

Компетенции для анализа результата исполнения алгоритма: задание вычисления количества различных результатов для заданных команд и длины программы в ЕГЭ по информатике

Попов Владислав Сергеевич,

эксперт (проектная группа «Информатика в предпрофессиональном образовании») Института развития профильного обучения, ГАОУ ВО города Москвы «Московский городской педагогический университет»
E-mail: popovvs@mgpu.ru

Алефиренко Евгения Александровна,

старший методист (проектная группа «Информатика в предпрофессиональном образовании») Института развития профильного обучения, ГАОУ ВО города Москвы «Московский городской педагогический университет»
E-mail: alefirenkoea@mgpu.ru

Черницына Лариса Юрьевна,

методист (проектная группа «Информатика в предпрофессиональном образовании») Института развития профильного обучения ГАОУ ВО города Москвы «Московский городской педагогический университет»
E-mail: chernicynalyu@mgpu.ru

В статье рассмотрены комбинаторные, графические, программные методы решения задания вычисления количества различных результатов описанной программы для формального исполнителя. Для каждого рассмотренного метода приведены одно или несколько решений и соответствующие компетенции для базового и углубленного уровней изучения учебного предмета «Информатика». При комбинаторном методе решения выделяются различные типы программ, в графическом решении используется дерево, вершинами которого являются получаемые числа, а дугами – команды исполнителя, в качестве решений на языке программирования приведены четыре листинга программ, каждая последующая программа имеет более высокий уровень абстракции и задействует всё более глубокие знания и навыки программирования. Для надлежащего изучения описанного предметного содержания в различной образовательной среде авторы предлагают использование укрупнённых дидактических единиц, перевёрнутого класса, интегрированного обучения. Различные используемые методы решения предложенного задания выявляют необходимость демонстрации предметных и межпредметных связей, формирования компетенций и компетентностей в их совокупности.

Ключевые слова: компетенция, компетентность, компетентностный подход, алгоритм, результат исполнения алгоритма, вычисление количества чисел, ЕГЭ, ЕГЭ по информатике.

Введение

Демонстрационный вариант контрольных измерительных материалов ЕГЭ по информатике 2012 года содержал задание В13 [1, с. 16] на вычисление количества чисел – результатов исполнения описанного алгоритма: «У исполнителя Кузнечик две команды: 1) Прибавь 3. 2) Вычти 2. Первая из них увеличивает число на экране на 3, вторая – уменьшает его на 2 (отрицательные числа допускаются). Программа для Кузнечика – это последовательность команд. Сколько различных чисел можно получить из числа 1 с помощью программы, которая содержит ровно 5 команд?» Правильным ответом на задание является число 6.

В следующем 2013 году задание В13 было изменено на задание вычисления количества программ, для решения которого необходимо использовать *методы динамического программирования [2, 3] вместо ранее применяемых комбинаторных и графических методов.*

Знакомство обучающихся с исполнителем «Кузнечик» обычно начинается на уровне основного общего образования [4, с. 123], вместе с тем этот и другие исполнители используются для обучения программированию не только на уровнях основного и среднего, но также – на начальном уровне общего образования [5, с. 66–68]. Некоторые авторы отмечают целесообразность использования данного исполнителя в *интегрированном обучении школьников математике и информатике [6, 7].* Элементы содержания учебного предмета «Информатика» для обучающихся на уровне основного общего образования представлены в виде практикума [8, с. 42–44].

Хотя способы решения задания существенно изменились, в статье показано, что устаревшее задание не потеряло своей актуальности: обучающиеся, которые выбрали информатику для сдачи ЕГЭ, по-прежнему демонстрируют низкие образовательные результаты относительно вышеприведённого задания, в то время как теоретические и практические основы для его решения должны быть усвоены уже на уровне основного общего образования. В статье показаны результаты выполнения задания обучающимися в 2024–2025 учебном году, аналитический и программный способы решения вышеприведённого задания В13 демо-версии 2012 года.

Материалы и методы исследования

Материалы исследования: задание на вычисление количества чисел для заданных команд и их количества, результаты его выполнения обучающимися на уровне среднего общего образования.

Методы исследования: для демонстрации способов решения задания используются комбинаторные, графические методы и методы программирования, для каждого рассмотренного метода выделены формируемые компетенции/компетентности на основе компетентностного подхода в образовании; для анализа результатов выполнения задания обучающимися используются базовые статистические методы; методы декомпозиции по уровням и видам применяются при обсуждении формируемых компетенций.

