

Розов Константин Владимирович

*Аспирант, ассистент кафедры Информатики и дискретной математики,
Новосибирский государственный педагогический университет, г. Новосибирск.
E-mail: konstantin_dubrava@mail.ru*

Розова Мария Сергеевна

*Магистрант, Новосибирский государственный педагогический университет,
г. Новосибирск. E-mail: rozovams03@gmail.com*

**ФОРМИРОВАНИЕ И РАЗВИТИЕ АЛГОРИТМИЧЕСКОЙ
КУЛЬТУРЫ В РАМКАХ УЧЕБНОГО ПРЕДМЕТА
«ИНФОРМАТИКА»**

Статья посвящена проблеме оценки сформированности алгоритмической культуры у обучающегося в рамках учебного предмета «Информатика» в связи с противоречием между возможным формальным наличием у него знаний всех компонентов алгоритмической культуры и одновременным отсутствием ряда признаков понимания этих компонентов, что, на наш взгляд, не позволяет охарактеризовать учащегося как обладающего высоким уровнем алгоритмической культуры.

Цель статьи – расширить множество признаков сформированности алгоритмической культуры в рамках учебного предмета «Информатика».

Проведен ретроспективный анализ понятия алгоритмической культуры, его становление и место в системе целей обучения информатике в школе, а также основные направления научных исследований, связанных с алгоритмической культурой. Рассматриваются основные компоненты алгоритмической культуры, понятия алгоритма и исполнителя.

Предлагается авторское понятие «концептуальные признаки алгоритмической культуры». Приводятся ключевые принципы, понимание которых обучающимся может свидетельствовать о высоком уровне сформированности его алгоритмической культуры. Показана взаимосвязь алгоритмической культуры и культуры программирования.

Ключевые слова: алгоритмическая культура, алгоритмизация и программирование, компьютерная грамотность, культура программирования, обучение информатике, оценка качества образования.

Rozov Konstantin Vladimirovich

*Postgraduate Student, Assistant of the Department of Informatics and Discrete
Mathematics, Novosibirsk State Pedagogical University, Novosibirsk.
E-mail: konstantin_dubrava@mail.ru*

Rozova Maria Sergeevna

*Master student, Novosibirsk State Pedagogical University, Novosibirsk.
E-mail: rozovams03@gmail.com*

**FORMATION AND DEVELOPMENT OF ALGORITHMIC CULTURE
IN THE FRAMEWORK OF THE EDUCATIONAL SUBJECT
"COMPUTER SCIENCE"**

The article is devoted to the problem of assessing the formation of an algorithmic culture in a student within the framework of the subject "Computer Science" in connection

with the contradiction between the possible formal knowledge of all components of the algorithmic culture and the simultaneous absence of a number of signs of understanding of these components, which, in our opinion, does not allow to characterize the student as having a high level of algorithmic culture.

The objective of the article is to expand many signs of the formation of an algorithmic culture in the framework of the subject “Computer Science”.

A retrospective analysis of the concept of algorithmic culture, its formation and place in the system of goals for teaching computer science at school, as well as the main directions of scientific research related to algorithmic culture, are carried out. The main components of the algorithmic culture, the concepts of the algorithm and the executor are considered.

The author offers the concept of “conceptual signs of an algorithmic culture”. Key principles are given, the understanding of which for students can indicate a high level of formation of its algorithmic culture. The interrelation of algorithmic culture and programming culture is shown.

Keywords: algorithmic culture, algorithmization and programming, computer literacy, programming culture, computer science training, education quality assessment.

В начале 70-х годов М. П. Лапчик ввел понятие «Алгоритмическая культура» в педагогическую теорию и практику. В своем диссертационном исследовании «Использование общеобразовательных аспектов программирования для ЭВМ в совершенствовании среднего математического образования» он раскрыл содержание, а также основные компоненты понятия алгоритмическая культура [5].

В общем виде понятие алгоритмическая культура понимается как «совокупность специфических представлений, умений и навыков, связанных с овладением наиболее общими компонентами алгоритмизации, которое на современном этапе развития общества начинает рассматриваться как естественное требование к части общей культуры каждого человека и, следовательно, определять целенаправленный компонент общего школьного образования» [4, с. 32].

