

УДК 608:378.147

UDC 608:378.147

**ИСПОЛЬЗОВАНИЕ ИНСТРУМЕНТОВ ТРИЗ В ОБУЧЕНИИ  
ПРОГРАММИРОВАНИЮ  
USING TRIZ INSTRUMENTS IN TEACHING PROGRAMMING**

*Гаврилова В.В., Шакун В.А., студенты гр. 814301*

*Белорусский государственный университет информатики и радиоэлектроники,  
г. Минск, Республика Беларусь*

*Князькова В.С. – старший преподаватель*

*Gavrilova V.V., Shakun V.A.*

*Belarusian State University of Informatics and Radioelectronics,  
Minsk, Republic of Belarus*

**Аннотация.** ТРИЗ – теория решения изобретательских задач, которая создавалась для решения сложных технических, конструкторских задач. Однако основы теории универсальны и могут применяться для решения различных задач в целом. Мы изучили и рассмотрели инструменты ТРИЗ с целью усовершенствования процесса обучения программированию. Ознакомившись с имеющимися инструментами ТРИЗ и выделив проблемы, существующие в преподавании программирования, мы выявили подходы, которые помогут исправить их. Исходя из наших решений, провели урок с учениками школы и получили положительные результаты.

**Ключевые слова.** ТРИЗ, использование инструментов ТРИЗ, использование инструментов ТРИЗ в обучении программированию, обучение программированию, цель ТРИЗ-педагогике, задачи ТРИЗ, принципы ТРИЗ, решение задач с помощью ТРИЗ, составление «волшебного экрана».

**Annotation.** TRIZ stands for the “Theory of Inventive Problem Solving”, which was created to solve complex technical and design problems. However, the foundations of the theory are universal and can be applied to solve various problems in general. We studied and reviewed TRIZ tools in order to improve the process of teaching programming. Having familiarized ourselves with the available TRIZ tools and highlighting the problems that exist in teaching programming, we have identified approaches that will help to fix them. Based on our decisions, we conducted a lesson with the students of the school and received positive results.

**Keywords.** TRIZ, the use of TRIZ tools, the use of TRIZ tools in teaching programming, teaching programming, the goal of TRIZ pedagogy, TRIZ tasks, TRIZ principles, solving problems using TRIZ, drawing up a "magic screen".

ТРИЗ, или теория решения изобретательских задач — набор методов решения задач и усовершенствования систем, в основе которых лежит креативный подход [1]. ТРИЗ ставит задачу: без сплошного перебора вариантов выйти сразу на так называемые сильные решения. Соответственно, ТРИЗ-педагогика призвана сделать процесс мыслительной деятельности целенаправленным, осознанным и управляемым.

К основным принципам ТРИЗ относятся:

принцип объективности законов развития систем – строение, функционирование и развитие систем подчиняются объективным законам, таким как общим законам природы. Сильные решения – это решения, соответствующие объективным законам и явлениям;

принцип противоречия – под воздействием внешних и внутренних факторов возникают и разрешаются противоречия. Системы развиваются, преодолевая противоречия. Сильные решения – это решения, преодолевающие противоречия;

принцип конкретности – каждый класс систем имеют свои особенности, облегчающие или затрудняющие изменение этой системы. Сильные решения – это решения, учитывающие индивидуальные особенности систем [2].

Цель ТРИЗ-педагогике – развитие гибкого мышления и фантазии, способности решать сложные задачи изящными и эффективными способами [3].

В процессе решения ТРИЗ-задач используется сразу несколько подходов:

метод мозгового штурма;

синектика: сравнение и нахождение сходства в предметах и явлениях;

морфологический анализ: выявление всех возможных способов решения;

метод фокальных объектов: установление ассоциативных связей с различными объектами, и т.д. [4].

Наиболее важные понятия ТРИЗ следующие:

противоречие. Система должна обладать свойством А, чтобы повысить эффективность, и не должно обладать этим свойством, чтобы не навредить. Требуется точная формулировка противоречия, т.к. она позволит понять корень проблемы;

ресурсы. Вначале пробуем использовать те ресурсы, которые уже есть. Для нас важны информационные ресурсы, ресурсы времени и человеческие ресурсы;

идеальный конечный результат (ИКР). Решать задачу проще, когда сформулирован идеальный ответ.

В итоге алгоритм решения задачи с помощью методов ТРИЗ вырисовывается следующий:

определяем тип задачи: изобретательская, исследовательская, конструкторская или прогнозная;

формулируем к задаче противоречие и идеальный конечный результат. Содействуем обострению проблемы и продвигаемся к сильным решениям;

выявляем ресурсы. Сначала рассматриваем внутренние, далее дешевые, и в конце остальные;

применяем приёмы и принципы решения задач, если до этого пункта решение ещё не появилось;

анализируем решение с позиции идеальности.