Результаты исследования и их обсуждение

Результаты выполнения задания обучающимися

В результате выполнения вышеприведённого задания обучающимися 10–11-х классов на подготовительных курсах технического вуза-партнёра проектов «Инженерный класс в московской школе» и «ИТ-класс в московской школе» верный ответ привели 10 из 22 ($\approx 45,5\%$) учеников базовой группы и 4 из 13 учеников профильной группы ($\approx 31\%$). После получения информации о неверном ответе три ученика базовой группы и два ученика профильной группы смогли верно исправить свой ответ, после данных исправлений процент правильных ответов увеличился до 59% и 46% для базовой и профильной групп соответственно. Превышение доли корректных ответов в базовой группе по отношению к профильной в числе прочих причин может быть обусловлено небольшим размером выборки. Следующим по частоте, но неверным ответом являлось число 32 – количество различных программ из 5 команд. Среди других неверных ответов также встречались числа 2, 5, 7, 13, 16, 26, 30, 44, 50, 120.

Хотя рассматриваемое задание является гораздо более простым для выполнения по сравнению с современными заданиями 23 ЕГЭ по информатике на тему динамического программирования, процент его выполнения соответствует среднему проценту выполнения современного задания 23 ЕГЭ по информатике [9]. Методы комбинаторики и программирования для решения рассматриваемого задания также могут быть использованы при решении современного задания 8 ЕГЭ по информатике [10].

Аналитическое решение задания (выделение типов программ, дающих одинаковый результат)

Для аналитического решения задания следует вычислить количество типов программ, содержащих ровно пять команд, причём программы внутри каждого типа должны обеспечивать одинаковый результат вычислений. Типы программ, примеры программ для каждого типа, числовой результат представлены в таблице 1. Для двух заданных ко-

манд (1. Прибавь 3; 2. Вычти 2) следовало вычислить, сколько различных чисел можно получить из числа 1 с помощью программы, которая содержит ровно 5 команд.

Таблица 1. Шесть типов программ длины 5 и шесть чисел-результатов

Тип программы	Примеры программ	Числовой результат
1. Все команды 1	11111 (всего 1 программа)	$1 + 3 + 3 + 3 + 3 + 3 = 16$
2. Четыре команды 1, одна команда 2	11112, 11121, 11211, 12111, 21111 (всего 5 программ)	$1 + 3 + 3 + 3 + 3 - 2 = 11$
3. Три команды 1, две команды 2	11122, 12121, ..., 22111 (всего 10 программ)	$1 + 3 + 3 + 3 - 2 - 2 = 6$
4. Две команды 1, три команды 2	11222, 21212, ..., 22211 (всего 10 программ)	$1 + 3 + 3 - 2 - 2 - 2 = 1$
5. Одна команда 1, четыре команды 2	12222, 21222, 22122, 22212, 22221 (всего 5 программ)	$1 + 3 - 2 - 2 - 2 - 2 = -4$
6. Все команды 2	22222 (всего 1 программа)	$1 - 2 - 2 - 2 - 2 - 2 = -9$

Каждая отдельная программа является комбинацией из приведённых команд (1. Прибавь 3; 2. Вычти 2).

Здесь и далее авторы предлагают выделить следующие компетенции для базового и углубленного уровней изучения учебного предмета «Информатика».

Компетенции на базовом уровне: выделение различных типов программ, запись примеров программ, вычисление результатов исполнения программ, понимание коммутативности (переместительного закона) для приведённых команд и основ комбинаторики.

Компетенции на углубленном уровне: подсчёт количества программ различных типов, знание и применение формул комбинаторики (вычисление количества комбинаций, перестановок, размещений, сочетаний).

Владение приведёнными компетенциями позволяет достичь перечисленных в федеральных рабочих программах планируемых результатов освоения программы по информатике на уровнях основного общего и среднего общего образования [11–14].

Описанное аналитическое решение задания на основе выделения различных типов программ использует приёмы и методы комбинаторики и может быть использовано в качестве задания для изучения основ этого раздела математики.

Аналитическое решение задания с использованием графа

Вышеприведённый метод выделения отдельных типов программ, дающих одинаковый числовой результат, не является единственным аналитическим методом решения рассматриваемого задания. Ещё

одним методом решения задания выступает графический способ (рисунок 1).

