количества цифр числа.

Ответ.

Описание алгоритма	Фрагмент программы
Инициализация счетчика	<code>count=0</code>
Пока в числе <i>X</i> не закончились цифры	<code>while X>0:</code>
увеличить счетчик цифр числа на единицу	<code>count=count+1</code>
«отбросить» от числа <i>X</i> последнюю цифру	<code>X=X//10</code>
Вывод результата	<code>print(count)</code>

Задание 12.

Опишите алгоритм вывода на экран цифр числа в обратном порядке.

Задание 13.

Опишите алгоритм нахождения количества четных цифр в числе.

Задание 14.

Заполните пропуски в программе для решения задачи вывода на экран цифр числа в обратном порядке:

```
while ____:
    print(____)
X=____
```

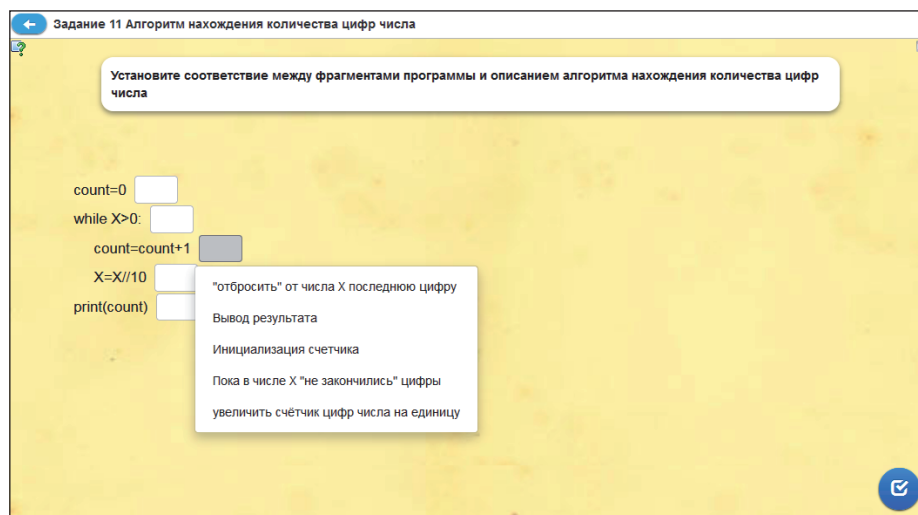


Рис. 9

Задание 15.

Заполните пропуски в программе для решения задачи нахождения суммы цифр числа:

```
s=___
while ___:
    s=s+___
    X=___
print(s)
```

Задание 16.

Заполните пропуски в программе для решения задачи нахождения количества цифр числа:

```
counter=___
while ___:
    counter=___
    X=___
print(___)
```

Задание 17.

Заполните пропуски в программе для решения задачи нахождения количества четных цифр числа:

```
counter=___
while ___:
    digit=___
    if digit%2==___:
        counter+=1
    X=___
print(counter)
```

Задание 18 (рис. 10).

Установите соответствие между фрагментом программы и его назначением.

