



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ГУМАНИТАРНО-
ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ЮУрГГПУ»)

ФИЗИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ

КАФЕДРА ИНФОРМАТИКИ, ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
МЕТОДИКИ ОБУЧЕНИЯ ИНФОРМАТИКЕ

Программно-методическая поддержка обучения решению олимпиадных задач на
обработку строк для школьников

Выпускная квалификационная работа
по направлению 44.03.05 Педагогическое образование (с двумя профилями
подготовки)

Направленность программы бакалавриата

«Информатика. Экономика»

Проверка на объем заимствований:

67,57 % авторского текста

Работа рекомендована к защите
рекомендована/не рекомендована

«26» апреля 2018 г.

и.о. зав. кафедрой И, ИТ и МОИ

 Рузаков А.А.

Выполнил:

Студент группы ОФ-513/094-5-1

Щербаков Александр Павлович



Научный руководитель:

к.п.н., доцент кафедры ИИТиМОИ

Паршукова Наталья Борисовна



Челябинск

2018



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное бюджетное образовательное учреждение
высшего образования**

**«ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ГУМАНИТАРНО-
ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ЮУрГГПУ»)**

ФИЗИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ

**КАФЕДРА ИНФОРМАТИКИ, ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
МЕТОДИКИ ОБУЧЕНИЯ ИНФОРМАТИКЕ**

**Программно-методическая поддержка обучения решению олимпиадных задач на
обработку строк для школьников**

**Выпускная квалификационная работа
по направлению 44.03.05 Педагогическое образование (с двумя профилями
подготовки)**

Направленность программы бакалавриата

«Информатика. Экономика»

Проверка на объем заимствований:
_____ % авторского текста

Работа _____ к защите
рекомендована/не рекомендована

« ___ » _____ 20__ г.
и.о. зав. кафедрой И, ИТ и МОИ

_____ Рузаков А.А.

Выполнил:
Студент группы ОФ-513/094-5-1
Щербаков Александр Павлович

Научный руководитель:
к.п.н., доцент кафедры ИИТиМОИ
Паршукова Наталья Борисовна

**Челябинск
2018**

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	3
ГЛАВА 1. ТЕОРЕТИЧЕСКИЙ АНАЛИЗ ПРОБЛЕМЫ ОБУЧЕНИЯ РЕШЕНИЮ ОЛИМПИАДНЫХ ЗАДАЧ НА ОБРАБОТКУ СТРОК.....	6
1.1. Олимпиады по программированию для школьников.....	6
1.2. Автоматические системы проверки задач.....	13
1.3 Типология задач на олимпиадах по программированию.....	20
1.4 Интернет-ресурсы для поддержки олимпиад, которые помогут качественно подготовить учащихся.....	22
1.5 Анализ существующих автоматизирующих систем проверки задач.....	25
Выводы по главе 1.....	29
ГЛАВА 2. ИСПОЛЬЗОВАНИЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ ДЛЯ ОБУЧЕНИЯ РЕШЕНИЮ ОЛИМПИАДНЫХ ЗАДАЧ НА ОБРАБОТКУ СТРОК ДЛЯ ШКОЛЬНИКОВ.....	30
1.1. Методическая разработка курса обучения решению олимпиадных задач по программированию на обработку строк для школьников.....	30
1.2. Программно-методическая поддержка обучения решению олимпиадных задач на обработку строк для школьников.....	37
1.3. Апробация результатов исследования в МБОУ СОШ № 121в рамках учебных занятий.....	45
Выводы по главе 2.....	46
ЗАКЛЮЧЕНИЕ.....	47
СПИСОК ЛИТЕРАТУРЫ.....	49
ПРИЛОЖЕНИЯ.....	51

ВВЕДЕНИЕ

Олимпиады являются важным инструментом системы образования, они выступают в роли эффективного механизма отбора и поддержки талантливой молодежи. Поэтому необходимо, чтобы происходило постоянное совершенствование и улучшение проводимых соревнований, а также совершенствовалась методика подготовки школьников к олимпиадам.

Любому обществу нужны одарённые люди, и задача общества состоит в том, чтобы рассмотреть и развить способности всех его представителей. К большому сожалению, далеко не каждый человек способен развивать свои способности. Очень многое зависит и от семьи, и от школы.

Олимпиада один из самых лучших путей выявления познавательного интереса учеников к программированию. Школьные олимпиады по программированию – это массовый вид соревнований учащихся, цель проведения которых – вовлечение большего числа учеников во внеклассную работу по данному предмету, повышение их интереса к алгоритмическим знаниям, развитие самостоятельности, совершенствование общеучебных и специфических для программирования умений учащихся. Такие цели актуальны всегда.

Основными целями и задачами олимпиады являются создание условий для реализации интеллектуального потенциала обучающихся, пропаганды научных знаний.

Проведение олимпиады служит развитию интереса молодежи к программированию, охватывающей все стороны современного мира. Участие в олимпиаде может стать первым шагом к разумному использованию громадного аппарата вычислительной техники, к участию в развитии информационной среды, а также к возможности стать высококлассным специалистом в области информационных технологий. Прикладной же целью олимпиады по программированию является выявление и привлечение к обучению в вузах области наиболее талантливой и подготовленной в области ИТ молодежи.

Наряду с развитием познавательного интереса олимпиады позволяют более правильно решать вопрос о выборе учениками программирования как дисциплины для более глубокого изучения, что поможет им в дальнейшем определить свою будущую профессию. Следовательно, олимпиады по программированию служат решению задач профориентационной работы с учащимися.

Тематика задач на олимпиадах по программированию довольно большая. Но многие темы даже не рассматриваются в школьном курсе информатики и для лучшего результата на олимпиадах нужно изучать дополнительную информацию. Так тема «Строки», которая хоть и рассматривается в базовом курсе информатики, но в силу ограниченности учебного времени на изложение этой темы в школьном курсе информатики, умений, учащихся будет явно недостаточно для получения высоких результатов. Помимо теории, требуется еще и большой практический опыт, чтобы учащиеся не испытывали дискомфорт или неуверенность при работе с автоматизированной системой проверки. Для помощи учителю в процессе подготовки учащихся к олимпиадам по программированию был создан web-ресурс, позволяющий учителю эффективнее проводить занятия подобного типа.

Объектом исследования является процесс подготовки школьников к олимпиаде по программированию.

Предметом исследования является программно-методическая поддержка для подготовки школьников к решению олимпиадных задач по программированию на обработку строк.

Целью данной квалификационной работы является разработка программно-методической поддержки преподавания темы «Обработка строк» в подготовке школьников к олимпиаде по программированию.

Из всего вышесказанного можно выделить следующие **задачи** данной работы:

1. Изучить теоретические основы решения задач по теме «Обработка строк»

2. Разработать содержание уроков по решению задач по теме «Обработка строк»
3. Разработать информационную и техническую поддержку для обучения учащихся решению задач на обработку строк

Методы исследования: теоретический анализ литературных источников, апробация результатов исследования.

Гипотеза исследования состоит в том, что если в рамках изучения темы «Обработка строк», будет создан курс и специально разработанные задачи, с применением автоматизированной системы проверки задач на уроке, то это позволит улучшить усвоение учениками данной темы, и улучшит их результат на олимпиадах по программированию.

ГЛАВА 1. ТЕОРЕТИЧЕСКИЙ АНАЛИЗ ПРОБЛЕМЫ ОБУЧЕНИЯ РЕШЕНИЮ ОЛИМПИАДНЫХ ЗАДАЧ НА ОБРАБОТКУ СТРОК

1.1. Олимпиады по программированию для школьников

Олимпиады по программированию стали проводиться относительно недавно по сравнению с олимпиадами по математике, физике, химии. Они появились лишь в 80-х годах, когда началось увлечение студентов инженерных специальностей и школьников физико-математических школ алгоритмизацией и программированием.

Олимпиады по программированию, за это время стали широко распространены. В 1989 году появилась первая международная олимпиада школьников по информатике IOI (International Olympiad in Informatics), где участникам предлагается решить и запрограммировать алгоритмические задачи. Олимпиады такого вида позволяют выявлять склонности, к математике и программированию у школьников, а также умение работать в условиях стресса и способности решать сложные задачи в сжатые сроки. Крупные IT компании выступают спонсорами ряда соревнований по программированию. Их интерес понятен: талантливые молодые программисты – основа их бизнеса [4].

Олимпиады по программированию предназначены выявлять более талантливых в области информатики учеников, развивать их способности и повышать интерес к данному предмету. Данные олимпиады дают возможность ученикам получить раннюю профориентацию, что способствует становлению в дальнейшем российских специалистов в области информатики, вычислительной техники и программирования. Но хорошее знание школьного курса информатики не гарантирует успешного выступления на олимпиадах, необходимо заниматься с учащимися и в внеурочное время.

Важной особенностью задач, используемых при проведении школьного и муниципального этапов, является ориентация их на проверку развития у

учащихся теоретического мышления, логики, а также творческих способностей и интуиции.

Во время проведения школьного этапа олимпиады, задачи должны быть подобраны таким образом, чтобы их сложность не отпугивала школьников от участия в олимпиаде по программированию, а наоборот они должны давать возможность показать все свои качества с лучшей стороны.

Часто бывает, что используемые задачи на олимпиадах по программированию, не соответствуют «Стандарту основного и среднего (полного) общего образования по информатике и ИКТ» [14]. Более того, в качестве решения этих задач на олимпиаде требуется предъявить отлаженные программы, написанные на языке программирования высокого уровня, а не описания алгоритмов.

Поэтому оценивать работу конкретного учителя по информатике, по результатам олимпиад по программированию не совсем корректно, так как охват школьной программы по информатике, совершенно не хватает для хорошего выступления школьников на олимпиадах по программированию различного уровня.

Сегодня в России идут изменения в системе образования. Среди основных направлений и первоочередных мер образовательной политики выделены: обновление содержания образования, реализация ФГОС основного и среднего общего образования, предпрофильного и профильного обучения на старшей ступени общеобразовательной школы.

В связи с актуализацией и активизацией олимпиадного движения все острее встает проблема подготовки учащихся к участию в олимпиадах.

Обычно олимпиада проводится в два этапа: заочный (отборочный) и очный (заключительный). Этапы могут состоять из одного или нескольких туров, включающих в себя конкурсные испытания различных видов: как академические соревнования (выполнение профилированных заданий по общеобразовательным предметам или комплексам предметов, соответствующих профилю Олимпиады), так и научно-образовательные, творческие соревнования (защита научно-исследовательских работ,

выполнение проектов, сочинения на темы, связанные с профильной областью предметных знаний, испытание творческих способностей участников Олимпиады и т. д.).

Олимпиада проводится на основе общеобразовательных программ среднего (полного) общего образования. Оргкомитет Олимпиады определяет профили, предметы (комплексы предметов), по которым проводятся интеллектуальные соревнования, утверждает олимпиадные задания, разработанные методической комиссией.

Правильность выполнения заданий всех этапов Олимпиады, оценивание работ осуществляет жюри.

За годы проведения олимпиад школьников по информатике в печатных изданиях и в Интернете было опубликовано много различных материалов, связанных с олимпиадными задачами и методами их решения. Тем не менее, вопрос, как включиться в олимпиадное движение и лучше подготовиться к олимпиадам по информатике, не перестает быть актуальным и сегодня.

