

Министерство образования и молодежной политики Свердловской области
Управление образования Артемовского городского округа
Муниципальное бюджетное общеобразовательное учреждение
«Средняя общеобразовательная школа № 9»
Уральское ГУ банка России г. Екатеринбург
Свердловская область, Артемовский район, п. Буланаш, ул. Комсомольская, 21,
тел. 5-52-50,
e-mail: schola9@yandex.ru



УТВЕРЖДАЮ

Директор МБОУ «СОШ №9»

А.А. Ларионова

«01» июня 2022 г.

Приложение
к основной образовательной программе
среднего общего образования
МБОУ «СОШ №9»
(в редакции от 01.06.2022 г)

**ДОПОЛНИТЕЛЬНАЯ
ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ
ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА
«ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ
PYTHON»
НАПРАВЛЕННОСТЬ: ТЕХНИЧЕСКАЯ
10-11 КЛАСС**

Пояснительная записка

Программа дополнительного образования «Программирование на языке Python» направлена на технический профиль, более глубокое изучение тем, которые рассматриваются в курсе информатики 10 и 11 классов по информатике, подготовку к соревнованиям и олимпиадам по информатике, достижение опыта решения нестандартных и креативных заданий по информатике и развитие учебной и творческой мотивации.

Специфика программирования как синтетической науки, включающей элементы математики, философии, стилистики, психологии и инженерии, состоит именно в том, что преобладающий способ реализации ее собственных знаний — это работа со знаниями более «конкретных» дисциплин (предметных областей).

Сочетание обучения методам, а не фактам, работы на примерах из смежных школьных дисциплин, универсальности большинства изучаемых приемов превращает, таким образом, информатику в дисциплину, максимально интегрирующую предметы школьного курса. Такая концепция приводит к повышению эффективности процесса обучения в целом.

Необходимо подчеркнуть, что в ряде курсов наиболее существенным приобретением ученика становятся не сколько знания, умения и навыки, а метаумения и метанавыки, определяющие правильное формирование подхода к проблеме, способа ее рассмотрения, поиска решения и реализации, а в конечном счете и организацию знания, получаемого ребенком. Это является следствием системного подхода, применяемого как непосредственно в процессе обучения, так и при планировании занятий. В конечном счете, речь идет о выработке у ученика элементов системной методологии мышления. При полном отсутствии этих компонент получается стандартное выучивание и воспроизведение, при полном наличии — в идеале становится ненужным традиционный учебный процесс (ребенок способен получать и организовывать материал самостоятельно). В ходе обучения достигается компромисс между этими формами, он определяется индивидуально для каждого ребенка, в зависимости от его мотиваций и склонностей.

Во время обучения проводятся занятия в лекционной, практической форме, кроме этого в форме личностных и командных соревнований, где обучающиеся могут показать полученные на занятиях знания, получить опыт участия в соревнованиях по программированию.

В программе рассматривается один из актуальных языков программирования с применением классических и улучшенных алгоритмов обработки данных. К особенностям программы можно отнести, что кроме обучающихся демонстрирующих успехи в технической направленности, могут обучаться и другие, обучаясь справочному сопровождению функций и самой программы. Для успешного обучения рекомендуется знать основы алгоритмизации, которым обучаются в 7 классе, а так же основам работы со школьным языком программирования Кумир.

Программа рассчитана на 35 учебных недель (по 2 часа в неделю), в дальнейшем можно будет приступить к продвинутому уровню обучения языку программирования Python и написанию простейших программ.

Итоги обучения по данной программе оцениваются успешностью выступления обучающихся в различных олимпиадах по программированию: Всероссийская олимпиада школьников по информатике, Технокубок и другие.

Актуальность и новизна программы

В настоящее время проектная составляющая в школьном обучении программированию сильно уменьшилась за последние 15-20 лет, превратившись в моделирование в электронных таблицах (MS Excel, Libre office Calc, Open office), знанию возможностей MS PowerPoint, основы языка гипертекстовой HTML, систему управления базами данных, работе в растровых и векторных редакторах и, к сожалению, уступив в

массовости олимпиадам, подразумевающим небольшой объем каждой задачи и крайне малое время жизни кода. Изучение основ каждой дисциплины невозможно полностью раскрыть за урочное время, а также учитывая загруженность детей другими уроками, обучение программированию нужно выделять в отдельный предмет, чтобы не изучать его малыми дозами, к тому же на внеурочные занятия по программированию будут приходить заинтересованные ученики, которые хотят узнать интересные особенности проходимых операторов.