В период начала развития понятия алгоритмическая культура его рассматривали в рамках курса математики, в котором происходило знакомство обучающихся с алгоритмизацией. В настоящее время алгоритмическая культура по-прежнему рассматривается преимущественно по отношению к математической подготовке.

Рассматривая особенности формирования алгоритмической культуры при

решении учебных математических задач О. К. Шевченко, Н. В. Измайлова, Н. Л. Борисова справедливо утверждают, что «для усвоения математических знаний недостаточно только одной их формулировки, а необходимо объяснение правила на примере решения задачи, выполнение ряда упражнений на применение правила, ибо основной формой усвоения математики является математическая деятельность – решение задач» [14, с. 241]. Мы считаем, что для формирования алгоритмической культуры в рамках учебного предмета «Информатика» так же недостаточно изучения формулировок, важна именно деятельность – составление алгоритмов, программирование, и понимание ряда принципов для успешного освоения этой деятельности.

Главным фактором, определившим пути развития понятия «алгоритмическая культура», стало введение в учебный план отечественной школы обязательного общеобразовательного предмета «Основы информатики и вычислительной техники» (далее – ОИВТ).

С учетом содержания первых пробных учебных пособий по курсу ОИВТ М. П. Лапчиком в первом отечественном учебном пособии для студентов педагогических вузов по курсу «Методика

преподавания информатики» были выделены компоненты понятия алгоритмической культуры обучающихся, а также показаны роль и место данного понятия в системе целей обучения информатике в школе.

Компоненты алгоритмической культуры, описанные М. П. Лапчиком [6]:

1. понятие алгоритма и его свойства;
2. понятие языка описания алгоритмов;
3. уровень формализации описания;
4. принцип дискретности (пошаговости) описания;
5. принцип блочности;
6. принцип ветвления;
7. принцип цикличности;
8. выполнение алгоритма;
9. организация данных.

Компоненты алгоритмической культуры образуют ядро понятия «компьютерная грамотность». Развитие у обучающихся компьютерной грамотности относилось к числу главных задач обучения информатики в первые годы существования общеобразовательного предмета ОИВТ.

Включение вопросов, связанных с алгоритмизацией, в содержание понятия компьютерной грамотности можно рассматривать в связи с отчетливо проявившимся переосмыслением целей обучения информатике в школе. Так, ведущими специалистами в области информатизации образования обосновывается необходимость перехода от обеспечения компьютерной грамотности, часто трактовавшейся как некая сумма знаний и умений, непосредственно связанных с работой на компьютере, к полноценному образованию в области информатики, основу которого должно составлять изучение всех аспектов процессов обработки информации [6].

Л. Л. Босова отмечает, что «В соответствии с ФГОС основного общего образования можно выделить несколько ключевых направлений обучения информатике ... Второе направление (формирование алгоритмической куль-

туры обучающихся) включает изучение таких вопросов как алгоритм и его свойства, логические значения и операции, алгоритмические конструкции и запись с их использованием алгоритмов для конкретного исполнителя; знакомство с одним из языков программирования» [3, с. 332].

В процессе развития информатики как научного направления и учебной дисциплины понятие «алгоритмическая культура» не изменилось.

Научные исследования, связанные с алгоритмической культурой, можно разделить на три группы.

Для первой группы характерно обнаружение и формирование значимых навыков, умственных действий человека, работа которого связана с составлением и применением алгоритмов, а также алгоритмических предписаний при решении задач. В данной группе намечается целенаправленное проявление алгоритмической линии, первая стадия которой – формирование составляющих алгоритмической культуры (М. П. Лапчик, Л. Г. Лучко, В. М. Оксман, Л. С. Оксман, А. А. Шрайнер, Б. А. Шрайнер и др.) [7; 8; 10; 14; 15].

Во второй группе рассматривается проблема формирования у обучающихся понятия «алгоритм» как одного из главных составляющих в области математики (В. А. Байдак, Ю. А. Макаренко, А. А. Столяр и др.) [1; 9; 12].