В олимпиадной информатике, а именно, во всероссийской олимпиаде школьников, развивается направление, поддерживаемое международным сообществом: алгоритмизация и программирование. Связь этого направления со школьным курсом информатики ограничено лишь несколькими разделами. Для достижения результатов в рамках этого направления знать и уметь требуется много, очень много, и любой школьный учебник не входит в этот необходимый минимум. Но, даже если применить дифференцированный подход к обучению школьников, вряд ли этих часов хватит для подготовки к олимпиадам отдельных школьников с "нуля". Уровень задач олимпиад всех рангов год от года повышается, повышается и уровень подготовки самих участников олимпиад. Поэтому для участников олимпиад необходимы дополнительные знания и умения, которых они не получают в стандартной школьной программе.

Для достижения высоких результатов в них необходимо знать основы программирования, как минимум одного языка, но бывает, что разные задачи лучше решать на разных языках, для достижения лучших результатов. Но

знание только основ программирования, не дает шансов на победу. Для этого нужна целенаправленная подготовка и знание наиболее используемых алгоритмов.

Вот некоторые особенности подготовки школьников к олимпиадному программированию:

1. В школьной программе нет такого предмета «программирование» и даже такого раздела. То есть, обучаемый должен иметь собственную, довольно сильную мотивацию.

2. Действует ограничение, что при решении задач желательно использовать только один из языков программирования (C++ или ПАСКАЛЬ).

3. Постоянные тренировки идут почти на спортивном уровне.

4. Большие затраты времени, длительность олимпиады с разбором часто превышает 6 часов.

5. Алгоритмы и формулы, применяемые при решении большинства задач, изучаются только в ВУЗах.

Традиционно олимпиады школьников являются личными соревнованиями, проводятся по многоуровневой системе, в несколько этапов:

1. Районная олимпиада
2. Городская олимпиада
3. Региональная олимпиада
4. Национальная олимпиада

Победители всероссийской олимпиады получают право участия в международных олимпиадах по информатике. Высокие результаты, показанные на олимпиадах, могут давать некоторые привилегии при поступлении в вуз.

Также проводятся многоуровневые командные олимпиады среди школьников, по правилам аналогичным правилам международных студенческих олимпиад.

Интернет-олимпиада по программированию нужна для того чтобы, активизировать работу по пропаганде профессиональной ориентации и привлечению учащихся к научному творчеству и исследовательской работе во внеурочное время под руководством педагогов в области информатики и программирования.

Сейчас проводится много разных олимпиад по информатике и программированию. В России самую долгую историю имеют олимпиады по правилам ACM ICPC, а также довольно похожие на них олимпиады школьников. В целом они по подготовке очень похожи (хотя немного отличаются тактиками) [4].

Обычно первая задача не требует для своего решения каких-то особенных знаний кроме общепринятых стандартов программирования (простые циклы и/или ветвления). Олимпиадной ее делает какой-нибудь маленький нюанс в условии, который нужно обязательно учесть.

Вторая задача требует хорошего навыка работы с математическими моделями и разнообразием комбинаций циклов и ветвлений, стандартов и специальных знаний начального уровня (например, навык работы со строками и массивами и т.п.).

Третья и четвертая задачи – высокий уровень знаний абстрактных структур данных, теории графов, рекурсий, различных сортировок, умений использовать вспомогательные алгоритмы (функции и процедуры) и т.д.

В современной школе при хорошем раскладе (2 ч информатики в неделю, с 5 класса по 11 класс, например) без дополнительной нагрузки на учителя, возможно, подготовить ребенка для решения первой и, может быть, даже второй задачи на уровне школьного или городского этапа ВОШ. Если же информатика преподается только 1 час в неделю, то при огромных усилиях педагога ребенок сможет решить в лучшем случае первую задачу на школьном (а если пройдет дальше, то небольшой вероятностью на городском этапе).

Рассмотрим учебно-методический комплекс, которыми пользуются большинство учителей информатики сегодня, с точки зрения подготовки по

программированию уже к 9 классу на уроках в средней школе. На сегодняшний день самыми распространенными являются следующие УМК:

- Угринович Н.Д. УМК «Информатика», 7-9 классы
- Семакин М.Г. и др. УМК «Информатика», 7-9 классы
- Босова Л.Л., Босова А.Ю. УМК «Информатика», 5-9 классы.

Итак, рассмотрим эти УМК только по критерию наличия разделов программирования в рекомендованных рабочих программах к началу 9 класса.

Таблица 1.

№	УМК	Класс	Раздел программирования, ч
1	Угринович Н.Д. УМК «Информатика», 7-9 классы	7 класс	-
		8 класс	-
		9 класс	Основы алгоритмизации и объектно-ориентированного программирования, 14 ч
2	Семакин М.Г. и др. УМК «Информатика», 7-9 классы	7 класс	-
		8 класс	-
		9 класс	Управление и алгоритмы, 12 ч Введение в программирование, 15 ч
3	Босова Л.Л., Босова А.Ю. УМК «Информатика», 5-9 классы.	5 класс	-
		6 класс	Алгоритмы, 10 ч
		7 класс	-
		8 класс	Основы алгоритмизации, 10 ч Начала программирования, 10 ч
		9 класс	Алгоритмизация и программирование, 8 ч

При внимательном рассмотрении видим, что к началу 9 класса можно подготовить ребенка к решению хотя бы одной-двух олимпиадных задач только по УМК Босовой Л.Л., Босовой А.Ю., т.к. основы алгоритмического мышления в нем начинают закладываться уже в 6 классе [5]. На этапе изучения тем по алгоритмизации в 6 классе легко можно отобрать детей, которые хорошо усваивают программирование, для дальнейшей

целенаправленной работы с ними. Далее, в 8 классе объемно рассматриваются темы основ алгоритмизации с последующим переключением на применение конкретного языка программирования.

Конечно, для успешного выступления на олимпиадах по программированию этого будет явно недостаточно. Если же учителю поставили именно такую цель, то необходимы внеурочные формы дополнительной подготовки, такие как кружок или факультатив по программированию, где целенаправленно будут изучаться теория, приемы и методы программирования на конкретном языке.

1.2. Автоматические системы проверки задач

Проверка и оценка решений участников является достаточно трудоемким и ответственным процессом при проведении олимпиад по информатике. Ответственность за этот процесс несет жюри олимпиады. От того, как он организован и проведен, зависит правильность и объективность результатов олимпиады.

На первых олимпиадах по информатике процесс проверки решений участников требовал привлечения большого количества специалистов, поскольку каждая задача тестировалась в присутствии участника, и каждый тест запускался вручную членом проверочной комиссии. Все это накладывало определенные ограничения на комплект тестов и систему оценки для каждой задачи, так как сложно было в достаточном объеме протестировать задачу ввиду временных ограничений.

С появлением компьютерных сетей технология проверки работ участников немного улучшилась, поскольку не надо было при тестировании загружать программы участников в компьютер с дискеты. Дискеты использовались в этом случае лишь для резервного хранения на случай, если произойдут какие-либо сбои в работе сервера или компьютерной сети [4].

Стремительное развитие информатики и информационных технологий не только повлияло на содержание олимпиадных задач, но и привнесло много нового в процесс проверки работ участников. Начало автоматизации этого процесса было положено в 1995 г. на Международной олимпиаде по информатике в Нидерландах. После этого и на всероссийских олимпиадах по информатике стали использоваться первые системы компьютерной проверки работ участников.

В последние годы процесс проверки работ участников на заключительных и федеральных окружных этапах всероссийских олимпиад по информатике полностью автоматизирован, и сейчас трудно себе представить, как без используемых там систем можно обойтись на олимпиадах. Важно отметить, что, помимо автоматизации процесса проверки решений, такие системы дали возможность более тщательно проверять

работоспособность программ участников. Если раньше количество тестов для каждой задачи было ограничено количеством проверяющих и существующими технологическими возможностями, то сейчас таких ограничений практически нет, и при тестировании используется то количество тестов, которое жюри и научный комитет считают нужным (но не более 100) [4].

В условиях автоматизированной проверки работ участников проверяемая программа автоматически запускается для каждого участника и по завершении ее работы выходные данные анализируются для оценки полученного решения задачи. При этом анализ полученного решения также осуществляется автоматически как путем простого сравнения ответа с правильным решением, так и с использованием более сложных алгоритмов проверки. С этой целью в состав автоматизированной проверяющей системы входят специальные проверяющие программы (чекеры), которые разрабатываются для каждой задачи в отдельности.

Если сам процесс проверки решений участников может быть полностью автоматизирован, то процесс разработки самих тестов от этого не упростился, а усложнился. Об использовании на олимпиадах по информатике полной системы тестов для каждой задачи говорить не приходится, но можно разработать тесты, которые полностью проверяют логику возможных алгоритмов, используемых при ее решении [4].

Тесты разрабатываются таким образом, чтобы в максимальной степени способствовать выявлению особенностей алгоритма решения задачи и давать возможность жюри дифференцировать полученные участниками решения по степени их корректности и эффективности.

В общем случае тесты делятся на следующие группы:

1. простые тесты;
2. тесты на все частные случаи, позволяющие выявить особенности используемых алгоритмов;
3. общие тесты (достаточно случайные тесты);
4. антиэвристические тесты;

5. тесты максимальной размерности (тесты с использованием максимальных значений входных переменных, позволяющие оценить эффективность предложенных алгоритмов или их работоспособность в экстремальных условиях).

Если есть набор тестов для задачи, то разрабатывается методика для оценивания решения участника. В общем случае она выглядит следующим образом.

1. Определяется максимальное количество баллов, которое участник может получить за правильное решение.

2. Максимальное количество баллов за задачу распределяется между всеми тестами из набора тестовых входных баллов.

3. В случае правильного ответа на данный тест участнику начисляется определенное выше для этого теста количество баллов, в противном случае - ноль баллов, за исключением тех случаев, когда для начисления баллов за тест необходимо прохождение некоторых других тестов.

4. Общая оценка определяется как сумма баллов, набранных за правильно исполненные программы.

При определении максимального количества баллов за задачу можно использовать два подхода. Первый основан на предварительной оценке членами жюри относительной сложности отобранных задач и последующем назначении максимального количества баллов за задачу с учетом этих оценок. Второй подход заключается в том, что каждая задача оценивается одинаково, например, исходя из 100 баллов, независимо от того, какое мнение относительно ее сложности имеют члены жюри [4].

В последнее время на заключительных этапах всероссийских и международных олимпиад по информатике наиболее часто используется второй подход, т. е. каждая задача оценивается исходя из 100 баллов, независимо от ее предполагаемой сложности. Это объясняется следующими фактами.

Сказать перед началом тура, какая задача будет для участников сложной, а какая нет, практически невозможно, за исключением очевидных

случаев. Попытки вводить различные коэффициенты сложности задачи до тура и после тура были на первых всесоюзных и всероссийских олимпиадах по информатике, но потом от этого отказались, так как на результаты участников влияют многие факторы, учесть которые очень сложно.

Например, если расположить задачу первой по порядку в списке задач тура, то, какой бы сложности она ни была, все равно большинство участников начнут сначала решать именно эту задачу и результаты ее решения будут достаточно хорошими. С другой стороны, если в тексте задачи указывать в явном виде уровень сложности задачи (максимальное количество баллов, которое может получить участник), то многие неуверенные в своих силах участники начинают решать задачи, которые оценены меньшим количеством баллов, в то время как сильные участники поступают наоборот. В результате как те, так и другие могут потратить много времени на решение первой, выбранной ими задачи и не дойти до других задач не потому, что они сложные, а потому, что не хватило времени на их решение. К тому же на олимпиадах по информатике разного Уровня не так уж редки случаи, когда сильные участники долго решали самую простую задачу, но решить ее так и не смогли. Это уже проблемы психологической устойчивости участников, которые играют не менее важную роль, нежели уровень подготовленности к соревнованиям [4].