Работа над проектными задачами, подразумевающими не только логику и расчеты, но и визуализацию, требует от учеников гораздо больших навыков и даже совсем другого стиля мышления, чем сейчас дается школьникам как в общеобразовательном курсе программирования, читающихся в огромном количестве на разных курсах дополнительного образования.

Здесь можно присоединиться ко многим вузовским преподавателям, физикам, биологам, лингвистам и другим профессиональным представителям предметных областей, да и самим программистам, утверждающим, что сейчас даже среди школьников физико-математической направленности стали редки те, кто может взять задачу за пределами урока по информатике при изучении раздела программирование и грамотно решить ее, не переписывая по сто раз и не теряя понимание собственного кода программы. Такая тенденция удручает.

Актуальность подхода заключается в длительном характере работы над проектом (в течение 2 недель - 1 месяца) и сравнительно большой для школьников объем кода проекта (более 100 строк). Это объясняется тем, что в работах будет присутствовать как расчетная, так и интерфейсная часть, которая почти всегда составляет большую часть кода проекта, что вообще характерно для визуализации. Такое положение дел неизбежно приводит к необходимости строгой архитектуры проекта, что зачастую игнорируется на уроках информатики по программированию. «Традиционный» подход, где эти части смешиваются, принципиально неприменим для задач, решение которых требует больше 1-2 учебных часов и значительный объем кода, а это имен тот тип задач, который характерен для вузов и для профессиональной деятельности, и к которому надо готовить школьника, чтобы в дальнейшем не пришлось его переучивать.

Новизна подхода заключается в большом упоре на технологическую грамотность проекта и его проработанность с точки зрения современных технологий программирования. Зачастую проектные работы, выполненные школьниками, смотрятся неплохо с точки зрения интерфейса, но их внутренняя архитектура, надежность, технологическая грамотность на всех уровнях оставляют желать много и много лучшего, что обычно объясняется тем, что в этих проектах «важнее всего физика, а не программирование». На практике же это обычно означает, что школьник (а далее и студент) не может написать и отладить программу большего объема даже на аналогичную тему, путается в коде и в отладке, и в конце концов бросает проект, а плохое качество кода не дает его последователям в полной мере использовать разработанные методики моделирования. Чтобы научиться работать с проектами не школьного уровня в будущем, ему необходимо серьезно переучиваться и менять стиль мышления при программировании, что удается не каждому. Остальные вместо самостоятельной работы над задачей «нанимают программистов» и вынуждены тратить время и силы на постановку им технических заданий и верификацию данных, и финансы на собственно разработку. Таким образом, такой исследователь, как ученый-физик, -математик, -биолог, -химик становятся несамостоятельными, что ограничивает эффективность их работы. Используемый подход, основанный на принципе «разрабатывать небольшие учебные проекты так, как это делается в больших», призван устранить эту часто встречающуюся проблему

Цели и задачи обучения

К общей цели обучения курсу в контексте основного образования с учетом специфики учебного предмета является формирование универсального стиля научного и практического мышления, основанного на системном подходе, направленного на самостоятельное творческое познание и исследование окружающего мира. Это конкретизируется в виде изучения общих закономерностей функционирования, создания и применения крупных программ, разрабатываемых учениками.

Общеучебные задачи курса

1. Организация процессов мышления и познания.
 - Обучение наиболее общим приемам работы с информацией, пригодных для использования в любой предметной области.
 - Интеграция программ школьного курса, взаимное усиление информатики и других школьных предметов по принципу: технология работы с информацией — из информатики, конкретные примеры и задачи — из смежных предметов.
2. Развитие самостоятельности в работе и творческой активности.
3. Усвоение и развитие технологии рационального решения задач:
 - Определение адекватных способов решения учебной задачи на основе заданных алгоритмов
 - Комбинирование известных алгоритмов деятельности в ситуациях, не предполагающих стандартное применение одного из них.
 - Разработка собственных алгоритмов решения задач.
 - Использование для решения познавательных и коммуникативных задач различных источников информации.
 - Владение умениями совместной деятельности (согласование и координация деятельности с другими ее участниками, объективное оценивание своего вклада в решение общих задач коллектива; учет особенностей различного ролевого поведения).