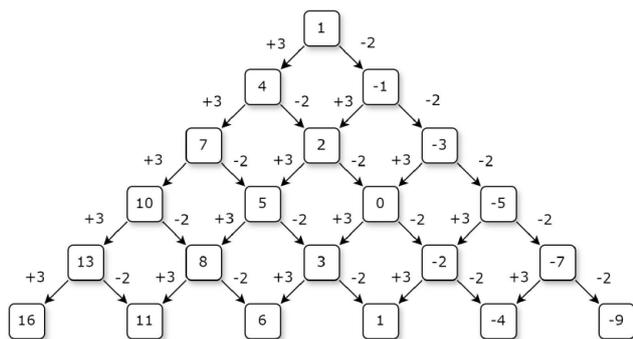


Рис. 1. Графический способ решения

На рисунке 1 корневой вершиной является число 1 – исходное число задания, листьями дерева являются все возможные результаты выполнения программ длиной в 5 команд, другие вершины дерева содержат промежуточные числа в возможных траекториях вычислений программ. Всего существует 6 конечных узлов, также называемых листьями дерева на рисунке 1, соответственно, можно получить 6 различных чисел из числа 1 с помощью программы длиной 5 команд.

Компетенции на базовом уровне: использование графов и деревьев для визуализации информации, построение дерева вручную, построение дерева с использованием готового программного обеспечения.

Компетенции на углубленном уровне: построение дерева с использованием библиотек визуализации данных языков программирования, знание и использование двоичных деревьев, включая двоичные кодовые деревья, двоичные деревья поиска, построение и чтение графа переходов конечного автомата.

Как на базовом, так и на углублённом уровне применение описанного графического способа решения задания открывает широкие возможности для изучения терминологии и методов теории графов.

Решение задания на языке программирования Python

На рисунке 2 приведён простой программный код для решения задания.

Приведённое на рисунке 2 решение опирается на использование вложенных циклов для полного перебора всех возможных комбинаций команд. В первой строке программы создаётся пустой список `results`. Представленные целочисленными переменными команды `k1`, `k2`, ..., `k5` в теле цикла максимальной вложенности позволяют к целому числу `num`, исходное значение которого всегда равно 1, или прибавить 3, или вычесть 2. После выполнения пяти команд полученный в переменной `num` числовой результат записывается в список `results` с помощью метода `append`: `results.append(num)`. После перебора всех возможных комбинаций программ длины 5 в конце программы полученный список выводится в отсортированном

виде, что позволяет легко подсчитать количество различных итоговых чисел: `print(sorted(results))`.

```

results = []
for k1 in 1, 2:
    for k2 in 1, 2:
        for k3 in 1, 2:
            for k4 in 1, 2:
                for k5 in 1, 2:
                    num = 1
                    if k1 == 1:
                        num += 3
                    elif k1 == 2:
                        num -= 2
                    if k2 == 1:
                        num += 3
                    elif k2 == 2:
                        num -= 2
                    if k3 == 1:
                        num += 3
                    elif k3 == 2:
                        num -= 2
                    if k4 == 1:
                        num += 3
                    elif k4 == 2:
                        num -= 2
                    if k5 == 1:
                        num += 3
                    elif k5 == 2:
                        num -= 2

                    results.append(num)
print(sorted(results))
  
```

Рис. 2. Решение задания при помощи программного кода

Приведённый программный код является простым для реализации, однако обладает целым рядом недостатков, основными среди которых являются дублирование фрагментов кода и отсутствие вычисленного количества различных чисел, являющихся результатами выполнения программы. Некоторые из описанных недостатков, а именно дублирующиеся условные операторы и отсутствие вычисленного ответа на задание, исправлены в программном коде на рисунке 3.

```

s = set()
for k1 in 1, 2:
    for k2 in 1, 2:
        for k3 in 1, 2:
            for k4 in 1, 2:
                for k5 in 1, 2:
                    prog = [k1, k2, k3, k4, k5]
                    num = 1
                    for k in prog:
                        if k == 1:
                            num += 3
                        elif k == 2:
                            num -= 2
                    s.add(num)
print('Количество различных чисел:', len(s))
print('Числа:', s)
  
```

Рис. 3. Оптимизированное решение задания при помощи программного кода

Как и в программном коде на рисунке 2, в программе на рисунке 3 вложенные циклы `for` производят перебор всех возможных комбинаций команд для программ длины 5. Для каждой комбинации пять команд, каждая из которых представлена в виде цифры 1 или 2, записываются в список `prog`, после чего в теле цикла `for k in prog` каждая

команда преобразует число, или прибавляя 3, или вычитая 2. Для хранения различных результатов исполнения программ используется множество `s`, позволяющее хранить неповторяющиеся элементы.