Наличие набора тестов и распределение максимального количества баллов за задачу между всеми тестами является необходимым условием для реализации процесса автоматизированной проверки работ участников. В настоящее время при проведении различных соревнований и турниров по информатике используется достаточное количество проверяющих систем. Рассмотрим, что представляет собой автоматизированная система проверки работ участников, которая используется при проведении федеральных окружных и заключительных этапов Всероссийской олимпиады школьников по информатике.

Система состоит из комплекса программ, причем взаимодействие участника олимпиады и администратора системы осуществляется через Web-

интерфейс. В процессе проведения туров система обеспечивает выполнение таких функций, как:

- отправка решений участников на проверку;
- получение результатов проверки;
- сохранение файлов и их восстановление;
- печать файлов.

После объявления предварительных результатов каждого тура система предоставляет каждому участнику возможность доступа:

- к сданным на проверку его решениям;
- к тестам, используемым при проверке решений участников;
- к результатам проверки его решений.

С учетом необходимости высокой надежности и безопасности работы программного обеспечения, используемого участниками во время туров, в правила соревнований включены очень жесткие требования к программам участников. В частности, в них категорически запрещалось:

- использовать в любой форме доступ к сети;
- использовать системные вызовы;
- читать и создавать файлы, не указанные в тексте задач;
- использовать любые попытки воздействия на систему безопасности сети и систему оценки решений участников;
- запускать на исполнение программы, не указанные в тексте задач;
- изменять настройки файловой системы;
- читать информацию в файловой системе.

При работе с системой основными для пользователя являются две Web-страницы: страница регистрации участника олимпиады и основная страница. Страница регистрации участника олимпиады предназначена для входа в систему. Все участники соревнований вместе с материалами, раздаваемыми в начале тура, получают пароль. Переход к основной странице возможен только после успешной регистрации участника олимпиады на странице регистрации.

С основной страницы можно отправить решение задачи на проверку, просмотреть результаты проверки других отправленных задач или полученные ответы. В начале страницы выводится информация о текущем туре — время начала и конца тура, сколько времени прошло с начала тура и сколько осталось. Страница содержит также ссылки на страницу регистрации участника.

При сдаче решения участник олимпиады должен указать задачу, которую он сдает, язык программирования (возможно, конкретный компилятор, так как для языка Pascal и C++ на олимпиадах разрешается использование нескольких компиляторов), выбрать имя файла, содержащего исходный текст решения. Сразу же после отправки можно посмотреть результат: прошли ли тесты, указанные в условии задачи, и принята ли задача на дальнейшую проверку или нет [6].

Процесс обработки переданного на проверку файла успешно завершается, если выполняются все требования, описанные в условии задачи. Одна часть этих требований является общей для всех задач, другая часть требований описывается в условии каждой конкретной задачи.

Если результатом решения задачи должны быть выходные файлы, то процесс передачи их на проверку несколько отличается от ранее описанной процедуры. В этом случае, перед тем как послать выходной файл, необходимо поместить на основной странице дополнительную информацию. В остальном передача на проверку выходного файла не отличается от передачи на проверку файла с исходным кодом программы.

Участники олимпиады могут использовать систему в режиме тестирования во время тура, если решением задачи является файл с исходным кодом программы. Для этого на основной странице системы есть специальное окно. Полезность использования этого режима заключается в том, что в процессе такого тестирования проверяются все ограничения по памяти и времени исполнения программы.

После окончания тура все принятые системой на проверку решения проходят тестирование в автоматическом режиме [6].

Участниками олимпиад и членами жюри часто вводится неофициальная классификация задач олимпиадного программирования. Это связано с тем, что, несмотря на различие сюжетов, алгоритмы и структуры данных, используемые для решения задач часто похожи друг на друга и их можно разделить на несколько категорий. Категории обычно носят названия классов алгоритмов или соответствующих разделов математики. Необходимо отметить, что всегда возможно придумать задачу, не относящуюся ни к одной из категорий. Для этого вводят категорию «Разное». Кроме того, одна и та же задача может входить сразу в несколько разделов.

1.3 Типология задач на олимпиадах по программированию

Участниками олимпиад и членами жюри часто вводится неофициальная классификация задач олимпиадного программирования. Это связано с тем, что, несмотря на различие сюжетов, алгоритмы и структуры данных, используемые для решения задач часто похожи друг на друга, и их можно разделить на несколько категорий. Категории обычно носят названия классов алгоритмов или соответствующих разделов математики. Необходимо отметить, что всегда возможно придумать задачу, не относящуюся ни к одной из категорий. Для этого вводят категорию «Разное». Кроме того, одна и та же задача может входить сразу в несколько разделов.

- Пробная задача
- Утешительная задача
- Эмуляция
- Теория чисел
- Комбинаторика
- Теория вероятностей
- Теория игр
- Теория графов
- Длинная арифметика
- Динамическое программирование
- Двоичный поиск
- Жадный алгоритм
- Линейное программирование
- Структуры данных
- Работа со строкой
- Последовательности
- Полный перебор
- Оптимизация
- Физика
- Вычислительная геометрия

- Разное

На каждую категорию можно придумать задачи самого разного уровня сложности. Поэтому факт отнесения задачи к одной из категорий не может свидетельствовать о её сложности.

Тематика задач на олимпиадах по программированию довольно большая. Для того что бы выучить все темы может уйти достаточно большое количество времени и сил у участников олимпиады по программированию. Для участия в олимпиадах по программированию нужно знать следующий минимум по теме «Строки» [4]:

Задачи на олимпиадах на строки решаются исходя из простых операций на строках:

- определение количества символов;
- замена символов в строке;
- удаление символа в строке;
- вставка символа в строку;

А так же сложных алгоритмов, таких как:

- анализ символа на принадлежность к группе;
- поиск строки в блоке текста;
- обращение строки;
- алгоритмы приближенного поиска подстрок.

Это лишь часть того, что должен знать участник олимпиады по программированию, решающий задачу на обработку строк. Помимо теории, требуется еще и большой практический опыт, чтобы учащиеся не испытывали дискомфорт или неуверенность при работе с автоматизированной системой проверки.

1.4 Интернет-ресурсы для поддержки олимпиад, которые помогут качественно подготовить учащихся

Список web-ресурсов, которые помогут качественно подготовить учащихся достаточно большой. Существует множество сайтов, в которых представлено большое количество различных алгоритмов по программированию. Так сайт: Дистанционная подготовка по информатике. Данный сайт, поддерживаемый Московским центром непрерывного математического образования, содержит большое количество задач по программированию различного уровня. Идеально подходит для тех, кто делает первые шаги в программировании: во многих разделах есть ссылки на теоретический материал по соответствующей теме, к большинству задач приложен подробный разбор.

Для всех заданий доступна автоматизированная проверка решений. На сайте также размещены авторские курсы, составленные ведущими специалистами в области олимпиадной информатики. Более опытные школьники найдут задачи олимпиад самого высокого уровня, включая всероссийские и международные.

Следующий web-ресурс, на который стоит обратить внимание при подготовке школьников это Codeforces (Codeforces.com) [1]. Портал, объединяющий огромное количество участников соревнований по программированию по всему миру. На сайте регулярно проводятся онлайн-соревнования для школьников самого разного уровня: от начинающих до многократных чемпионов мира. Многие известные компании, в том числе ВКонтакте, Mail.Ru, Тинькофф Банк и AIM Tech проводят на платформе официальные соревнования.

Помимо этого, на портале обсуждается все, что связано с программированием, начиная от только-только опубликованных статей о структурах данных и заканчивая эмоциями о недавно прошедшем соревновании. На сайте также содержится большой архив задач, доступных для автоматизированной проверки.

Вики-конспекты это - энциклопедия по дискретной математике и теории алгоритмов, составленная студентами ИТМО. В ней описано большинство алгоритмов, используемых на олимпиадах по программированию. Многие статьи содержат примеры задач и псевдокоды приведенных алгоритмов. Конспекты написаны очень подробно и качественно. Это один из немногих ресурсов на русском языке по данной теме.

Не менее полезный сайт Maximal (e-maxx-ru.1gb.ru) [2]. Maximal - это мини-энциклопедия, содержащая наиболее популярные алгоритмы в олимпиадном программировании, к большинству, из которых приведены реализации и примеры использования. Сайт отличается чуть более неформальным стилем изложения (что иногда может сказаться на качестве статей или корректности алгоритмов), однако он облегчает восприятие информации. На сайте размещены ссылки на полезные книги для более детального изучения приведенных алгоритмов, а также разобраны некоторые конкретные задачи, представляющие особенный интерес [2].

Сайт Олимпиады по информатике, посвященный олимпиадам школьников по программированию в Санкт-Петербурге, официальный сайт Всероссийской командной олимпиады школьников (ВКОШП), индивидуальной олимпиады школьников по информатике и программированию (ИОИП). Одним из главных достоинств этого сайта является очень богатый архив проводимых в России мероприятий, в том числе Всероссийской олимпиады: сайт содержит презентации с разбором задач и результатами соревнований. Также здесь регулярно проводятся личные и командные соревнования для школьников.

Сайт Olympiads.ru, посвященный олимпиадам школьников по программированию в Москве, официальный сайт Открытой олимпиады школьников по программированию, задачи на которой не уступают по сложности заданиям Всероссийской, а иногда изящнее и интереснее. Помимо этого, олимпиада включает заочный тур, задачи которого часто требуют изучения новых алгоритмов в течение соревнования. На сайте опубликованы

материалы прошедших соревнований, а также ссылки на информацию о предстоящих событиях.

Все эти web-ресурсы просто необходимы для тех, кто обучает школьников программированию, многие ресурсы содержат список задач, которые уже были на олимпиадах различного уровня. А алгоритмы, рассматриваемые в этих ресурсах, помогут участникам в олимпиадах решать больше и более качественно задачи.

1.5 Анализ существующих автоматических систем проверки задач

Важной составляющей при изучении языков программирования является применение полученных знаний на практике путем решения различных задач. Так как задачи по программированию легко формализуются, для их проверки удобно использовать системы, которые позволяют осуществлять этот процесс автоматически. На сегодняшний день существует большое число проектов, где используются подобные системы. Общий подход в таких проектах таков: для проверки задачи пользователю необходимо загрузить исходный код с решением на сайт системы или же просто скопировать его в текстовое поле. Далее решение проверяется системой на некотором заранее подготовленном наборе тестовых данных, после чего выводится результат проверки. Далее приведен обзор некоторых проектов, где используются системы для автоматической проверки задач по программированию.

Проект «Школа Программиста», который находится по адресу <https://acmp.ru/> [9]. Данный проект создан для повышения у школьников Красноярского края уровня программирования и способностей, направленных на решение олимпиадных задач. На сайте данного проекта содержится архив задач по программированию со встроенной проверяющей системой (рис. 1). Для того чтобы начать решать задачи, необходимо зарегистрироваться и перейти в раздел «Архив задач», где на текущий момент (08.05.18) содержится 700 задач различной сложности. Проверка решений осуществляется следующим образом: пользователь, решив задачу на одном из поддерживаемых системой языке программирования, загружает файл с исходным кодом на странице с задачей (или копирует исходный код в текстовое поле после условия задачи), после чего попадает на страницу «Состояние системы», где может видеть результат проверки (рис. 2). Ограничением данной системы является то, что сторонний пользователь не может добавлять свои задачи в уже существующий архив [9].