Внутрипредметные общие задачи курса

1. Освоение знаний, составляющих основу научных представлений об информации, информационных процессах, системах, технологиях и моделях.
2. Овладение умениями работать с различными видами информации с помощью компьютера и других средств информационных и коммуникационных технологий, организовывать собственную информационную деятельность и планировать её результат.
3. Выработка навыков применения средств ИКТ в повседневной жизни, при выполнении индивидуальных и коллективных проектов, в учебной деятельности.
4. Развитие познавательных интересов, интеллектуальных и творческих способностей средствами ИКТ.
5. Воспитание ответственного отношения к информации с учетом правовых и этических аспектов её распространения, избирательного отношения к полученной информации.
6. Выработка навыков применения средств ИКТ в повседневной жизни, при выполнении индивидуальных и коллективных проектов, в учебной деятельности, дальнейшем освоении профессий, востребованных на рынке труда.

Ценностные ориентиры содержания учебного предмета

Важнейшей частью курса является формирование системы учебных ценностей (предпочтений) ученика. В конечном счете, это формирование и есть основная инвариантная методологическая задача курса, так как все остальное — технология и будет неотвратимо изменяться с течением времени.

1. Во главу угла ставится задача, ее решение и, главное, путь от задачи к решению, а не программирование (кодирование) как таковое.

2. Для записи алгоритма на языке программирования выбирается минимальное подмножество средств языка, чтобы не акцентировать внимания на кодировании и для более легкого перехода на другие языки программирования.

3. Самостоятельность решения является ключевым условием, которое необходимо доказать при сдаче работы.

4. Понимание учащимся тех средств, с помощью которых он решил задачу, ставится выше уровня самих средств решения.

5. Аккуратность и надежность решения ставятся выше «программистских трюков», иногда позволяющих в отдельных случаях добиться несколько лучших результатов.

6. Главным методологическим принципом является системный подход.

Принципы технологии преподавания

1. Принцип самостоятельного исследования объектов и явлений с точки зрения системного подхода. Методы подачи материала:

- Обучение навыкам самостоятельной работы с информацией
- Техника и технология информационного поиска («где что?»).
- Определение структуры системы понятий («что к чему относится?»).
- Поиск базовой системы понятий («что главное?»).
- Связь с другими предметными областями («где еще?»).
- Исследование «черных ящиков» (приборов, программ, учебных тем).
- Использование знаний об аналогах и более общих объектах.
- Выявление общих принципов устройства, работы.
- Выявление характерных свойств объектов и иерархия этих свойств.

2. Принцип творчества, для:

- Закрепления полученных теоретических навыков.
- Эмоционального подкрепления в ходе творческой работы.
- Организации и планирования проектов, обучения планированию.
- Взаимодействия учеников в коллективных проектах.

3. Принцип эмоционального подкрепления, для:

- Увеличения эффективности усвоения.
- Развития уверенности в своих силах.
- Развития эстетики.

4. Принцип технологичности:

– Постоянное стремления повышать КПД своей работы.
– Умение работать над конкретной задачей в контексте ее возможного развития.

– Развитие технология решения задачи вместе с ее собственно решением.
– Обучение технологии на сравнении «плохих» и «хороших» примеров (от первых — ко вторым).

– Формирование технологически корректной системы ценностей ученика на уровне «люблю работать высокоэффективно» и «не люблю рутинные и ненадежные действия, ищу, как сделать это лучше».

5. Принцип смежных дисциплин:

– Разбор материала из смежных предметов, работа на примерах из смежных областей.

– Взаимное усиление информатики и смежных дисциплин.

– Интеграция смежных дисциплин.

6. Принцип открытой педагогической технологии:

– Открытая и ясная система ценностей, установок.
– Подчеркнутая расстановка акцентов в подаче материала.
– Понимание «критических мест» в ходе изучения и способов их преодоления.

– Понимание связи материала и методики его подачи.

- Активное использование взаимодействия учеников и взаимного обучения

Результаты освоения курса

Результатом прохождения курса учащимися должны стать понимание основных принципов программирования, его применения для решения различных межпредметных задач и владение основными необходимыми для этого алгоритмическими конструкциями.

Личностные результаты

– Владение навыками анализа и критичной оценки получаемой информации с позиций ее свойств, практической и личной значимости, развитие чувства личной ответственности за результат.

– Оценка окружающей информационной среды и формулирование предложений по ее улучшению.

– Организация индивидуальной информационной среды, в том числе с помощью применения средств программирования и автоматизации.

– Использование различных источников информации различных типов для повышения своего образовательного уровня и подготовке к продолжению обучения.