Организовать полный перебор комбинаций команд в программах без использования вложенных циклов способна функция `product` модуля `itertools` (рисунок 4).

```
from itertools import product
progs = list(product((1, 2), repeat = 5))
s = set()
for prog in progs:
    num = 1
    for k in prog:
        if k == 1:
            num += 3
        elif k == 2:
            num -= 2
    s.add(num)
print('Получено', len(s), 'чисел:', s)
```

Рис. 4. Решение задания при помощи программного кода с использованием модуля `itertools`

```
from itertools import product
print(len(set([sum(prog, 1) for prog in list(product((3, -2), repeat = 5))])))
```

Рис. 5. Один из вариантов наиболее короткого программного кода для нахождения ответа на задание

В листинге на рисунке 5 с использованием *генератора списка* для каждой возможной программы выполняется вычисление итогового числа, после чего полученный список итоговых чисел преобразуется во множество. Программа выводит количество элементов полученного множества.

Компетенции на базовом уровне: разработка переборного алгоритма на основе вложенных циклов, использование и реализация полного перебора.

Компетенции на углубленном уровне: организация перебора комбинаций с использованием готовой функциональности (например, модуля `itertools` в языке программирования Python), выбор и использование подходящих структур данных, использование генераторов списков, применение принципа DRY (“don’t repeat yourself”) и сокращение дублирования кода.

Приведённые варианты решений задания программным кодом не только используют необходимые алгоритмические конструкции и структуры данных, демонстрируют повышение уровня абстракции программ, применение принципа DRY, вариативность поиска решений, но и показывают, что на базе рассмотренного задания возможно довольно глубокое изучение и актуализация знаний раздела информатики «Алгоритмы и программирование».

Одно задание, много методов

В данной работе показано три метода решения рассмотренного задания: комбинаторное выделение типов программ, использование графа, решение компьютерной программой. Целевым результатом

В список `progs` будут присвоены кортежи всех возможных программ длины 5 для команд 1 и 2, каждый кортеж-элемент списка представляет собой комбинацию из 5 команд: [(1, 1, 1, 1, 1), (1, 1, 1, 1, 2), (1, 1, 1, 2, 1), (1, 1, 1, 2, 2), (1, 1, 2, 1, 1), (1, 1, 2, 1, 2), (1, 1, 2, 2, 1), (1, 1, 2, 2, 2), ..., (2, 2, 2, 2, 1), (2, 2, 2, 2, 2)]. Цикл `for prog in progs` итерируется по элементам списка, вложенный цикл `for k in prog`, представленный также в тексте предыдущей программы на рисунке 3, осуществляет последовательный перебор всех команд программы, представленной кортежем-элементом списка. Как и в предыдущем программном коде, для хранения всех возможных числовых результатов программ используется множество `s`, мощность которого равна количеству чисел – возможных результатов программ, описанных в задании.

Наконец, для учеников, знающих программирование на углублённом уровне, будет полезным дополнительное задание: запись решения в одну строку (рисунок 5).

обучения на уровне среднего общего образования должно стать уверенное владение обучающимися всеми описанными методами. При этом обучающиеся образовательных организаций имеют различные педагогические условия: разное количество часов учебного предмета «Информатика» в зависимости от различных учебных планов и образовательных профилей, отличающееся календарно-тематическое планирование и учебно-методическое обеспечение, педагогов с их индивидуальным опытом и стилем преподавания. По мнению авторов, использование педагогических технологий укрупнения дидактических единиц и [15–17] перевёрнутого класса [18], а также интегрированного обучения [6, 7] при рассмотрении предложенного задания способны сформировать компетенции обучающихся, находящихся в различных образовательных средах, в рассмотренных областях комбинаторики, теории графов, программирования.