Поскольку ни один из названных принципов и алгоритмов не отвергается ни информатикой вообще, ни программированием в частности, пробуем применить их в этой области. Авторами

статьи в феврале-марте 2021 года было проведено исследование на занятиях по изучению программирования для детей младшего и среднего школьного возраста в школе "Айтигенио". Цель исследования – улучшить и ускорить обучение программированию при помощи инструментов ТРИЗ.

Гипотеза исследования: инструменты ТРИЗ помогают улучшить и ускорить обучение программированию.

Задачи, которые были поставлены:

- изучить существующие инструменты ТРИЗ;
- обнаружить проблемы в преподавании программирования;
- исправить проблемы при помощи инструментов ТРИЗ;
- применить на практике выделенные решения проблем.

В преподавании программирования мы выявили следующие проблемы:

- сложность в объяснении теории и алгоритмов решения задач;
- отсутствие визуализации работы алгоритмов;
- сложность восприятия сухой теории в книгах;
- неправильный подход к преподаванию программирования: нет диалога со студентами, чтение теории «с листика», отсутствие квалификации преподавателя и др., малое количество уделенного времени и внимания каждому учащемуся;
- недостаточное количество практики.

Изучение имеющихся инструментов ТРИЗ позволило выявить подходы, которые помогут исправить эти проблемы:

«волшебный экран» для объяснения теории (таблица 1);

построение диалога с учащимися, направление их к правильному ответу в процессе объяснения теории и алгоритма решения задач, а также, что немаловажно – просить задавать вопросы. Создание коротких анимированных видео на различные темы в программировании с наглядной демонстрацией работы алгоритмов;

контроль знаний преподавателем, повышение квалификации, проведение конференций между преподавателями, лимитированное количество человек на занятии, составление правильного графика, при котором практические занятия проводятся после изучения теоретического материала;

закрепление знаний на практике: демонстрация процесса программирования, параллельное объяснение действий преподавателя (с чего начинаем, посредством каких инструментов, сравнение подходов к решению задачи), выдача дополнительных видеуроков и домашнего задания с последующей обратной связью.

Таблица 1

«Волшебный экран» для объяснения темы «Массивы»

<p>Прошлые надсистемы: любые места для хранения, например, ящики для хранения, картотека в библиотеке</p>	<p>Родовое понятие: "Структура данных" По языкам использования: C++, C#, Java, и т.д.</p>	<p>Будущие надсистемы: Коллекции</p>
<p>Похожие системы: Списки Стек Очередь Дек Множества Кортеж</p>	<p>Структура массива:</p> 	<p>Создание коллекции:</p> 
<p>Прошлые надсистемы: Указатели</p>	<p>Настоящие подсистемы: Упорядоченная структура данных с прямым доступом к элементам. Размер не меняется, строго типизированный.</p>	<p>Будущие подсистемы: Нестрогая типизированная структура данных, которая может менять свой размер, будет эффективной при добавлении и удалении элементов.</p>

Ход проведения эксперимента представлен на рисунке 1.

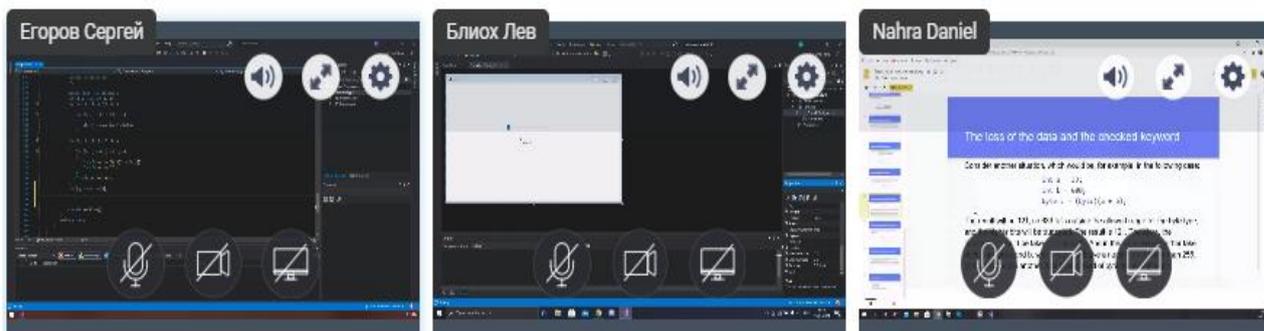


Рисунок 1 – пример экспериментального занятия с учениками []

Для объяснения теории был использован метод «волшебного экрана», во время решения практических заданий и алгоритмических задач наводящими вопросами помогали найти правильный ответ. При этом не ограничивали учащихся в идеях для проявления максимальной креативности и находчивости.