Метапредметные результаты

– Владение основными общеучебными умениями информационного характера: анализа ситуации, планирования деятельности, обобщения и сравнения данных и другими.

– Получение опыта использования методов и средств информатики: моделирования; формализации структурирования информации; компьютерного эксперимента при исследовании различных объектов, явлений и процессов.

– Умение создавать и поддерживать индивидуальную информационную среду, обеспечивать защиту значимой информации и личную информационную безопасность.

– Владение навыками работы с основными, широко распространенными средствами информационных и коммуникационных технологий.

– Умение осуществлять совместную информационную деятельность, в частности при выполнении проекта.

Предметные результаты

В сфере познавательной деятельности

– Освоение основных понятий и методов программирования и компьютерного моделирования.

– Выделение основных информационных процессов в реальных системах, нахождение сходства и различия протекания информационных процессов в различных предметных областях.

– Выбор языка представления информации в соответствии с поставленной целью, определение формы представления информации, отвечающей данной задаче (таблицы, схемы, графы, диаграммы и др.).

– Преобразование информации из одной формы представления в другую без потери смысла и полноты информации.

– Оценивание информации с позиций ее свойств (достоверность, объективность, полнота, актуальность и т.п.).

– Развитие представлений об информационных моделях как основном инструменте познания, общения, практической деятельности, знания основных областей применения метода моделирования.

– Разработка и запись алгоритмов, т.е. построение модели решения задачи с применением основных алгоритмических конструкций для описания алгоритмов, проверка правильности алгоритма, нахождение и исправление типовых ошибок.

– Освоение основных конструкций языка программирования.

- Знание основных свойств алгоритма, типов алгоритмических конструкций: следование, ветвление, цикл; понятия вспомогательного алгоритма и других.
- Знание программного принципа работы компьютера.
- Умение выполнять базовые операции над объектами: цепочками символов, числами, списками, деревьями; проверять свойства этих объектов; выполнять и строить простые алгоритмы.
- Решение различных задач из разных сфер человеческой деятельности с помощью средств информационных технологий.
- Приобретение опыта использования информационных ресурсов общества и электронных средств связи в учебной и практической деятельности.

В сфере ценностно-ориентационной деятельности

- Понимание роли информационных процессов как фундаментальной реальности окружающего мира и определяющего компонента современной информационной цивилизации.
- Умение выделять критерии оценки информации, получаемой из разных источников.
- Оценка информации, в том числе получаемой из средств массовой информации, свидетельств очевидцев, интервью: использование ссылок и цитирование источников информации, анализ и сопоставление различных источников.
- Выявление проблем жизнедеятельности человека в условиях информационной цивилизации и оценка предлагаемых путей их разрешения, умение пользоваться ими для планирования собственной деятельности.

В сфере коммуникативной деятельности

- Понимание особенностей представления информации различными средствами коммуникации (на основе естественных, формализованных и формальных языков).
- Осознание основных психологических особенностей восприятия информации человеком.
- Владение навыками использования основных средств телекоммуникаций, формирования запроса на поиск информации в Интернет с помощью программ навигации (браузеров) и поисковых программ, осуществления передачи информации по электронной почте и др.

В сфере трудовой деятельности

- Понимание принципов действия различных средств информатизации и автоматизации, их возможностей и ограничений.
- Владение навыками использования технических средств информационных технологий для решения различных задач.
- Знакомство с основными программными средствами компьютера (круг решаемых задач, система команд и пр.).
- Умение тестирования используемого оборудования и программных средств.
- Выбор соответствующего средства информационных технологий для решения поставленной задачи.

В сфере физической деятельности

- Понимание особенностей работы со средствами информатизации, их влиянии на здоровье человека.
- Владение профилактическими мерами при работе с средствами информатизации.
- Соблюдение требований безопасности и гигиены в работе с компьютером и другими средствами информационных технологий.

Цели курса 10-11 класс.

Основная цель курса: подготовка обучающихся 10-11 классов к олимпиадам по программированию. Кроме данной цели, изучение курса в 10-11 классе даст возможность достичь результатов:

Личностного направления:

- 1) Способность четко, ясно и грамотно высказывать собственные идеи в письменной и устной форме, понимать смысл поставленной проблемы, выстраивать собственную аргументацию, приводить контр примеры и примеры;
- 2) Критичность мышления, умение определять некорректные логически высказывания, разграничивать факты и гипотезы;
- 3) Развитие инициативы, находчивости, активности при решении сложных математических задач, творческого мышления;
- 4) Приобретение умения работы в команде, брать на себе разные роли командной работы, видеть слабые свои и чужие слабые стороны;

Предметного направления:

- 1) Обучение владению базовым понятийным аппаратам по основным темам программирования
- 2) Обучение работать со справочной информацией и мануалами (извлекать, анализировать и применять необходимую информацию), проводить классификацию, ясно и грамотно выражать свои мысли и идеи в письменной и устной форме с использованием терминологии
- 3) Развитие умения понимания и использования средств наглядности (презентации, видео, схемы и т.д.) для иллюстрации, аргументации.