Используемые методы различных разделов математики и программирования на различном уровне абстракции выявляют необходимость демонстрации предметных и межпредметных связей, формирования компетенций и компетентностей в их совокупности: математических, информационных, цифровых, а также компетенций/компетентностей [19, с. 17] в области программирования. Таким образом, для анализа результата исполнения алгоритма необходимы не только знания, навыки и деятельностные установки в области информатики, алгоритмики и программирования, но также – знания и навыки различных разделов математики и способности их эффективного практического применения.

Заключение

В статье рассмотрены разнообразные методы решения задачи вычисления количества различных результатов для заданных команд и длины программы, использованы методы и инструменты комбинаторики, теории графов, алгоритмизации и программирования, являющиеся содержанием учебного предмета «Информатика». Для полноценного изучения описанных методов рекомендуется укрупнение дидактических единиц, реализация модели перевёрнутого класса и технологий интегрированного обучения в педагогическом процессе. Разнообразие методов решения рассмотренной задачи не только отражает предметные и межпредметные связи, используемые при изучении тем раздела «Алгоритмы и программирование», но также является аргументом в пользу формирования не только отдельных компетенций/компетентностей, но их совокупности в целостном педагогическом процессе.

Литература

1. Демонстрационный вариант контрольных измерительных материалов единого государственного экзамена 2012 года по информатике и ИКТ. Москва: Федеральный институт педагогических измерений, 2011. 36 с.
2. Рябина Е.Е. Решение задач ЕГЭ с использованием алгоритмов динамического программирования // Информационно-коммуникационные технологии в педагогическом образовании. 2019. № 6(63). С. 71–78. EDN GKGDSA.
3. Чапурных А.А. Различные способы решения задания № 23 КЕГЭ на перебор вариантов и динамическое программирование // Наука, общество, образование в условиях цифровизации и глобальных изменений: сборник статей III Международной научно-практической конференции, Пенза, 20 августа 2022 года. Пенза: Наука и Просвещение (ИП Гуляев Г.Ю.), 2022. С. 44–48. EDN LAKNFW.
4. Козлов С. В., Быков А.А. Обучение школьников выполнению и анализу простейших алгоритмов управления исполнителями в среде КуМир // Современные наукоемкие технологии. 2023. № 10. С. 123–128. DOI: 10.17513/snt.39803. EDN KICPUL.
5. Босова Л. Л., Босова А.Ю. Школьная информатика в условиях цифровой трансформации общества. Москва: Московский педагогический государственный университет, 2024. 182 с. ISBN 978-5-4263-1351-4. DOI: 10.31862/9785426313514. EDN APRPXQ.
6. Глизбург В.И. Подготовка магистров педагогического образования к интегрированному обучению школьников математике и информатике // Вестник Российского университета дружбы народов. Серия: Информатизация образования. 2019. Т. 16, № 4. С. 318–327. DOI: 10.22363/2312-8631-2019-16-4-318-327. EDN KTHMVQ.
7. Самойлова Е.С. Интеграция школьных курсов информатики и математики в свете новых федеральных государственных образовательных стандартов // Вестник Российского университета дружбы народов. Серия: Информатизация образования. 2013. № 1. С. 54–59. EDN PUZBOJ.
8. Монахова Г. А., Монахов Н.В. Практикум по курсу «Содержание предмета информатика». Часть 1. Москва, 2016. 134 с. ISBN 978-5-9908269-0-8. EDN YMYOKF.
9. Попов В. С., Абросимова-Романова Л.А. Анализ рисков выпускников и выделение необходимых компетенций в связи с изменением среднего процента выполнения заданий ЕГЭ по информатике за 2019–2023 годы и введением компьютерного формата экзамена // Новые информационные технологии в образовании: Сборник научных трудов XXIV Международной научно-практической конференции, Москва, 30–31 января 2024 года. Москва: ООО «1С-Пабблишинг», 2024. С. 326–330. EDN RJIRBK.
10. Попов В. С., Леляев П.А. Способы решения задачи № 8 ЕГЭ по информатике // Потенциал: Математика, Физика, Информатика. 2021. № 2. С. 53–57.
11. Федеральная рабочая программа среднего общего образования. Информатика (базовый уровень) (для 10–11 классов образовательных организаций). Москва: Институт стратегии развития образования. 2023. 38 с.
12. Федеральная рабочая программа среднего общего образования. Информатика (углублённый уровень) (для 10–11 классов образовательных организаций). Москва: Институт стратегии развития образования. 2023. 52 с.
13. Федеральная рабочая программа основного общего образования. Информатика (базовый уровень) (для 7–9 классов образовательных организаций). Москва: Институт стратегии развития образования. 2023. 45 с.
14. Федеральная рабочая программа основного общего образования. Информатика (углублённый уровень) (для 7–9 классов образовательных организаций). Москва: Институт стратегии развития образования. 2023. 51 с.
15. Эрдниев П.М. Укрупнение дидактических единиц как технология обучения: В двух частях. Часть I. Москва: Акционерное общество «Издательство «Просвещение», 1992. 175 с. EDN YKBUYF.
16. Эрдниев П.М. Укрупнение дидактических единиц как технология обучения: В двух частях. Часть II. Москва: Акционерное общество «Издательство «Просвещение», 1992. 255 с. EDN VIRMOR.
17. Заводчикова Н. И., Быкова И.А. Укрупнение дидактических единиц при изучении метода динамического программирования в средней школе // Актуальные проблемы теории и практики обучения физико-математическим и техническим дисциплинам в современном

