

*Гурбанныязов Бекдурды
старший преподаватель,
Кульджаев Батыр
преподаватель,
кафедры информационных технологий
Туркменского национального института
мировых языков имени Довлетмаммеда Азади
Ашхабад, Туркменистан.*

ПРОБЛЕМЫ В ОБУЧЕНИИ ПРОГРАММИРОВАНИЮ

Аннотация: В этой статье рассматриваются существенные проблемы в области обучения программированию студентов.

Ключевые слова: программирование, курсы, образование, новички, проблемы.

*Gurbannuuzov Bekdurdy
Senior lecturer,
Kuljayev Batyr
is a teacher Departments of Information Technology
Turkmen National Institute of
world languages named after Dovletmammet Azady
Ashgabat, Turkmenistan.*

PROBLEMS IN LEARNING PROGRAMMING

Abstract: This article discusses significant problems in the field of teaching programming to students.

Keywords: programming, courses, education, beginners, problems.

Компьютерные программы присутствуют практически во всех аспектах современного бизнеса и других аспектах повседневной жизни. Разработка и сопровождение этих программ имеет жизненно важное значение, и для этого требуется довольно большое количество профессионалов в области программирования, обладающих глубокими знаниями о концепциях программирования. Однако было отмечено, что образование в этой области сопряжено со многими повторяющимися проблемами и трудностями, с которыми новички в программировании сталкиваются во время учебы. Этот факт приводит к относительно высокому проценту неудач, которые в течение многих лет создают негативное мнение и страх перед посещением курсов программирования.

Одна из самых больших проблем для новичков в программировании заключается в том, что существует огромный разрыв между интуитивным способом их мышления и способом мышления, подходящим для

компьютеров. Человеческий разум намного более развит, чем любой компьютер. Он работает таким образом, что способен обрабатывать огромное количество связей и ассоциаций, чтобы что-то сделать или понять. Компьютеры не могут этого сделать. Им нужен четкий путь, четкие границы и охват всех возможных сценариев.

Главная способность, которой не хватает компьютерам, - это способность к аналогии, ассоциации и адаптации. В то время как люди обладают этими характеристиками, компьютеры - нет, и их приходится инструктировать механически с безупречной точностью и правилами, которые охватывают все случаи, с которыми, как ожидается, будет иметь дело компьютер. В принципе, можно просто сказать, что люди разумны, а компьютеры - нет, и это главное различие, которое вызывает столкновение между интуитивным способом рассуждения, который используют новички в программировании, и способом мышления, необходимым для написания правильных компьютерных программ. Есть три класса распространенных ошибок в понимании концепций программирования среди новичков, которые были выявлены:

- Ошибка параллелизма.
- Ошибка интенциональности.
- Ошибка эгоцентризма.

Ошибка параллелизма означает ошибочное понимание того, что компьютер может знать о нескольких строках программирования одновременно. Например, компьютер может вернуться назад и выполнить какое-либо конкретное условие после того, как его условия были выполнены, независимо от его бездействия в качестве строки программирования, которая была пройдена и завершена.

Ошибка интенциональности означает, что новички в программировании часто предполагают, что будет делать программа, основываясь только на части ее кода. Они часто видят что-то, что приводит к некоторому выводу о том, что будет делать программа, и они воспринимают этот вывод как факт, поэтому интерпретируют остальную часть кода не объективно, а скорее в свете своего сформированного вывода.

Ошибка эгоцентризма означает, что новички в программировании часто не дают компьютеру достаточного количества инструкций по программированию, потому что они предполагают, что компьютер каким-то образом поймет, чего они хотят, независимо от написанного ими кода. В таком состоянии ума новички часто пропускают различные важные условия или циклы.

Чтобы сделать программирование более подходящим для начинающих программистов, правильным курсом действий было бы изменить существующую методологию и структуру учебной программы, чтобы сделать программирование более подходящим для стиля обучения среднего студента и желаемого темпа. Другим направлением также могло бы стать повышение мотивации студентов путем разъяснения важности программирования для их профессиональной карьеры.

Можно сделать вывод, что для того, чтобы попытаться решить проблему новичков в программировании, в обучение программированию можно было бы включить несколько шагов. Предлагаемые шаги следующие:

- Ввести дополнительный курс программирования перед вводным курсом программирования, который способствовал бы алгоритмическому мышлению.
 - Повысить мотивацию студентов к изучению программирования.
 - Объяснить студентам, что программирование - это навык, а не просто знание.
 - Внедрить элементы конструктивизма в учебный процесс.
 - Познакомьте с обучением на примере.
 - Познакомьте с анимацией и другими методами визуализации в сочетании с взаимодействием.
 - Внедрите интерактивное визуальное моделирование.
 - Включите поддержку нескольких стилей обучения.

Из-за существующего разрыва между интуитивным способом мышления и алгоритмическим мышлением, необходимым для разработки надлежащих компьютерных программ, было бы полезно ввести еще один курс программирования, который учитывал бы это различие и позволял студентам обучаться этому новому способу рассуждения.

Профессионалы в области программирования являются жизненно важной частью современного делового мира, и обучение новичков программированию имеет огромное значение. Курсы программирования являются неотъемлемой частью всех курсов по информатике. Однако сообщается о довольно высоком проценте неудач и постоянных проблемах новичков в программировании с пониманием концепций и структур программирования, что приводит к выводу, что научиться программировать - сложная задача, требующая больших усилий как от студентов, так и от преподавателей. Абстрактный характер концепций программирования - это то, с чем учащиеся не привыкли иметь дело, и это приводит к тому, что учащиеся не могут иметь четкого представления об

этих концепциях. Студенты часто перестают посещать лекции, потому что в какой-то момент они сбиваются с пути, что в результате снижает их мотивацию и увеличивает их страх иметь дело с программированием.

Литература:

1. М. Захарова, Мир педагогики и психологии, 6, 47 (2020).
2. Алаутинен, С. & Смоландер, К. (2010). Самооценка учащихся в курсе программирования. стр. 155-159
3. Л. Лелу, Блокчейн от А до Я. Все о технологии десятилетия (Эксмо, 2018).
4. Хави, Н. (2010). Причинно-следственные связи успеха и неудачи, сделанные студентами бакалавриата на вводном курсе компьютерного программирования. Компьютеры и образование, Том 54, май 2010 г.
5. С. Шелковникова, Е. Краснова, Р. Аведова, 14-я Международная технологическая конференция, 2014.