Задачи курса.

- 1) Изучить практические основы программирования. В ходе обучения предстоит научиться писать самостоятельно программный код, анализировать готовые программы, читать чужой программный код.
- 2) Обучиться умению представлять как будет работать написанный текст программы
- 3) Обучиться тестированию программы в мысленном режиме, без использования систем программирования
- 4) Обучиться строить гипотезы о неработоспособности кода программы, проверять ее, а также вносить исправления и доработки в готовый код.

Содержание программы.

1. Типы данных. Переменные. Линейные программы. Обучение работы оператора ввода и вывода, форматному вводу и выводу. Знакомство с понятием переменная (именование переменной, значение переменной, ссылка на значение), с помощью которой храниться информация. Изучение целых, вещественных, логических типов данных, списков и массивов в Python. Работа с разными типами данных, а так же множество допустимых с ней операций и множество принимаемых значений.

2. Условный оператор. Обучение работы с полной и сокращенной формой ветвления. Изучение сложных условий, применение логических переменных для работы с тестом программы.

3. Операторы циклов. Один самых важных модулей. В нем разбирается цикл с предусловием while, цикл со счетчиком for (старт, стоп, шаг). Рассмотрение тонкостей применения условного оператора и его прерывания.

4. Массивы. Обучение работы со списками и массивами. Применение массивов к классическим алгоритмам. Массивы с известным количеством элементов и динамические массивы. Алгоритмы обработки потока данных.

Содержание учебно-тематического плана рабочей программы 10-11 класса.

№ п/п	Тема	Количество часов
1	Типы данных. Линейные программы. Локальные переменные	13
2	Условный оператор. Разветвляющиеся программы	15
3	Операторы циклов. Программирование циклических алгоритмов	20
4	Массивы. Программирование с использованием массивов.	22
5	Функции. Локальные и глобальные переменные. Программирования с использованием функций	12
6	Строковые переменные. Программирования с использованием строковых переменных	12
7	Объединение функций и добавление их в библиотеку. Понятие структуры.	8
8	Экспертные системы. Создание экспертных систем по физике и математике	38

Тематическое планирование рабочей программы.

№ п/п	Тема урока (количество часов)	Основные понятия	Основные виды деятельности ученика	Примечание
1.	Введение в программирование на Python. (1)	Программирование, программист, программа, комментарий, оператор, вывод на экран, система программирования, транслятор, компилятор, интерпретатор, отладчик		
2.	Линейные программы. Ввод переменных. Работа с переменными. Сумма чисел. (2)	Линейная программа, переменная, идентификатор, ввод чисел, оператор присваивания, список вывода, сравнение ссылки и присваивания в Python	Практическая работа №1 «Линейные программы»	
3	Линейные программы. Арифметические выражения. Операции с целыми переменными. Форматный вывод. (2)	Арифметическое выражение, приоритет операций, форматный вывод.	Практическая работа №2 «Операции деления»	
4	Линейные программы. Операции с вещественными переменными. (2)	Вещественные переменные	Практическая работа №3 «Операции с вещественными числами»	
5	Линейные программы. Случайные и псевдослучайные числа. (2)	Случайные числа, псевдослучайные числа	Практическая работа №4 «Случайные числа»	

№ п/п	Тема урока (количество часов)	Основные понятия	Основные виды деятельности ученика	Примечание
6	Отработка навыков решения задач на программирование линейных алгоритмов (2)		Практические задания	
7	Решение олимпиадных заданий с использованием линейных алгоритмов. (2)			
8	Разветвляющие программы. Условный оператор. Неполная форма условного оператора. (2)	Условный оператор, полная форма условного оператора, неполная форма условного оператора	Практическая работа №5 «Ветвления»	
9	Разветвляющие программы. Сложные условия. (2)	Сложное условие, операции «И», «ИЛИ», «НЕ»	Практическая работа №6 «Логические переменные. Сложные условия»	
10	Разветвляющие программы. Составной оператор. Вложенный условный оператор. (2)	Составной оператор. Вложенный условный оператор.	Практическая работа №7 «Вложенные условия»	
11	Разветвляющие программы. Логические переменные. Порядок выполнения логических операций. (2)		Практическая работа №8 «Логические переменные»	
12	Разветвляющиеся программы. Экспертная система. (3)	Экспертная система	Практическая работа №9 «Экспертная система»	
13	Отработка навыков решения задач на программирование разветвляющихся алгоритмов (2)		Практические задания	
14	Решение олимпиадных заданий с использованием разветвляющихся алгоритмов.			