ЗАДАЧА №13		
Быки и коровы		
<i>(Время: 1 сек. Память: 16 Мб Сложность: 26%)</i>		
<p>Петя и Вася часто играют в различные логические игры. Недавно Петя поведал Васе о новой игре «Быки и коровы» и теперь они играют в эту игру сутками. Суть игры очень проста: Петя загадывает четырехзначное число, состоящее из различных цифр. Вася отгадывает задуманное Петей число, перебирая возможные варианты. Каждый раз Вася предлагает вариант своего числа, а Петя делает Васе подсказку: сообщает количество быков и коров, после чего Вася с учетом подсказки продолжает отгадывание числа до тех пор, пока не отгадает. Быки – это количество цифр в предложенном Васей числе, совпадающих по значению и стоящих в правильной позиции в задуманном Петей числе. Коровы – количество цифр, совпадающих по значению, но находящихся в неверной позиции. Например, если Петя задумал число 5671, а Вася предложил вариант 7251, то число быков равно 1 (только цифра 1 на своем месте), а число коров равно 2 (только цифры 7 и 5 не на своих местах). Петя силен в математике, но даже он может ошибаться. Помогите Пете написать программу, которая бы по заданному Петей и предложенному Васей числам сообщала количество быков и коров.</p>		
Входные данные		
<p>В единственной строке входного файла INPUT.TXT записано два четырехзначных натуральных числа A и B через пробел, где A – заданное Петей число, а B – предложенный Васей вариант.</p>		
Выходные данные		
<p>В выходной файл OUTPUT.TXT нужно вывести два целых числа через пробел — количество быков и коров.</p>		
Примеры		
№	INPUT.TXT	OUTPUT.TXT
1	5671 7251	1 2
2	1234 1234	4 0
3	2034 6234	2 1

Рис. 1. Пример задачи и решения для проекта «Школа программиста»

№	Ф.И.О.	Город	Задачи	Рейтинг
1	Зубков Олег Владимирович	Иркутск	700	28720
2	Меньшиков Фёдор Владимирович	Вологда	700	28720
3	Шумейко Максим Владимирович	Старокопстантинов	698	28641
4	Владимир Игоревич Лукьянчиков	Самара	699	28625
5	Фолунин Владимир Александрович	Ульяновск	699	28625
6	Демиденко Виталий	Красноярск	698	28610
7	Андрей и Костя	Майкоп	698	28609
8	Пашментов Никита Владимирович	Вологда	698	28543
9	Чернышов Андрей Максимович	Миасс	697	28488
10	Кохась Константин Петрович	Санкт-Петербург	695	28414

Рис. 2. Пример результата проверки для проекта «Школа программиста»

Проект «Timus Online Judge» acm.timus.ru. [3] Данный проект, аналогично предыдущему, содержит архив задач по программированию с автоматической проверяющей системой (рис. 3). Для того, чтобы решать задачи и принимать участие в олимпиадах, необходима регистрация. Проверка решений осуществляется так же, как и в проекте «Школа программиста»: пользователь загружает файл с исходным кодом решения или копирует код в текстовое поле (рис. 4), после чего попадает на страницу «Состояние системы», где может видеть результат проверки (рис. 5). В данной системе имеется возможность предложить свою собственную задачу, которая может быть добавлена в уже существующий архив [3].

1723. Книга Сандро

Ограничение времени: 0.5 секунды

Ограничение памяти: 64 МБ

Прошло уже немало лет с тех пор, как Лич Сандро ушёл на заслуженный отдых. Иногда по вечерам, когда ему становится совсем тоскливо, он берёт в руки книгу, которую ему подарили воспитанники-маги по случаю выхода на пенсию.

Вот и сейчас великий маг взял с полки книгу и углубился в чтение. В одной из глав рассказывалось про знаменитое открытие Сандро — много лет назад он придумал универсальное заклинание. Оказалось, что любая его подстрока (последовательность подряд идущих букв) тоже является заклинанием, а сила любого заклинания равна количеству раз, которое это заклинание встречается в универсальном (например, строка «ие» встречается в строке «qieie» дважды, а строка «aba» в строке «abababa» — трижды).

Сейчас у Сандро много свободного времени, и он решил найти самое сильное заклинание. Помогите ему в этом.

Исходные данные

Единственная строка содержит универсальное заклинание, которое открыл Сандро. Заклинание — непустая строка из строчных латинских букв длиной не более 50.

Результат

Выведите любое из заклинаний, обладающих, по мнению Сандро, наибольшей силой.

Пример

исходные данные	результат
tebidohtebidoh	tebidoh

Автор задачи: Иван Бурмистров (подготовка — Ольга Соболева)

Источник задачи: Уральская региональная командная олимпиада по программированию 2009

Метки: [строковые алгоритмы](#) (скрыть метки для нерешенных задач)

Сложность: 66 [Версия для печати](#) [Отправить на проверку](#) [Обсуждение на форуме \(38\)](#)

[Все попытки \(12122\)](#) [Все успешные попытки \(5516\)](#) [Рейтинг решений \(4396\)](#)

Рис. 3. пример задачи проекта «Timus Online Judge»

Отправка решения на проверку

Если у вас нет JUDGE_ID, пожалуйста, [зарегистрируйтесь](#). Перед тем как отправлять на проверку свою первую программу, ознакомьтесь с [руководством](#).

JUDGE_ID: Язык: FreePascal 2.6 ▾ Задача:

Исходный код решения (не более 64 КБ):

```
var a:integer;
begin
  read(a);
  write(a);
end.
```

Или файл с исходным кодом решения (не более 64 КБ):
 Файл не выбран

Рис. 4. пример формы для отправки решения на проверку для проекта «Timus Online Judge»

Результаты проверки решений

ID	Дата	Автор	Задача	Язык	Результат проверки	№ теста	Время работы	Выделено памяти
7875773	10:57:03 8 май 2018	MyCatWantsToKill	1086. Криптография	Java 1.8	Accepted		0.234	7 072 КБ
7875772	10:48:28 8 май 2018	vup1	1409. Два бандита	FreePascal 2.6	Accepted		0.001	220 КБ
7875771	10:47:38 8 май 2018	vup4	1409. Два бандита	FreePascal 2.6	Accepted		0.001	220 КБ
7875770	10:43:47 8 май 2018	vup1	1409. Два бандита	FreePascal 2.6	Wrong answer	2	0.015	220 КБ
7875769	10:42:42 8 май 2018	vup1	1409. Два бандита	FreePascal 2.6	Compilation error			
7875768	10:41:48 8 май 2018	vup1	1409. Два бандита	FreePascal 2.6	Wrong answer	2	0.001	220 КБ
7875767	10:41:04 8 май 2018	Ahmadjon(TUJ)	1075. Нитка в пространстве	Java 1.8	Wrong answer	4	0.093	2 232 КБ

Рис. 5. пример результата проверки для проекта «Timus Online Judge»

Несмотря на то, что каждый из проектов содержит большое количество разнообразных задач, ни один из них не может быть успешно использован при изучении темы «Обработка строк» на школе, так как процесс изучения этой дисциплины имеет некоторые особенности, которые и не позволяют использовать уже готовые (в том числе и рассмотренные) системы. К таким особенностям относится, например, организация учебного процесса, а именно: в нашем случае необходимо, чтобы учитель имел возможность создать свой класс и добавить в него свои задачи, после чего школьнику по каждому из уроков выдались бы задачи, обязательные для решения. При этом у школьника должна иметься возможность просматривать результаты других участников своего класса. Также необходимый нам набор задач требует отдельной реализации, так как добавление этих задач в стороннюю систему может оказаться трудоемким процессом. Ещё одна важна особенность – преподаватель должен иметь дополнительные административные возможности в системе, а именно: просмотр решений всех участников, возможность редактировать профили участников, возможность редактировать текущие задачи и группы. Ни одна из существующих на данный момент систем не подходит под эти требования, именно поэтому и требуется создать свою собственную систему.

Выводы по главе 1

В данной главе были рассмотрены основные аспекты, касающиеся олимпиад по программированию, была рассмотрена основная типология задач, которые используют на олимпиадах по программированию различного уровня. Особое внимание было уделено задачам на обработку строк. Также были проанализированы основные курсы изучения данной темы в современных учебниках информатики, которые используются в нынешних учебных программах. Были представлены основные web-ресурсы, которые будут полезны при подготовке учеников к олимпиаде по программированию.

Анализ учебных комплексов по Информатики и ИКТ для 7-11 классов таких авторов как: Угринович Н.Д, Семакин М.Г., Босова Л.Л. и др., на предмет наличия там конкретных знаний об алгоритмах на обработку строк, выявил недостаточное содержание актуальной информации на данную тему, которая необходима для успешного участия на олимпиадах по программированию любого уровня [13] [12] [5].

Исходя из всего выше сказанного, можно сделать вывод о необходимости разработки методики решения олимпиадных задач на обработку строк для школьников. А так же разработка автоматизированной системы проверки задач, которая будет не только помогать учителю гораздо быстрее и удобней готовить учеников, но и улучшит качество подготовки школьников к участию на олимпиадах по программированию.

ГЛАВА 2. ИСПОЛЬЗОВАНИЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ ДЛЯ ОБУЧЕНИЯ РЕШЕНИЮ ОЛИМПИАДНЫХ ЗАДАЧ НА ОБРАБОТКУ СТРОК ДЛЯ ШКОЛЬНИКОВ

1.1. Методическая разработка курса обучения решению олимпиадных задач по программированию на обработку строк для школьников

Важной задачей образования является работа с одаренными учащимися, их подготовка к предметным олимпиадам. Олимпиада по программированию занимает одно из ведущих мест, в связи с интенсивным развитием информационных технологий, как в нашей стране, так и за рубежом.

Участие в олимпиадах позволяет развивать творческие способности школьников и обеспечивает высокую мотивацию к образовательной деятельности.

Классическая олимпиада по программированию – это олимпиада, которая предполагает наличие обширных познаний в математике и языках программирования.

Решение олимпиадных задач позволяет раскрыть творческий потенциал школьника во время подготовки к олимпиаде, учитывая возрастные особенности ребенка и перспективу его развития. Использование многоуровневых олимпиадных задач, позволяет школьникам применить свой творческий потенциал, независимо от уровня подготовки.

Курс занятий по олимпиадному программированию (решение олимпиадных задач на строки) ориентирован на учащихся 8-10х классов, обладающих повышенной мотивацией к изучению информатики и имеющих начальные знания в области алгоритмизации на уровне понимания простейших алгоритмов.

Данный элективный курс позволяет провести подготовку к олимпиадам по программированию учеников с 8-го класса, используя методическую коллекцию олимпиадных задач.

Основная цель курса: подготовить учащихся к практическому использованию полученных знаний при решении учебных задач, а затем профессиональной деятельности, вовлечение учащихся в пробное участие в олимпиадах по программированию разного уровня.

Данная программа представляет большую практическую значимость с точки зрения совершенствования непрерывной работы с одаренными школьниками в рамках олимпиадного движения по информатике и школьного образования.

Пояснительная записка.

Программа данного курса рассчитана на учащихся 8-10 классов, имеющих навыки работы с каким-либо языком программирования и желающих расширить свои знания в области решения задач по программированию. Для успешного изучения курса необходимо, чтобы учащийся имел представление об алгоритмизации, процедурном программировании, знал основные операторы и управляющие конструкции языков программирования.

Предпочтительным языком программирования является Паскаль в силу легкости освоения и наглядности. Данный язык поддерживаются разработанной автоматизированной системой тестовой проверки решений задач по программированию olympcode.h1n.ru, использование которой является неотъемлемой частью данного курса.

Основные задачи курса:

- развитие логического мышления учащихся;
- сформировать первоначальные знания о разделе программирования «Строки»;
- развитие навыков программирования алгоритмических структур на обработку строк;

- научить учеников решать задачи на обработку строк;
- получить практические навыки программирования строк;
- развитие интеллекта учащихся;
- способствовать развитию познавательного интереса к программированию.