образовательном пространстве: IV Всероссийская (с международным участием) научно-практическая конференция, посвященная 75-летию факультета физики, математики, информатики Курского государственного университета, Курск, 16–17 декабря 2020 года. Курск: Курский государственный университет, 2020. С. 289–294. EDN GSRACM.

18. Сиротина О.А. Интерактивная технология обучения «Перевернутый класс» как средство повышения качества образования // Проблемы научно-практической деятельности. Перспективы внедрения инновационных решений: сборник статей Всероссийской научно-практической конференции, Ижевск, 17 декабря 2019 года. Часть 2. Ижевск: Общество с ограниченной ответственностью «Омега Сайнс», 2019. С. 265–268. EDN DWHDNT.
19. Зимняя И.А. Ключевые компетентности как результативно-целевая основа компетентностного подхода в образовании. Москва: Исследовательский центр проблем качества подготовки специалистов Московского государственного института стали и сплавов (технологического университета), Сектор гуманизации образования, 2004. 40 с.

COMPETENCIES FOR ANALYZING THE RESULT OF ALGORITHM EXECUTION: THE TASK OF CALCULATING THE NUMBER OF DIFFERENT RESULTS FOR GIVEN COMMANDS AND THE LENGTH OF THE PROGRAM IN THE UNIFIED STATE EXAMINATION IN COMPUTER SCIENCE

Popov V.S., Alefirenko E.A., Chernitsyna L.Yu.
Moscow City University

The article considers combinatorial, graphical, and programmatic methods for solving the task of calculating the number of different results of the described program for a formal performer. For each considered method, one or more solutions and relevant competencies for the basic and advanced levels of studying the subject “Computer Science” are given. With the combinatorial solution method, various types of programs are distinguished, the graphical solution uses a tree, the vertices of which are the numbers obtained, and the arcs are the executor’s commands, four program listings are given as solutions in the programming language, each subsequent program has a higher level of abstraction and uses more and more in-depth knowledge and programming skills. For proper study of the described subject content in various educational environments, the authors propose the use of enlarged didactic units, a flipped classroom, and integrative learning. The various methods used to solve the proposed task reveal the need to demonstrate intradisciplinary and interdisciplinary connections, the formation of competencies in their entirety.

Keywords: competence, competency, competence approach, algorithm, result of algorithm execution, calculation of the amount of numbers, Unified State Examination, Unified State Examination in computer science.

References

1. Demonstration version of control measuring materials of the 2012 Unified State Exam in Computer Science and ICT. Moscow: Federal Institute of Educational Measurement, 2011. 36 p.
2. Ryabinina E.E. Solution of USE examples using dynamic programming algorithms // Information and communication technologies in pedagogical education, 2019, No. 6 (63), pp. 71–78. EDN GKGD SA.