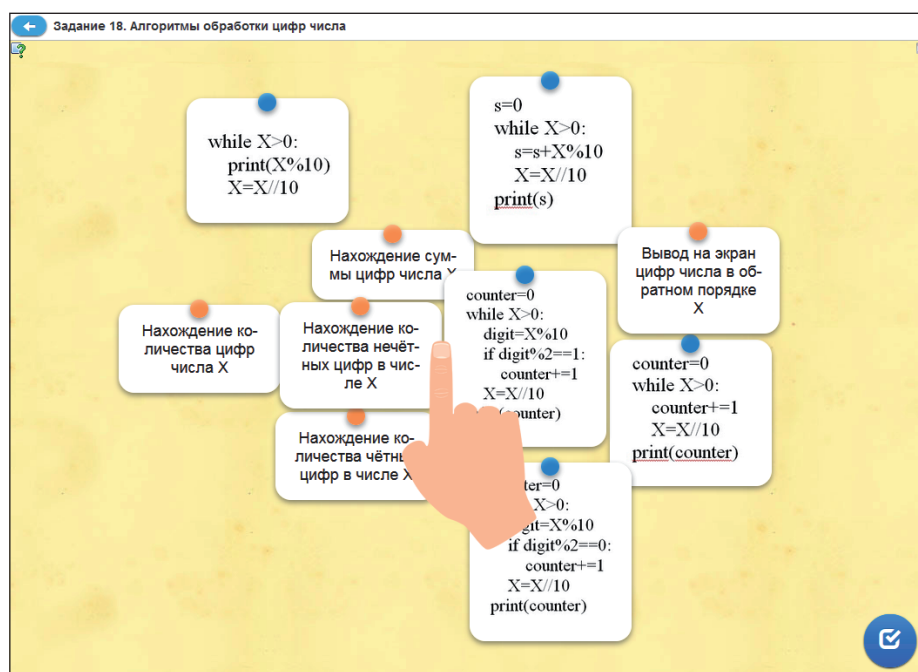


Рис. 10

Ответ.

Назначение фрагмента программы	Фрагмент программы
Нахождение суммы цифр числа X	<pre>s=0 while X>0: s=s+X%10 X=X//10 print (s)</pre>
Нахождение количества цифр числа X	<pre>counter=0 while X>0: counter+=1 X=X//10 print (counter)</pre>
Нахождение количества четных цифр в числе X	<pre>counter=0 while X>0: digit=X%10 if digit%2==0: counter+=1 X=X//10 print (counter)</pre>
Нахождение количества нечетных цифр в числе X	<pre>counter=0 while X>0: digit=X%10 if digit%2==1: counter+=1 X=X//10 print (counter)</pre>
Вывод на экран цифр числа X в обратном порядке	<pre>while X>0: print (X%10) X=X//10</pre>

5. Заключение

Для автоматизации умственного действия, связанного с алгоритмом, необходимо выполнение школьниками достаточного количества стандартных задач на написание программы. Подобные задачи можно найти в [3, 10]. Дальнейшее развитие навыка будет происходить при рассмотрении задач, в которых данный алгоритм является одним из этапов решения [7, 10].

Предложенные в статье типы заданий могут использоваться для организации фронтальной или самостоятельной работы учащихся. Удобно перенести указанный набор упражнений в цифровую среду — соз-

дать коллекцию заданий на LearningApps [9] или урок на Stepik.org [8].

Опыт показывает, что использование при изучении программирования заданий, разработанных в соответствии с основными положениями теории поэтапного формирования умственных действий, позволяет повысить уровень осознанности ориентировочной, исполнительской и контролирующей частей действий при написании программы.

Список источников

1. Босова Л.Л. Как учат программированию в XXI веке: отечественный и зарубежный опыт обучения программированию в школе // Информатика в школе. 2018. № 6. С. 3–11. EDN: XZOOJV.
2. Босова Л. Л. Программирование как инструмент формирования вычислительного мышления обучающихся // Информатика в школе. 2020. № 10. С. 4–10. EDN: GURIPH. DOI: 10.32517/2221-1993-2020-19-10-4-10
3. Босова Л. Л., Босова А. Ю., Аквиланов Н. А. Информатика. 7–9 классы. Компьютерный практикум. М.: БИНОМ. Лаборатория знаний, 2021. 192 с.
4. Быкова И. А., Заводчикова Н. И. Укрепленные упражнения как средство подготовки школьников к основному государственному экзамену по информатике // Информатика в школе. 2021. № 8. С. 12–17. EDN: WJSYCN. DOI: 10.32517/2221-1993-2021-20-8-12-17
5. Гальперин П. Я. Методы обучения и умственное развитие ребенка. М.: Издательство Московского университета, 1985. 45 с.
6. Заводчикова Н. И., Быкова И. А. Управление познавательной деятельностью учащихся при изучении программирования на уроках информатики // Математика и проблемы образования. Материалы 41-го Международного научного семинара преподавателей математики и информатики университетов и педагогических вузов. Киров, 2022. С. 213–215. EDN: OQIDQI.
7. Информатикс. <https://informatics.msk.ru/>
8. Образовательная платформа Stepik. <https://stepik.org/>
9. Онлайн-сервис Learningapps. <https://learningapps.org/>
10. 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.
11. Талызина Н. Ф. Педагогическая психология: учебное пособие для студентов средних педагогических учебных заведений. М.: Академия, 1998. 288 с.
12. Талызина Н. Ф. Управление процессом усвоения знаний (психологические основы). М.: Издательство Московского университета, 1984. 345 с.