По результату освоения дисциплины учащийся должен:

Знать:

- основные действия над строками;
- основные алгоритмы обработки строк;
- алгоритмы поиска в строке;
- палиндромы.

Уметь:

- решать олимпиадные задачи по программированию на обработку строк.

Владеть:

- приемами решения олимпиадных задач по программированию на обработку строк.

Факультативный курс заканчивается пробный участием в олимпиаде по программированию. Для контроля знаний используется рейтинговая система.

Итоговой работой будет являться участие в пробном этапе по программированию, где будут подобраны задачи на обработку строк.

В таблице 2 представлен тематический план факультатива «Обработка строк».

Основное содержание курса направлено на изучение основ алгоритмов на обработку строк, а также на особенности применения данных алгоритмов для решения различных олимпиадных задач по программированию. В предложенном курсе, в том числе рассматриваются вопросы, связанные с проведением современных олимпиад по программированию.

Тематический план

Номер	Тема урока	Количество часов		
		Всего	Теория	Практика
1	Введение в тему: обработки строк	1	1	0
2	Основные действия над строками	1	0,3	0,7
3	Основные алгоритмы обработки строк	4	1	3
4	Алгоритмы поиска в строке	2	1	1
5	Палиндромы.	2	1	1
6	Решение легких олимпиадных задач	2	0	2
7	Решение сложных олимпиадных задач	2	0	2
8	Пробное участие в олимпиаде	2	0	2
Итого:		16	5	11

Содержание программы

Урок №1.

Тема урока: Введение в тему: обработка строк.

Класс: 8-10.

Цели урока: Знакомство с основными понятиями обработки строк, виды задач, решаемых методом обработки строк.

Тип урока: урок усвоения новых знаний и умений.

Перечень средств ИКТ, используемых на уроке: интерактивная доска, видеопроектор, презентация.

Основные понятия: символьный тип, символ, алфавит, строка, длина цепочки, пустая строка, подстрока.

ЦОРы для урока: презентация «Введение в тему: обработка строк» , сайт автоматизированной проверки задач по программированию.

Методические рекомендации к уроку для учителя: Так как этот урок является основополагающим при изучении темы «Обработка строк», необходимо следить за тем, насколько успешно ученики усваивают новый

материал. Дальнейшее качество изучения данного курса будет зависеть от того как ученики поняли данную тему, т.к. в ней заложены основные понятия обработки строк. Дальнейшее изучение данного курса без знаний основ будет затруднительно, так как дети не будут понимать учителя. Поэтому учителю необходимо следить, что бы все ученики поняли эту тему хорошо.

Урок №2.

Тема урока: Основные операции над строками

Класс: 8-10.

Цели урока: Знакомство с основными действиями со строками в Pascal. Закрепление полученных знаний на практике.

Тип урока: урок усвоения новых знаний и умений и первичное закрепление знаний на практике.

Перечень средств ИКТ, используемых на уроке: интерактивная доска, видеопроектор, презентация.

Основные понятия: конкатенация (соединение) строк,

ЦОРы для урока: презентация «Основные операции над строками» , сайт автоматизированной проверки задач по программированию.

Методические рекомендации к уроку: Данная тема нужна для дальнейшего обучения курса. Тема будет достаточно сложная для понимания, поэтом у требуется следить за этим, а так же следует уделять особое внимание на сложные операции, коорые ученики не понимают.

Урок №3.

Тема урока: Основные алгоритмы обработки строк.

Класс: 8-10.

Цели урока: Знакомство с основными алгоритмами со строками в Pascal. Закрепление полученных знаний на практике.

Тип урока: урок усвоения новых знаний и умений и первичное закрепление знаний на практике.

Перечень средств ИКТ, используемых на уроке: интерактивная доска, видеопроектор, презентация.

Основные понятия: алгоритм.

ЦОРы для урока: презентация «Основные алгоритмы обработки строк» , сайт автоматизированной проверки задач по программированию.

Методические рекомендации к уроку:

Урок №4

Тема урока: Алгоритмы поиска в строке

Класс: 8-10.

Цели урока: Знакомство с алгоритмами поиска в строке в Pascal. Закрепление полученных знаний на практике.

Тип урока: урок усвоения новых знаний и умений и первичное закрепление знаний на практике.

Перечень средств ИКТ, используемых на уроке: интерактивная доска, видеопроектор, презентация.

ЦОРы для урока: презентация «Алгоритмы поиска в строке» , сайт автоматизированной проверки задач по программированию.

Урок №5

Тема урока: Палиндромы

Класс: 8-10.

Цели урока: Знакомство с полиндромами в Pascal. Закрепление полученных знаний на практике.

Тип урока: урок усвоения новых знаний и умений и первичное закрепление знаний на практике.

Перечень средств ИКТ, используемых на уроке: интерактивная доска, видеопроектор, презентация.

Основные понятия: Палиндром.

ЦОРы для урока: презентация «Палиндромы», сайт автоматизированной проверки задач по программированию.

Урок №6.

Тема урока: Решение легких олимпиадных задач.

Класс: 8-10.

Цели урока: Проверка знаний учащихся на легких задачах, а также научить учеников использовать полученных знаний на практике.

Тип урока: Практическая работа.

Перечень средств ИКТ, используемых на уроке: интерактивная доска, видеопроектор, презентация.

ЦОРы для урока: презентация «Решение легких олимпиадных задач», сайт автоматизированной проверки задач по программированию.

Урок №7.

Тема урока: Решение сложных олимпиадных задач.

Класс: 8-10.

Цели урока: Проверка знаний учащихся на сложных задачах, а также научить учеников использовать полученных знаний на практике.

Тип урока: Практическая работа.

ЦОРы для урока: презентация «Решение сложных олимпиадных задач», сайт автоматизированной проверки задач по программированию.

Перечень средств ИКТ, используемых на уроке: интерактивная доска, видеопроектор, презентация.

Урок №8.

Тема урока: Пробное участие в олимпиаде

Класс: 8-10.

Цели урока: **Закрепить и проверить знания учеников, с использованием автоматической проверяющей системы;**

Тип урока: самостоятельная работа учащихся;

Перечень средств ИКТ, используемых на уроке: интерактивная доска, видеопроектор, презентация.

ЦОРы для урока: презентация «Пробное участие в олимпиаде», сайт автоматизированной проверки задач по программированию.

1.2. Программно-методическая поддержка обучения решению олимпиадных задач на обработку строк для школьников

В качестве программно-методической поддержки методической разработки курса по теме «Программно-методическая поддержка обучения решению олимпиадных задач на обработку строк для школьников» был создан web-ресурс. Его может использовать как школьник, который хочет повысить свой уровень мастерства в олимпиадном программировании, так и для помощи учителю в процессе подготовки учащихся к олимпиадам по программированию. С помощью данного сервиса, учитель сможет эффективнее проводить занятия подобного типа. Ресурс имеет следующий ряд возможностей, которые являются типичными для систем подобного рода [4]:

- архив задач по олимпиадному программированию;
- система автоматической проверки решений;
- получение результатов проверки;
- раздел для проведения самостоятельных работ;
- дистанционный курс олимпиадного программирования;
- рейтинговая система оценки работы участников;
- система общения участников с администратором и между собой;
- описание решений задач;
- и др.

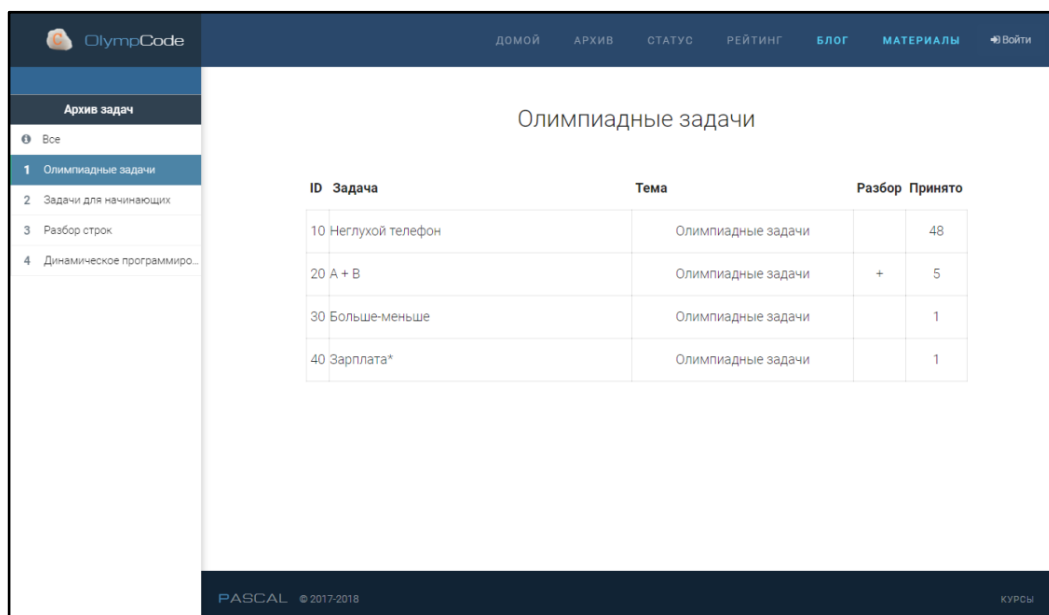


Рис. 6. Архив задач

На странице «Архив задач» (рис. 6) представлен список всех задач, разделенных по темам. Фильтровать задачи можно с помощью пунктов меню, которое расположено слева. В нем содержатся все те темы, в которых есть хотя бы одна задача. В правой части находится непосредственно сам список задач, уже отсортированный по темам. Если у задачи приведено полное объяснение с решением, то в той же строке в столбике разбор будет стоять знак плюс. В этот список, каждый учитель может добавить по необходимости свой набор задач, которые будут видеть только его ученики. Если пользователь авторизованный, то все его решенные задачи выделены зеленым цветом. В последнем столбце Принято указывается количество пользователей, которые полностью решили данную задачу т.е. прошли все приведенные тесты.

Для того чтобы перейти для непосредственного решения задачи, необходимо выбрать задачу из списка и нажать на гиперссылку в названии. После этого ученик попадает на страницу с ввода кода и непосредственного решения выбранной задачи.

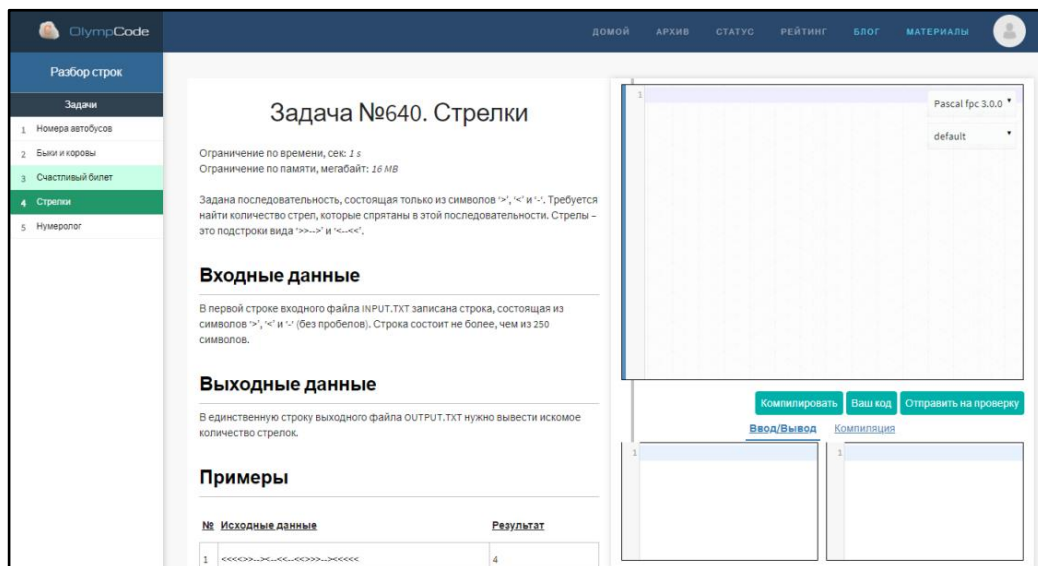


Рис. 7. Страница проверки задачи

На данной странице «проверка задачи», (рис. 7) в правой части страницы в меню находятся все задачи, которые входят в ту же тему, в которую входит выбранная пользователем задача. Так если выбранная задача входит в группу задач с темой «Строки», то все задачи из этой темы будут выведены в меню. Сам текст выбранной задачи расположен по центру страницы, задачи, как на олимпиадах по программированию, могут содержать все основные пункты: название, ограничения по времени и памяти, текст, входные и выходные данные, а также примеры этих входных и выходных данных, если нажать на строчку примера входных данных, то они на автомате проверятся на задаче.