№ п/п	Тема урока (количество часов)	Основные понятия	Основные виды деятельности ученика	Примечание
	(2)			
15	Программирование циклических алгоритмов. Организация цикла. (2)	Цикл, счётчик шагов цикла. цикл с условием, алгоритм Евклида, цикл по переменной, переменная цикла, старт, стоп, шаг цикла, список	Практическая работа №10 «Циклы с условием. Часть 1»	
16	Программирование циклических алгоритмов. Циклы с условием. Алгоритм Евклида (2)		Практическая работа №12 «Алгоритм Евклида»	
17	Программирование циклических алгоритмов. Обработка потока данных. (2)		Практическая работа №13 «Обработка потока данных»	
18	Программирование циклических алгоритмов. Циклы по переменной. Шаг изменения переменной цикла. (3)		Практическая работа №14 «Циклы с постусловием» Практическая работа №15 «Циклы по переменной»	
19.	Программирование циклических алгоритмов. Алгебраическая прогрессия. (5)		Практическая работа №16-19 «Применение циклических программ 1, 2, 3, 4 части»	
20	Отработка навыков решения задач на программирование циклических алгоритмов (4)			
21	Решение олимпиадных заданий с использованием циклических алгоритмов. (2)			
22	Массивы. Общие понятия об одномерных массивах. Создание массива.	Массив, индекс элемента, значение элемента, заполнение массива, вывод массива, ввод	Практическая работа №20 «Заполнение массивов»	

№ п/п	Тема урока (количество часов)	Основные понятия	Основные виды деятельности ученика	Примечание
	(2)	массива		
23.	Массивы. Обращение к элементу массива. Перебор элементов массива. (2)			
24	Массивы. Генератор массива. Вывод массива. (2)		Практическая работа №21 «Перебор элементов массива»	
25	Массивы. Ввод массива с клавиатуры. Заполнение массива случайными числами. (2)		Практическая работа №22 «Заполнение массива случайными числами»	
26	Алгоритмы обработки массивов. Сумма элементов массива. (3)		Сумма элементов массива, максимальный элемент, подсчёт элементов, сортировка, виды сортировок	Практическая работа №23 «Алгоритмы обработки массивов» Практическая работа №24 «Нахождение суммы элементов массива»
27	Алгоритмы обработки массивов. Подсчёт элементов массива, удовлетворяющих условию. (2)		Практическая работа №25 «Подсчёт элементов массива»	
28	Алгоритмы обработки массивов. Списки в Python. (1)			
29	Алгоритмы обработки массивов. Поиск минимального элемента в массиве. (2)		Практическая работа №25 «Поиск минимального элемента»	
30	Отработка навыков решения задач на программирование массивов. (2)			

№ п/п	Тема урока (количество часов)	Основные понятия	Основные виды деятельности ученика	Примечание
31	Решение олимпиадных заданий с использованием массивов. (4)			

Описание учебно-методического материала и материально-технического обеспечения образовательного процесса.

Предлагаемая программа составлена с соответствии с требованиями ФГОС среднего общего образования, в состав учебно-методического комплекса входят:

методическое пособие для учителя;

сетевая методическая служба авторского коллектива для педагогов на сайте издательства (<http://lbz.ru/>);

справочник по программированию на языке Python 3;

материалы для практических работ;

коллекция олимпиадных заданий по информатике.

Комплектация компьютерного класса.

В компьютерном классе установлено 10 компьютеров (рабочих мест) для обучающихся и 1 компьютер для учителя. Компьютеры объединены в локальную сеть с возможностью выхода в Интернет.

Характеристики компьютеров:

Кроме того в кабинете установлены:

документ-камера;

проектор;

сканер;

интерактивная доска.

Программное обеспечение компьютеров:

среда программирования Python;

текстовый редактор;

офисный пакет приложений.