В третьей группе вопрос о формировании алгоритмической культуры обучающихся не является специальным предметом исследования. Алгоритмы и алгоритмические предписания применяются исключительно для достижения частных-дидактических целей (Л. И. Боженкова и др.) [2].

Во всех группах педагогических исследований основным считается определение «алгоритм», описанное математиком Аль Хорезми в IX веке н. э. при создании правил для 4-х арифметических действий.

Для понимания сущности алгоритмической культуры необходимо рассмотреть основные компоненты алгоритмической культуры: понятия «алгоритм» и «исполнитель».

«Под алгоритмом понимают понятное и точное предписание (указание) исполнителю совершить последовательность действий, направленных на достижение указанной цели или на решение поставленной задачи» [11].

Из анализа современных учебных пособий по информатике можно сделать вывод о том, что первое определение, представленное в пробном учебном пособии по информатике, по-прежнему остается корректным и методически точным. Очень важной в данном определении является связь понятия «алгоритм» с понятием «исполнитель», для которого составляется алгоритм.

Исполнитель – это обобщенное понятие для описания любого исполнительного устройства или объекта, который может исполнять некоторые команды. Таким объектом может быть человек или животное (неформальные исполнители), некоторое техническое устройство, робот, компьютер, какое-либо периферийное устройство компьютера, система программирования, операционная система, инструментальное программное средство (формальные исполнители). Формальные исполнители, в отличие от неформальных, одну и ту же команду всегда исполняют одинаково. Именно на работе с формальными исполнителями основано изучение алгоритмизации и программирования в школьном курсе информатики.

Основными характеристиками исполнителя являются среда (обстановка), в которой он находится, определяющая условия его функционирования, и система команд.

Система команд исполнителя – это совокупность всех команд, которые исполнитель может выполнять. Для кон-

кретного исполнителя команды из его системы команд могут задаваться самым различным образом: в текстовом виде, в виде пиктограмм и т. д. При построении алгоритма необходимо учитывать условия применимости тех или иных команд, а также общие свойства алгоритма: дискретность, понятность, определенность, массовость и результативность.

Однако знание основных компонентов алгоритмической культуры, описанных выше, на наш взгляд, не может свидетельствовать о том, что человек в полной мере обладает алгоритмической культурой.

Для характеристики уровня сформированности алгоритмической культуры введём понятие *концептуальный признак* алгоритмической культуры. Под концептуальным признаком алгоритмической культуры будем понимать признак её сформированности, отсутствие которого позволяет опровергнуть сформированность алгоритмической культуры при наличии знания определений ключевых понятий «алгоритм» и «исполнитель», свойств, видов алгоритма и способов его записи, характеристик исполнителя, умения записать алгоритмы базовых видов (линейный, разветвляющийся, циклический) по крайней мере на одном алгоритмическом языке.

Концептуальными признаками сформированности алгоритмической культуры обучающегося являются факты понимания им следующих принципов:

- Формальный исполнитель будет выполнять команды буквально, именно так, как они написаны, а не так, как возможно хотелось бы составителю алгоритма при неверном понимании назначения команд, либо неверном построении логики решения задачи.

Внешним проявлением непонимания данного принципа у обучающегося выступает обвинение исполнителя в том, что тот неверно выполняет алгоритм.

Выражается это часто в высказываниях вида: «делает что-то не то», «не понимает, куда ему идти», «я говорю ему делать так, а он делает по-другому» и т. п. Однако при этом стоит учитывать характер высказываний, т. е. являются ли они демонстрацией искреннего недоумения обучающегося при его полной уверенности в правильности своего алгоритма, или сказаны в шутку.

– Составленный алгоритм в общем случае направлен на обработку какой-либо информации, которую необходимо где-то хранить. При этом может возникнуть необходимость также в хранении внутренних промежуточных и полученных извне данных в результате работы алгоритма.