Решать задачу школьники могут как непосредственно прямо на сайте, используя встроенный компилятор, для этого после ввода кода в соответствующее окно нажимается кнопка «Ваш код». А также можно, чтобы школьники решали задачу на компиляторе у себя на компьютере, а после решения вставить код и проверить его на сайте. Введя в поля Входных данных, входные данные для задачи и запустив задачу, ресурс вернет значение, которое задача вывела, в поле выходных данных.

Для проверки задачи на полное решение нажимается кнопка «Отправить на проверку». Каждое отправленное решение проходит на сервере проверку, как правило, не менее чем на 10 тестах, специально

составленных для анализа решений учеников. Задача считается решенной только в случае прохождения всех тестов без ошибок. В случае неверного решения процесс тестирования прерывается на тесте, определившем ошибку. Результаты тестирования задач можно видеть в разделе «Статус», там же отображаются результаты тестирования других пользователей. Итогом проверки является сообщение системы и номер теста, вызвавшего ошибку (если таковая имела место). При переходе на номер сданной задачи в разделе «Статус» пользователи могут просматривать тексты своих отправленных программ, а так же видеть там ошибки компиляции в случае статуса «Ошибка».

Компиляция проходит на специальных сайтах, которые позволяют подключаться к ним, и, отправляя через API запрос, возвращать результат задачи. В данном случае, используется сайт rextester.com, у данного ресурса достаточно большой выбор языков программирования, т.е. все необходимые для проведения олимпиад для школьников: Pascal, c++, Python. Результатом работы сайта является массив JSON, в который входят, следующие значения: результат выполнения программы, статус задачи решена она или нет, если есть ошибки, то выводится её описание.

После решения задачи школьники могут переходить к решению следующих задач из списка.

На странице «Рейтинг», выводится общий рейтинг всех пользователей на данном web-ресурсе. Каждый пользователь может увидеть свой уровень готовности по сравнению с другими пользователями. Что подталкивает их к дальнейшему обучению.

Следующий раздел сайта «Блог», (рис. 8) он предназначен для самостоятельного обучения школьников. Там подобрана информация, которая понадобится им при подготовке к олимпиадам по программированию. Она структурирована по темам, выбор которых происходит в меню с левой стороны.

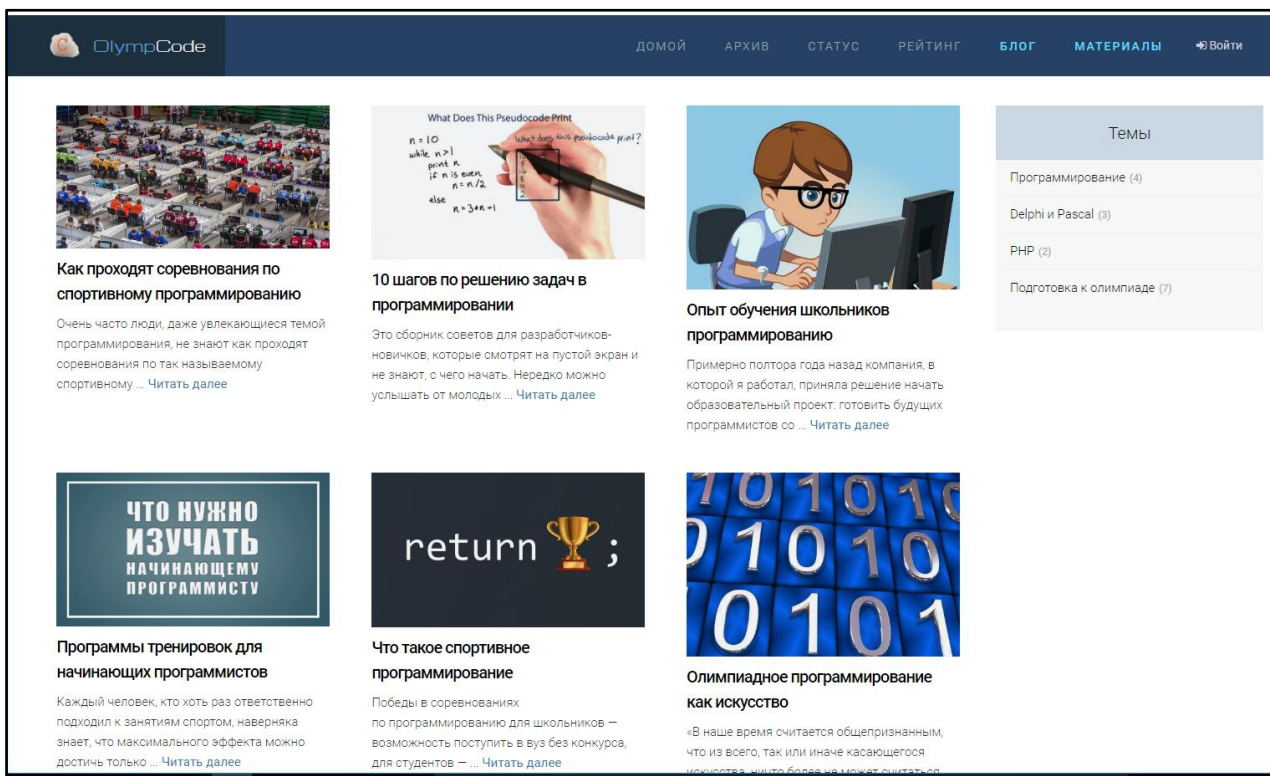


Рис. 8. Страница «Блог»

В разделе «Материалы», представлены курсы которые ученики смогут проходить либо сами, либо вместе с учителем.

Раздел «Материалы» (рис. 9) представляет собой систему дистанционной подготовки школьников к олимпиадам по программированию, которые ученики смогут проходить либо сами, либо вместе с учителем. Представленные здесь курсы структурированы по разделам и темам. Каждая тема содержит ряд задач для закрепления учебного материала. Проверка решений задач происходит в автоматическом режиме.

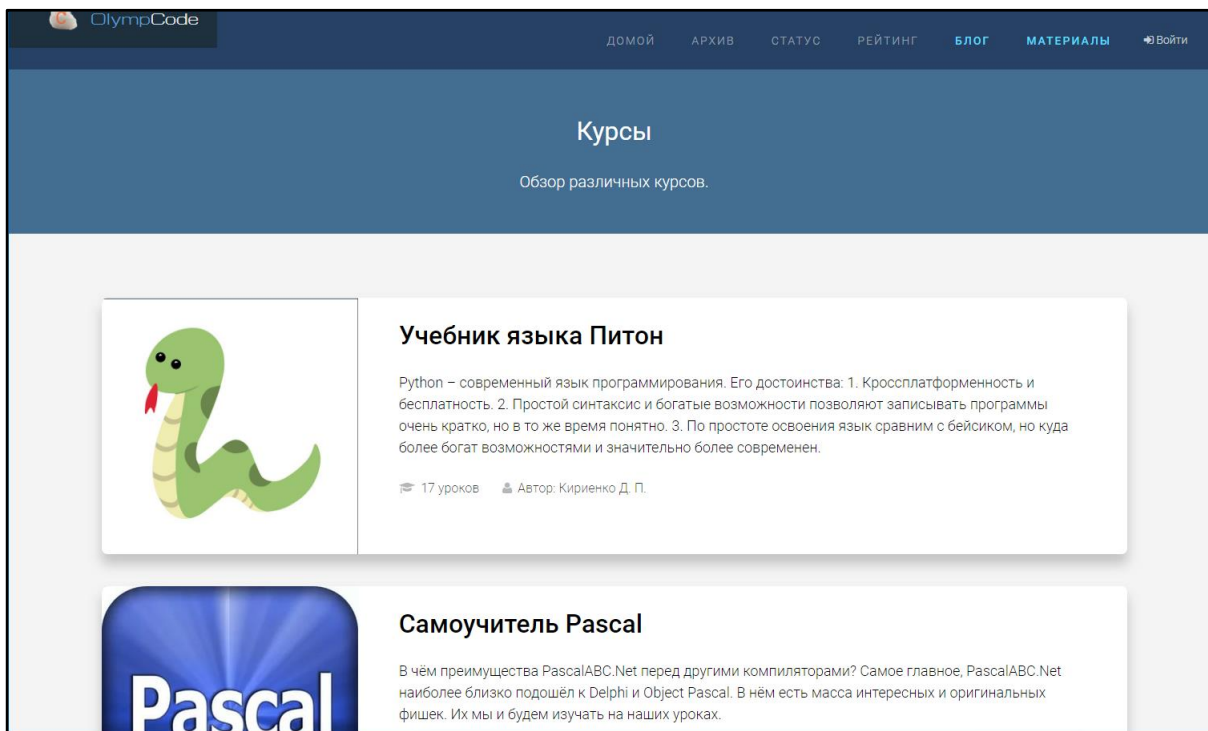


Рис. 9. Страница выбора курса

Каждый учитель, зарегистрированный на данном сайте, сможет добавить свой курс, который будет виден или всем пользователям, или определенному классу.

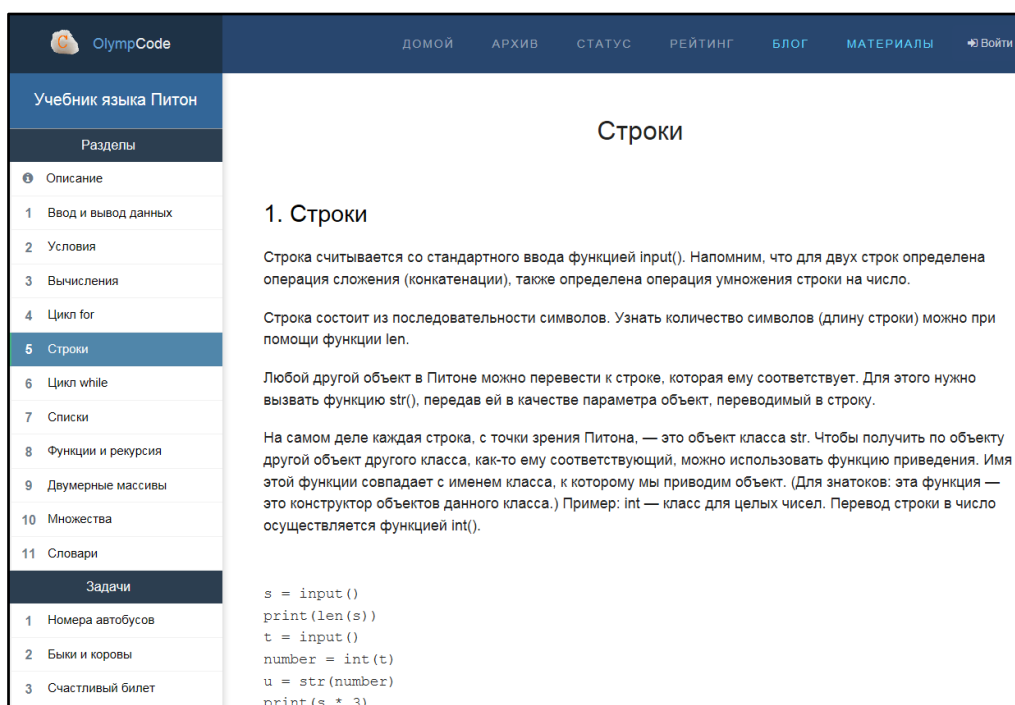


Рис. 10. Страница выбора курса

В данном разделе можно помещать уроки (рис. 10) и соответствующие задачи к ним, разделенные на группы. Учитель видит, на сколько, ученик прошел данный курс.