3. Chapurnykh A.A. Various ways to solve task № 23 of the computer Unified State Exam for overview of options and dynamic programming // Science, society, education in the context of digitalization and global changes: collection of articles from the III International scientific and practical conference, Penza, August 20, 2022. Penza: Science and Education (IP Gulyaev G. Yu.), 2022. Pp. 44–48. EDN LAKNFV.
4. Kozlov S. V., Bykov A.A. Training schoolchildren on the implementation and analysis of the simplest algorithms for managing performers in the Kumir environment // Modern high technologies, 2023, No. 10, pp. 123–128. DOI: 10.17513/snt.39803. EDN KICPUL.
5. Bosova L. L., Bosova A. Yu. School informatics in the context of digital transformation of society. Moscow: Moscow State Pedagogical University, 2024. 182 p. ISBN 978-5-4263-1351-4. DOI: 10.31862/9785426313514. EDN APRPXQ.
6. Glizburg V.I. The training of masters of pedagogical education to the integrated teaching of the schoolchildren in mathematics and computer science // RUDN Journal of Informatization in Education, 16(4), pp. 318–327. DOI: 10.22363/2312-8631-2019-16-4-318-327. EDN KTHMVQ.
7. Samoilova E.S. The integration of school courses of informatics and mathematics in the light of the new federal government educational standards // RUDN Journal of Informatization in Education, 2013, No. 1, pp. 54–59. EDN PUZBOJ.
8. Monakhova G. A., Monakhov N.V. Workshop on the course “Content of the Informatics Subject”. Part 1. Moscow, 2016. 134 p. ISBN 978-5-9908269-0-8. EDN YMYOKF.
9. Popov V. S., Abrosimova-Romanova L.A. Analysis of graduates’ risks and identification of necessary competencies in connection with the change in the average percentage of completing Unified State Exam tasks in Computer Science for 2019–2023 and the introduction of a computer exam format // New Information Technologies in Education: Proceedings of the 24th International Scientific and Research Conference “New Information Technologies in Education” (1C technologies for the data economy development and hyperautomation of education) January 30–31, 2024. Part 1. M. LLC “1C-Publishing” in 2024. Pp. 326–330. EDN RJIRBK.
10. Popov V. S., Lelyaev P.A. Methods for Solving Problem No. 8 of the Unified State Exam in Computer Science // Potential: Mathematics, Physics, Computer Science, 2021, No. 2, pp. 53–57.
11. Federal work program of secondary general education. Computer science (basic level) (for grades 10–11 of educational organizations). Moscow: Institute for Strategy of Education Development, 2023. 38 p.
12. Federal work program of secondary general education. Computer science (advanced level) (for grades 10–11 of educational organizations). Moscow: Institute for Strategy of Education Development, 2023. 52 p.
13. Federal work program of basic general education. Computer science (basic level) (for grades 7–9 of educational organizations). Moscow: Institute for Strategy of Education Development, 2023. 45 p.
14. Federal work program of basic general education. Computer Science (Advanced Level) (for Grades 7–9 of Educational Organizations). Moscow: Institute for Strategy of Education Development, 2023. 51 p.
15. Erdniev P.M. Consolidation of Didactic Units as a Teaching Technology: In Two Parts. Part I. Moscow: Joint-Stock Company “Prosveshchenie Publishing House”, 1992. 175 p. EDN YK-BUYF.
16. Erdniev P.M. Consolidation of Didactic Units as a Teaching Technology: In Two Parts. Part II. Moscow: Joint-Stock Company “Prosveshchenie Publishing House”, 1992. 255 p. EDN VIR-MOR.
17. Zavodchikova N. I., Bykova I.A. Enlargement of didactic units in the study of the dynamic programming method in high school // Actual problems of the theory and practice of teaching physical, mathematical and technical disciplines in the modern educational space: IV All-Russian (with international participation) scientific and practical conference dedicated to the 75th anniversary of the Faculty of Physics, Mathematics, Informatics of Kursk State University, Kursk, December 16–17, 2020. Kursk: Kursk State University, 2020. Pp. 289–294. EDN GSRACM.
18. Sirotnina O.A. Interactive learning technology “Flipped Classroom” as a means of improving the quality of education // Prob-

lems of scientific and practical activity. Prospects for the implementation of innovative solutions: collection of articles of the All-Russian scientific and practical conference, Izhevsk, December 17, 2019. Part 2. Izhevsk: Omega Science LLC, 2019. Pp. 265–268. EDN DWHDNT.

19. Zimnyaya I.A. Key competencies as a result-target basis for a competence-based approach in education. Moscow: Research Center for Problems of Quality of Specialist Training, National University of Science and Technology «MISIS», Sector for Humanization of Education, 2004. 40 p.