Понимание данного принципа обучающийся демонстрирует, осознанно добавляя в алгоритм переменные, не являющиеся обязательными для функционирования какого-либо оператора (например, цикла со счетчиком) еще до полного описания алгоритма. Этому предшествует понимание необходимости использования переменных в принципе для обеспечения массовости алгоритма.

– Данные, обрабатываемые алгоритмом, могут быть различного типа (числовые, строковые и др.) и иметь различную структуру (простые или сложные/составные). Методы работы с одними типами данных могут отличаться от методов работы с другими.

Определить понимание данного принципа у обучающегося проще всего при работе со строго типизированными языками программирования. Оно выражается в отсутствие логических ошибок, связанных с типами данных, ошибок несоответствия и преобразования типов, корректности вывода информации различных типов в выражениях.

– Описание алгоритмов на языках одного класса с помощью команд, не являющихся специфичными для этих язы-

ков, осуществляется похожим образом. Это означает, что зная, как описать алгоритм на языке А, можно описать его на другом языке В того же класса, с учетом правил этого языка. Либо, имея готовое описание алгоритма или его фрагмента, записанного на языке В, можно перевести его на язык А, проведя аналогию. При этом, что особенно важно, не обязательно иметь опыт описания алгоритмов на языке В.

Это значит, что обладающий высоким уровнем сформированности алгоритмической культуры обучающийся, следуя данному принципу, должен быть способен понять алгоритм, записанный на неизвестном ему языке программирования (языке, опыта программирования на котором у обучающегося нет), адаптировать описание алгоритма для своей собственной программы, при условии, что оба языка принадлежат одному классу (например, оба языка являются императивными), либо определить, что адаптация невозможна. При решении задания ЕГЭ по информатике, в котором представлены фрагменты кода на нескольких языках программирования, такому обучающемуся будет достаточно одного любого из них для верного решения поставленной задачи.

Как точно заметила С. Е. Царева, «Алгоритм в информатике – базовое понятие, лежащее в основе программирования» [13, с. 155]. Поэтому стоит упомянуть признаки *культуры программирования*, взаимосвязанные с алгоритмической культурой. Культура программирования по отношению к обучающимся школ, средних специальных и высших учебных заведений, в которых не ставится цель подготовки профессиональных программистов, однако программирование является обязательной частью учебного плана и может напрямую влиять на качество подготовки будущего педагога (педагогические колледжи и вузы), по нашему мнению, выражается в следующих принципах.

– Описание алгоритма на языке программирования или в форме псевдокода должно быть как можно более читаемым и понятным, информативным, причем не только для самого автора. Это обеспечивается, прежде всего, строгим соблюдением общепринятых соглашений (стандартов) о правилах оформления кода.

Выравнивание кода, использование имен, раскрывающих назначение именованной сущности – основные правила, нарушаемые обучающимися в процессе самостоятельного написания их первых компьютерных программ.

– Программы, предусматривающие взаимодействие с пользователем, должны содержать вывод сообщений в той или иной форме, понятной целевому пользователю, поясняющих текущее состояние программы (ожидание ввода выходных данных определенного типа, ожидание завершения процесса, демонстрация выходных данных и др.).

Следуя данному принципу, в программе нужно организовать условия для четкого понимания пользователем, каких действий с его стороны требует программа и какой результат от взаимодействия с ней он получает.

– Программы, обеспечивающие ввод данных для обработки некоторой системой, должны обеспечивать корректность входных данных.

Корректность ввода может быть достигнута блокированием возможности ввода нежелательных данных, либо выводом сообщений о некорректности

входных данных после завершения ввода в результате вычислений. Блокирование ввода подходит преимущественно для исключения ошибок несоответствия типов при дальнейших вычислениях. Например, когда пользователь вводит букву, а ожидалось число. Вывод сообщений о некорректности данных предотвращает вычисление неверного результата, полученного из изначально неверных выходных данных. Так, например, при написании программы для вычисления площади треугольника по трем сторонам, обучающийся должен организовать проверку на существование треугольника после ввода размеров сторон, либо задать вопрос о необходимости реализации такой функции программы в данный момент.