В разделе для учителя, находятся элементы доступные для редактирования данного сайта.

Разработанная система состоит из набора гипертекстовых документов. Для формирования системы применялись языки HTML, PHP, MySQL, JavaScript и таблицы стилей CSS, система была разработана с нуля. Важной частью всего ресурса является база данных, в которой хранятся значения всех созданных сущностей. В качестве реляционной СУБД была выбрана MySQL. MySQL –свободная система управления базами данных. Распространяется под GNU (General Public License) и под собственной коммерческой лицензией, на выбор. В качестве хостинга, который позволяет выкладывать сайт в интернет, был выбран <https://hostiman.ru>, для данных целей он полностью соответствует требованиям. Сам web-ресурс находится по адресу: <http://olympcode.h1n.ru>.

Данный ресурс можно применять и на обычных уроках информатики, по темам связанные с программированием. Она позволит существенно снизить нагрузку на учителя по просмотру исходных текстов решений задач. А сама система подготовки учащихся к решению олимпиадных задач будет способствовать интересу к информатике и программированию, тщательному отбору участников соревнований городского и регионального уровней, снижением психологического барьера перед боязнью решения сложных, но интересных задач.

Данная система предназначена как для учителей, которые ведут уроки по программированию, так и для учеников, которые сами хотят повысить свой уровень мастерства в олимпиадах по программированию. На данный момент в школах такая система мало используется, и проверка задач по программированию осуществляется вручную. Такой подход к проверке имеет недостатки: во-первых, это затратно по времени, так как одновременно таким образом можно проверять решение только у одного ученика в классе, при этом в целом время проверки ограничено временем занятия на уроке. Во-вторых, при таком подходе трудно гарантировать правильность решения, так

как некоторые ошибки в программе при ручной проверке трудно обнаружить [15].

Разрабатываемая система позволит устранить вышеперечисленные недостатки, а именно: значительно сократится время на проверку, так как система будет реализована в виде web-приложения, поэтому для проверки решения задачи достаточно будет зайти на сайт проверяющей системы, загрузить исходный код решения и получить результат. Также повысится качество проверки за счет возможности заранее создать набор тестовых данных для каждой из задач.

1.3. Апробация результатов исследования в МБОУ СОШ

№121 в рамках учебных занятий

Педагогическая апробация методической разработки интегрированных уроков по теме «обработка строк» проводилась во время педагогической практики в МБОУ СОШ № 121 г. Челябинска. Курс изучался как в общеобразовательном 9 классе, так и в профильных 9 классах, программа которого сформирована в соответствии с требованиями ФГОС. В течении одного занятия была рассмотрена следующая тема:

1. Урок № 1 «Решение олимпиадных задач» - 1 час.

Апробация методической разработки темы к разделу прошла успешно.

Способствовали этому, как и интерес учеников к данной теме, правильная мотивация, цели и задачи изучения темы. Задачи по олимпиадному программированию, рассмотренные на данном уроке, оказались для учащихся новыми, нестандартными, заставили по-новому взглянуть на программирование, но при этом не вызвали трудностей в освоении, так как были подобраны такой сложности, чтобы были и не сильно сложные, но и заставляли задуматься.

Также, на основе разработанной программной поддержки обучения решению олимпиадных задач на обработку строк для школьников, была подготовлена статья для научной конференции [15].

Выводы по главе 2

На основе теоретических положений, изложенных в первой главе, во второй главе представлено описание темы «Обработка строк», выполнено в виде факультативного курса и web-ресурса в виде автоматизированной системы для проверки задач.

Апробация курса проводилась в рамках педагогической практики в МБОУ СОШ № 121. На занятиях ученики 9 классов с удовольствием выполняли задания.

Таким образом, во второй главе исследования был разработан и апробирован модуль к разделу «Обработка строк» и программно-методическая поддержка к нему.

ЗАКЛЮЧЕНИЕ

В данной дипломной работе представлена программно-методическая поддержка обучения решению олимпиадных задач на обработку строк для школьников.

В ходе выполнения дипломной работы был разработан курс для школьников 8-10 классов, подобран список задач для данного курса, а так же разработана система автоматизированной проверки задач по программированию. Данная система позволяет ознакомиться с теоретической информацией, по которой планируется решение задач. Также осуществлены основные функции - возможность решения задач с разрешения преподавателя, получение отчёта о результатах работы школьников.

Подводя итог квалификационной работы, можно сказать, что цель исследования была достигнута в полной мере. Был разработан факультативный курс по изучению темы «Обработка строк» во внеурочной деятельности в школе. Так же была разработана программно-методическая поддержка к нему, в виде автоматизированной системы проверки задач.

Разработанный факультативный курс, рассчитанный на проведение занятий во внеурочное время в 8-10 классах, позволяет улучшить усвоение учениками данной темы, и улучшит их результат на олимпиадах по программированию

В ходе выполнения выпускной квалификационной работы были решены все поставленные задачи и, таким образом, достигнута цель и получены следующие результаты.

1. Изучены теоретические положения по проблеме исследования, в школьном курсе тема «строки» рассматривается не достаточно углубленно.

2. Разработан 16 - часовой школьный факультативный курс по изучению темы «Обработка строк» в школе для 8-10 классов.

3. Разработана программно-методическая поддержка факультативного курса в виде автоматизированной системы проверки задач по программированию.

Разработанный факультативный курс и автоматизированная система решения задач, соответствует всем целям и задачам, облегчая их достижение, позволяя улучшить усвоение учениками данной темы, и улучшит их результат на олимпиадах по программированию на разных этапах.

СПИСОК ЛИТЕРАТУРЫ

1. Codeforces.com [Электронный ресурс]. – <https://codeforces.com>, 2018.
2. e-maxx-ru.1gb.ru [Электронный ресурс]. – <https://e-maxx-ru.1gb.ru>, 2018.
3. Timus Online Judge [Электронный ресурс]. – [https:// acm.timus.ru](https://acm.timus.ru), 2018.
4. Беляев, С.Н. Школа программиста (<http://acmp.ru>) образовательный web-ресурс олимпиадного программирования для школьников [Электронный ресурс]. – <https://cyberleninka.ru/article/v/shkola-programmista-http-acmp-ru-obrazovatelnyy-internet-resurs-olimpiadnogo-programmirovaniya-dlya-shkolnikov>, 2018.
5. Босова, Л.Л. Информатика. Учебник для 8 класса [Текст] : учебник / Л.Л.Босова, А.Ю. Босова, – М. : БИНОМ. Лаборатория знаний, 2013. – 224с.
6. Кирюхин, В.М. Информатика: всероссийские олимпиады. Вып. 1 [Текст] : учеб. пособие / В.М. Кирюхин – М. : Просвещение, 2013. – 220 с.
7. Окулов, С.М. Алгоритмы обработки строк [Текст] : учеб. пособие / С.М. Окулов – 3-е изд. – М. : БИНОМ. Лаборатория знаний, 2015. – 258 с.
8. Примерная основная образовательная программа основного общего образования [Электронный ресурс]. – <http://fgosreestr.ru/wp-content/uploads/2017/03/primernaja-osnovnaja-obrazovatel'naja-programma-osnovogo-obshchego-obrazovaniija.pdf>, 2015.
9. Проект «Школа Программиста» [Электронный ресурс]. – <https://acmp.ru>, 2018.
10. Проект, направленный на повышение уровня программирования у школьников Красноярского края [Электронный ресурс] : Школа программиста. – <https://acmp.ru>, 2018.

11. Проект, позволяющий решать задачи по программированию из архива, а также участвовать в онлайн-олимпиадах по программированию [Электронный ресурс] : Timus Online Judge. – <http://acm.timus.ru>, 2018.
12. Семакин, И.Г. Информатика. 7–9 классы [Текст] : мет. пособие / И.Г. Семакин, М.С. Цветкова, – М. : БИНОМ. Лаборатория знаний, 2016. – 184с.
13. Угринович Н.Д. УМК для основной школы: 7–9 классы [Текст] : мет. пособие для учителя / И. Ю. Хлобыстова, М.С. Цветкова, – М. : БИНОМ. Лаборатория знаний, 2012. – 91с.
14. Федеральный государственный образовательный стандарт основного общего образования [Электронный ресурс]. – <http://минобрнауки.рф/документы/543>, 2012.
15. Щербаков, А.П. Программная поддержка обучения решению олимпиадных задач на обработку строк для школьников [Текст] / А.П. Щербаков, Н.Б. Паршукова / IV Всероссийская научно-практическая Интернет-конференции «Информатизация образования: проблемы и перспективы» на базе ФГБОУ ВО «Южно-Уральский государственный гуманитарно-педагогический университет». – 2018. – С. 125–130.

ПРИЛОЖЕНИЯ

Приложение 1

Разработка урока по теме “Решение легких олимпиадных задач”


Формы организации учебной деятельности обучающихся:

- фронтальная, практическая и индивидуальная работа

Перечень средств ИКТ, используемых на уроке:

- интерактивная доска;
- видеопроектор;
- презентация.


Деятельность учителя	Деятельность учащихся	Интерактивная доска (экран)
Этап 1. Организационный момент		
Здравствуйте, ребята. Можете садиться. Запишите домашнее задание: Тест: Стр. 98-99 Письменно в тетради	Приветствуют учителя. Садятся. Записывают.	Щербаков Александр Павлович

		<p>Домашнее задание</p> <p>Тест: Стр. 98-99 Письменно в тетради</p>
<p>Этап 2.</p>		
<p>Мы с вами живем в век информационных технологий. В каждом доме есть огромное количество программируемых устройств, и эти технологии продолжают проникать во все сферы нашей жизни. Сможете привести пример, как они влияют на неё? Хорошо, вот и поэтому такие соревнования как по программированию становятся все более популярным и зрелищным видом спорта, и</p>	<p>Слушают и запоминают, отвечают на вопросы.</p>	<p>Решение олимпиадных задач</p> 

<p>будущее у него очень многообещающее.</p> <p>Олимпиады по информатике начали проводиться еще до введения школьного курса информатики. С 1985 г. олимпиады по информатике приобрели массовый характер.</p> <p>Начали проводить школьные, городские, районные, всероссийские олимпиады.</p> <p>Победители всероссийских олимпиад принимают участие в международных. И т.д.</p>		
--	--	--

<p>Перед олимпиадами по программированию ставятся след задачи:</p> <ol style="list-style-type: none"> 1. Отбор наиболее способных учащихся. 2. Развитие интереса к предмету. 3. Профессиональное ориентирование учащихся. 	<p>Внимательно слушают учителя.</p>	<p style="text-align: center;">Цели олимпиады по программированию</p> <ol style="list-style-type: none"> 1. Отбор наиболее способных учащихся. 2. Развитие интереса к предмету. 3. Профессиональное ориентирование учащихся.
<p>Давайте разберемся с вами, зачем нужны олимпиады по программированию.</p> <p>Как вы думаете, зачем многие участвуют в различных олимпиадах по программированию?</p> <p>Да вы во многом правы.</p> <p>Во-первых:</p> <p>Задачи на соревнованиях по программированию, учат решать нестандартные задачи, с учетом минимального использования ресурсов</p>	<p>Внимательно слушают учителя и отвечают на вопросы.</p>	<p style="text-align: center;">Зачем нужны олимпиады по программированию</p> <ol style="list-style-type: none"> 1. Соревнования по программированию учат использовать алгоритмы. 2. Соревнования по программированию для многих становятся стартом в программирование. 3. Спортивное программирование развивает навыки писать код без багов, думать о крайних случаях, о производительности. 4. Спортивное программирование дает опыт работы в команде, коллективе. 5. Спортивное программирование может принести медали/дипломы, которые помогут при поступлении в ВУЗ. 6. Спортивное программирование дает возможность немного попутешествовать. 7. Спортивное программирование это просто очень интересно!