Для объяснения был выбран язык программирования Python, который является достаточно простым для понимания, и имеет интуитивно понятный синтаксис. Темы для объяснения теории были взяты следующие: «списки», «функции» и «объектно-ориентированное программирование». Составление «волшебного экрана» потребовало дополнительных затрат по времени со стороны преподавателя до занятия, а также объяснения структуры и составляющих «экрана» детям на занятии.

В эксперименте изучение первой темы таким способом давалось сложно из-за новизны материала и непривычности подхода. Однако в дальнейшем при использовании «волшебного экрана» для объяснения других тем наблюдалась положительная динамика и сокращение времени на изучение нового материала.

В дополнение к данному методу использовались также цифровые источники информации: анимированные видеоуроки и презентации, составленные методистами-преподавателями школы. Данные материалы построены таким образом, чтобы занимали не более 10 минут урока. Для повышения привлекательности и внимания учащихся в видеоуроках использовались понятные и близкие детям примеры. Например, для объяснения темы «списки» использовалась аналогия списка в магазин, который обычно составляют родители. В презентации добавили gif-изображения, а также картинки по теме.

Принимая во внимание существующие проблемы в преподавании программирования, были изучены основные способы их решения. Так, процесс решения задачи строился на основе диалога, обсуждения возможных подходов, вариантов решения. Ученики старались задать интересующий вопрос и уточнить правильность их метода решения. Тем не менее, некоторые ученики были закрыты к общению и решали задачи самостоятельно. На занятии было хорошо видно, что поиск и изобретение вариантов решения задачи пробуждает в детях азарт, а гордость за самостоятельно найденный ответ служит лучшей мотивацией. Так как нет единственно верного ответа, то это исключает и необходимость в оценке. Дети занимаются с удовольствием, без страха ошибиться или высказать неправильное мнение.

В конце занятия выдавался материал для закрепления темы: теоретическая информация или практические задания. Ученик мог обратиться вне занятия к тренеру для уточнения деталей задания или метода его решения.

В онлайн-школе был также организован и проведен хакатон для тренеров. На данном мероприятии преподаватели объединялись в команды и один день работали над придуманной задачей. На следующий день они презентовали свой продукт. Данный подход позволил поделиться опытом среди преподавателей, проявить креативность в создании программного обеспечения.

На канале YouTube онлайн-школы также периодически проводятся онлайн-трансляции, в которых показано, как тренер создает проект в режиме реального времени. На данный момент это реализовано только с языком программирования Scratch, который является самым простым для понимания. Тем не менее, в будущем планируется в том числе привлечь тренеров к созданию проектов на таких языках, как Python, Java, C# и C++.

На примере проведенного эксперимента было показано, что инструменты ТРИЗ действительно улучшают скорость восприятия информации, дают волю креативности ученикам, поддерживают качество образования и мотивируют на улучшение процесса обучения.

**Список использованных источников:**

1. ТРИЗ как метод развития творческого мышления [Электронный ресурс]. – Режим доступа: <https://netology.ru/blog/06-2020-what-is-triz>. – Дата доступа: 20.03.2021

*57-я научная конференция аспирантов, магистрантов и студентов БГУИР, 2021 г*

2. Возможности использования ТРИЗ в информатике (программирование) [Электронный ресурс]. – Режим доступа: <https://moluch.ru/conf/ped/archive/208/11263/>. – Дата доступа: 20.03.2021.
3. Что такое система ТРИЗ и как она учит детей смекалке и мыслить самостоятельно [Электронный ресурс]. – Режим доступа: <https://mel.fm/pedagogika/9506213-triz>. – Дата доступа: 20.03.2021.
4. Беляцкая, Т. Н. Креативные технологии бизнеса : учеб.- метод. пособие / Т. Н. Беляцкая, В. С. Князькова, С. Л. Фещенко. – Минск : Белорус. гос. ун-т информатики и радиоэлектроники, 2018. – 116 с.
5. Князькова, В. С. Управление креативным потенциалом сотрудников в электронной экономике/ В. С. Князькова, С. Л. Фещенко // Актуальные проблемы менеджмента: менеджмент как важнейший фактор экономического роста и подъема уровня жизни в регионах: материалы междунар. науч.-практ. конф., Санкт-Петербург, 16 нояб., 2018 г./ С-Петерб. гос.ун-т; редкол.: Ю.В.Кузнецов [и др.]. СПб., 2019.
6. Беляцкая, Т. Н. Экосистема электронной экономики: интеллектуальный потенциал / Т. Н. Беляцкая, В. С. Князькова // Вестник ГрГУ – 2018