Таким образом, знание основных компонентов алгоритмической культуры является необходимым, но не достаточным для того, что охарактеризовать человека как обладающего высоким уровнем сформированности алгоритмической культуры. Так же является необходимым, но не достаточным и наличие приведенных в статье авторских признаков сформированности алгоритмической культуры без знания ключевых понятий и навыков записи алгоритмов на алгоритмическом языке. Компоненты и концептуальные признаки алгоритмической культуры должны в равной степени влиять на оценку её сформированности у человека.

Список литературы

1. *Байдак В. А.* Преемственные связи формирования алгоритмической культуры учащихся в обучении математике: методические рекомендации для учителей и студентов физ.-мат. факультетов. – Омск: Изд-во ОГПИ, 1985. – 33 с.
2. *Боженкова Л. И.* Алгоритмический подход в обучении геометрии учащихся общеобразовательной школы: дис. ... канд. пед. наук. – М., 1990. – 226 с.
3. *Босова Л. Л.* О курсе информатики в российской школе // Известия кыргызской академии образования. – Бишкек: КАО – № 3. – С. 332–335.
4. *Лапчик М. П.* Информатика и информационные технологии в системе общего и профессионального образования: монография. – Омск: Изд-во ОмГПУ, 1999. – 294 с.

5. *Лапчик М. П.* Использование общеобразовательных аспектов программирования для ЭВМ в совершенствовании среднего математического образования: автореф. дис. ... канд. пед. наук: 13.00.02. – М.: НИИ СиМО АПН СССР, 1974. – 26 с.

6. *Лапчик М. П., Лучко Л. Г.* Содержание учебной дисциплины «Методика преподавания информатики» // Информатика и английский язык: сборник организационно-методических материалов для специальности педвуза. – Омск: Изд-во ОГПИ, 1993. – С. 79–86

7. *Лапчик М. П.* Методика преподавания информатики: учеб. пособие для студ. пед. вузов / М. П. Лапчик, И. Г. Семакин, Е. К. Хеннер; под общей ред. М. П. Лапчика. – М.: Академия, 2001.

8. *Лучко Л. Г.* Формирование алгоритмической культуры учащихся в процессе обучения базовому курсу информатики: дис. ... канд. пед. наук: 13.00.02. – Омск, 1999. – 152 с.

9. *Макаренков Ю. А., Столяр А. А.* Что такое алгоритм? Беседы со старшеклассником. – М.: Нар. асвета, 1989. – 127 с.

10. *Оксман В. М., Оксман Л. С.* О развитии алгоритмической культуры младших школьников в условиях компьютеризации // Новые исследования в педагогических науках. – 1990. – № 2 (56) – С. 52–55.

11. *Основы информатики и вычислительной техники: Проб, учеб. пособие для сред. учеб. заведений: в 2-х ч. – Ч. 1 / под ред. А. П. Ершова, В. М. Монахова. – М.: Просвещение, 1985. – 96 с.*

12. *Столяр А. А.* Педагогика математики: учеб. пособие для физ.-мат. фак. пед. интов. – Мн.: Выш. шк., 1986. – 414 с.

13. *Царева С. Е.* Формирование и развитие алгоритмической культуры в процессе обучения математике в основной школе // Актуальные проблемы обучения математике в школе и вузе: межвузовский сборник научных трудов. Посвящается 145-летию МПГУ. – М.: МПГУ, 2017. – С. 155–159.

14. *Шевченко О. К.* Формирование алгоритмической культуры курсантов при решении учебных математических задач / О. К. Шевченко, Н. В. Измайлова, Н. Л. Борисова // Мир образование – образование в мире. – 2017. – № 3 (67). – С. 239–245.

15. *Шрайнер А. А.* Повышение качества математического образования учащихся посредством формирования и развития их алгоритмической культуры: автореф. дис. ... канд. пед. наук: 13.00.02. – Новосибирск, 1997. – 18 с.

16. *Шрайнер Б. А.* Особенности формирования элементов алгоритмической культуры младших школьников // Начальное образование: проблемы и решения: сборник научных статей I Международной научно-практической конференции. – Наманган: НамГУ, 2018. – С. 26–32.