<p>компьютера, а так же использовать на практике различные алгоритмы. Однако спортивное программирование является массовым движением, с помощью которого учащиеся узнают, что такое динамическое программирование, структуры данных, потоки, строковые алгоритмы и многое другое. Далее</p> <p>Соревнования по программированию для многих становятся стартом в программирование.</p> <p>Спортивное программирование развивает навыки писать код без багов. Т.е. они учат сразу писать правильный код.</p> <p>Не маловажный фактор, что Спортивное программирование дает опыт работы в команде, коллективе, что сейчас очень важно.</p> <p>Спортивное программирование может</p>		
---	--	--

<p>принести медали/дипломы, которые помогут при поступлении в ВУЗ.</p> <p>Спортивное программирование дает возможность немного попутешествовать, так например, пройдя на этап выше можно отправиться в другой город или даже страну. И конечно Спортивное программирование это просто очень интересно!</p>		
<p>Ведь, не даром, Спортивное программирование ежегодно собирает больше участников, чем Олимпийские игры – в 2016 году в соревнованиях ACM ICPC принимали участие 46 381 человек из 103 стран, в то время как в летней Олимпиаде в Рио на всех этапах участвовали 11 544 спортсменов (в 4 раза меньше).</p>		<p>Финал чемпионата мира по спортивному программированию ACM ICPC</p>  <p>Спортивное программирование ежегодно собирает больше участников, чем Олимпийские игры – в 2016 году в соревнованиях ACM ICPC принимают участие 46 381 человек из 103 стран, в то время как в летней Олимпиаде в Рио на всех этапах участвовали 11 544 спортсменов (в 4 раза меньше).</p>

Россия не уступает по качеству подготовки другим странам, так в самом начале истории олимпиад, Россия не занимала особо призовые места, но в последнее время, победителями являются команды из России.

Внимательно слушают учителя.

Победители

Год	Место проведения	Победитель
2017	США, Ралид-Сити	Университет ИТМО
2016	Таиланд, Пхукет	СПбГУ
2015	Марокко, Марракеш	Университет ИТМО
2014	Россия, Екатеринбург	СПбГУ
2013	Россия, С.Петербург	Университет ИТМО
2012	Польша, Варшава	Университет ИТМО
2011	США, Орlando	Чкаловский университет
2010	КНР, Харбин	Шанхайский университет транспорта (Цзяотун)
2009	Швеция, Стокгольм	Университет ИТМО
2008	Канада, Банфф	Университет ИТМО
2007	Япония, Токио	Варшавский университет
2006	США, Сан-Антонио	Саратовский государственный университет

Начать готовиться к олимпиадам ни когда не поздно. В интернете есть множество ресурсов, где можно готовиться к олимпиадам. Описываются различные алгоритмы, есть возможность решать задачи или проверить свои силы в различных соревнованиях.

Внимательно слушают учителя.

The screenshot shows the Codeforces website interface. At the top, there's a navigation bar with 'Мои соревнования' (My Contests) and 'CODEFORCES' logo. Below it, a list of problems is visible, including '273. Ралид-Сити' and '274. Таиланд, Пхукет'. A sidebar on the right shows a 'Рейтинг (проблемы)' (Rating by Problems) table:

№	Имя	Рейтинг
1	ИТМО	2400
2	SPbGU	2200
3	ВШЭ	2150
4	Саратовский гос. ун-т	2120
5	ФУМ	2100
6	СФУ	2050
7	СФГУ	2000
8	СФУ	1950
9	СФГУ	1900
10	СФГУ	1850

Below the table, there's a section for 'Мой курс Advanced Algorithms at Harbour.Space University' with a photo of a modern building at night.

<p>Для участия в олимпиаде нужно знать хотя бы:</p> <p>базовые алгоритмические конструкции, стандартные математические функции, процедуры и функции для обработки строковых переменных, процедуры и функции для работы с массивами.</p> <p>Некоторое из данного списка, изучается в школьном курсе информатики, но большую часть нужно изучать самостоятельно.</p>	<p>Внимательно слушают учителя.</p>	<p>Минимальная база знаний для олимпиады по информатике.</p> <p>Язык программирования:</p> <ul style="list-style-type: none"> • базовые алгоритмические конструкции, • стандартные математические функции, • процедуры и функции для обработки строковых переменных, • процедуры и функции для работы с массивами. <p>Типовые алгоритмы.</p>
<p>При решении задач существуют часто встречающиеся ошибки такие как:</p> <p>Не соответствует формат ввода-вывода данных условию задачи</p> <p>Рассмотрены не все возможные случаи</p> <p>Не правильно задан тип данных (размерность)</p> <p>Потеря редактируемых файлов во время</p>		<p><u>Часто встречающиеся ошибки:</u></p> <ul style="list-style-type: none"> • Не соответствует формат ввода-вывода данных условию задачи • Рассмотрены не все возможные случаи • Не правильно задан тип данных (размерность) • Потеря редактируемых файлов во время тура

<p>тура</p> <p>Давайте разберем некоторые из них поподробней.</p>		
<p>При вводе данных нельзя выводить подсказки на ввод.</p> <pre>write('Уважаемый компьютер, введите, пожалуйста, значение n:'); { неправильно! } read(n); read(n); { правильно }</pre> <p>При выводе результатов пояснительные сообщения также выводить нельзя, кроме тех, которые явно указаны в формате выходных данных в условии задачи.</p> <pre>writeln('Результат=',rez); { неправильно! } writeln(rez); { правильно }</pre>		<p>При вводе данных нельзя выводить подсказки на ввод.</p> <pre>write('Уважаемый компьютер, введите, пожалуйста, значение n:'); { неправильно! } read(n); read(n); { правильно }</pre> <p>При выводе результатов пояснительные сообщения также выводить нельзя, кроме тех, которые явно указаны в формате выходных данных в условии задачи.</p> <pre>writeln('Результат=',rez); { неправильно! } writeln(rez); { правильно }</pre>

Так же при решении задач часто получается, так что задача проходит не по всем тестам, что это значит? А значит то, что у каждой задачи есть набор тестов, на который проверяется та или иная задача. И если решение не проходит по одному из тестов задача является не решенной. Например

№	Ввод	Вывод
1	10 5	15
2	5 10	15
3	0 0	0
4	0 -10	-10
5	152 352	504
6	153 0	153
7	-150 -645	-795
8	0 -9	-5
9	310 1000	1310
10	5 -5	0

Тест	Результат	Время	Память
1	Accepted	0,029	56 КБ
2	Accepted	0,008	56 КБ
3	Accepted	0,011	56 КБ
4	Accepted	0,008	56 КБ
5	Accepted	0,008	56 КБ
6	Accepted	0,008	56 КБ
7	Accepted	0,011	56 КБ
8	Accepted	0,008	56 КБ
9	Accepted	0,008	56 КБ
10	Accepted	0,009	56 КБ

Важно отметить, что **текст задачи нужно всегда внимательно читать** от начала и до конца, поскольку ключевое условие может быть спрятано, например, в формате входных или выходных данных, а также в приведенных примерах файлов входных и выходных данных.

При разработке программы следует также обратить особое внимание на описание **формата входных и выходных данных**, приведенное в условии задачи. Имена

Важно отметить, что **текст задачи нужно всегда внимательно читать** от начала и до конца, поскольку ключевое условие может быть спрятано, например, в формате входных или выходных данных, а также в приведенных примерах файлов входных и выходных данных.

При разработке программы следует также обратить особое внимание на описание **формата входных и выходных данных**, приведенное в условии задачи. Имена входного и выходного файлов также описаны в условии задачи, и неправильное их написание в программе считается ошибкой.

Необходимо помнить при написании программы, — это **сохранение редактируемых файлов** во время тура.

Полученная программа должна соответствовать заданной **размерности входных данных** и удовлетворять ограничениям на **память и время работы**, заданные в условии задачи.

<p>входного и выходного файлов также описаны в условии задачи, и неправильное их написание в программе считается ошибкой.</p> <p>Необходимо помнить при написании программы, — это сохранение редактируемых файлов во время тура.</p> <p>Полученная программа должна соответствовать заданной размерности входных данных и удовлетворять ограничениям на память и время работы, заданные в условии задачи.</p>		
<p>Прежде, чем переходить к решению задач, давайте разберем, как работать в системы для автоматической проверки задач.</p> <p>Перед работой на сайте, сначала необходимо зарегистрироваться.</p> <p>Необходимо указать: Фамилию Имя, Логин, Класс и Пароль.</p>	<p>Внимательно слушают учителя и отвечают на вопросы.</p>	

<p>После Регистрации, вас доступен список задач, выбираете из списка задачу и решаете её. После того как решите её вставляете код программы в поле для кода и нажимаете кнопку сохранить и отправить.</p> <p>Если программы выдаст сообщение ОК, можете переходить к следующей задаче.</p> <p>Иначе перепроверьте правильность решения задачи. И повторите отправку.</p>		
--	--	--

Этап 3. Практическая работа.

Сейчас давайте с вами перейдем к решению олимпиадных задач с использованием системы для автоматической проверки задач.

За решение задач вы получите оценки.

Давайте разберем первую задачу, прочитайте условие её и подумайте, как её можно решить?

Еще варианты?

Эта задача скорее является пародией на класс легких задач со сложной формулировкой, т.к. уж слишком она простая. Дело в том, что часто на олимпиадах попадается одна задача, имеющая легкое решение, но пугающая участников своей сложной формулировкой. Так и здесь: в задании имеется масса текста,

Слушают, отвечают на вопросы, решают задачи.

Решение олимпиадных задач

<p>включающего какие-то рекуррентные математические формулы, наводящие на мысль о том, что задача может быть серьезной. А на самом-то деле оказывается, что нужно считать число и вывести его же. Разве может быть еще что-то легче?</p> <p>Поэтому иногда полезно, увидев большой текст задания обратить внимание на то, что записано в разделах входных и выходных данных, иногда просто этого оказывается достаточно для решения задачи.</p> <p>Остальные задачи попробуйте решить сами в оставшееся время, задачи которые вы не сможете решить, разберем на следующем уроке.</p>		
--	--	--

Этап 4. Подведение итогов.

Спасибо за ваше внимание. Все молодцы!

Какие были сложности?

Какая задача была самая сложная?

На следующем уроке разберём с вами задачи, которые вы не смогли решить, а так же интересные моменты в каждой из каждой задачи. Если нет больше вопросов то урок окончен, все свободны.

Дети отвечают на вопросы и собираются уходить на перемену.

Спасибо за внимание!