

ОАНО «Школа «ЛЕТОВО»

«СОГЛАСОВАНО»
на заседании педагогического совета
Протокол №1 от 25 августа 2025 г.

«УТВЕРЖДЕНО» Приказом Директора
ОАНО «Школа «ЛЕТОВО»
№ 138-ОД от 26 августа 2025

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

Направленность программы: техническая

Название программы

Олимпиадные тренировки по программированию

Возраст обучающихся: 12 - 18 лет

Срок реализации программы: 1 год

Составители:

Учитель информатики Деникин А.В.

Учитель информатики

Гуровиц Владимир Михайлович

Подразделение:

Кафедра информационных технологий и дизайна

Москва, 2025

1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Олимпиадные тренировки» адресована учащимся 12- 18 лет, проявляющим интерес к программированию.

Алгоритмы и программирование в информатике занимает одну из важнейших позиций. Основной целью обучения программированию является формирование и развитие алгоритмического мышления. Сформировать умение свободно сформировать цель, описать её математическую модель и перенести на алгоритмический язык при этом получить решение для поставленной задачи. При этом решение задач не ограничивается какой-то конкретной областью, что должно позволить применять знания в любой требуемой области.

Целью программы является формирование алгоритмического мышления учащихся, формирование практических навыков по реализации алгоритмов различных типов на языке C++ как базовый язык, позволяющий понять принцип алгоритмизации и программирования в целом. Так как C++ один из ведущих языков, его обучение раскрывает огромные возможности перед обучающимися.

Ещё одной целью является подготовка к объектно-ориентированной парадигме которая так же входит в данный курс. Так же решение на стандартных задачах формирует способности, позволяющие решать задачи высокого уровня (олимпиады, соревнования, ЕГЭ).

Достижение поставленной цели связывается с решением следующих задач:

Обучающие:

изучить основные базовые алгоритмические конструкции;

изучить основные базовые приемы программирования;

освоить основные этапы решения задачи;

сформировать навык разработки, тестирования и отладки несложных программ;

сформировать навык выполнения проекта для решения конкретной задачи.

Развивающие:

развить интереса к программированию у учащихся;

пропедевтическая подготовка к олимпиадам по программированию;

развивать творческое воображение, математическое и образное мышление учащихся;

развивать навыки планирования проекта, умение работать в группе;

выявить талантливых и способных учащихся, делающих успехи в освоении программирования.

Воспитывающие:

воспитывать интерес к программированию;

воспитывать культуру общения между учащимися;

воспитывать культуру безопасного труда при работе за компьютером.

Отличительные особенности программы

Программа «Олимпиадные тренировки» будет способствовать:

- Развитию логического мышления, формированию алгоритмизации действий как на компьютере, так и в реальной жизни.
- В основе любого псевдокода лежит физическая, математическая или биологическая модель, которая позволяет получать в образовании уверенные метапредметные результаты.
- Ознакомление с новыми технологиями ИКТ и информационного общества

В программе рассматриваются базовые приемы программирования, такие как написание псевдокода, объявление переменных, вычисление выражений, использование ветвлений и циклических конструкций, при этом осваиваются приемы создания простых программ. Особенностью изложения

материала является одновременное изучение алгоритмической конструкции и ее программирование на C++. Данный способ изложения материала был опробован в течение нескольких лет при изучении темы «Алгоритмы и программирование» в рамках предмета «Информатика», показав себя наиболее рациональным.

Адресат программы: учащиеся 12-18 лет.

Форма обучения: очная.

Особенности организации образовательного процесса: основная форма работы на занятии групповая.

Срок реализации программы: 1 год.

Общее количество часов - 102, количество часов в неделю – 3 академических часа.

Уровень сложности программы

Данная программа базового уровня сложности.

Планируемые результаты освоения обучающимися программы

«Олимпиадные тренировки»

Предметные результаты

Программа способствует:

- развитию общеучебных умений и навыков на основе средств и методов информатики и ИКТ, в том числе овладению умениями работать с различными видами информации, самостоятельно планировать и осуществлять индивидуальную и коллективную информационную деятельность, представлять и оценивать ее результаты;
- целенаправленному формированию таких общеучебных понятий, как «информация», «сообщение», «данные», «алгоритм», «программа» и др.;
- воспитанию ответственного и избирательного отношения к информации; развитию познавательных, интеллектуальных и творческих способностей учащихся.

Личностные результаты

У обучающегося будут сформированы:

представления об информации как важнейшем стратегическом ресурсе развития личности, государства, общества;

понимание роли информационных процессов в современном мире;

владение первичными навыками анализа и критичной оценки получаемой информации;

ответственное отношение к информации с учетом правовых и этических аспектов ее распространения;

развитие чувства личной ответственности за качество окружающей информационной среды;

способность и готовность к принятию ценностей здорового образа жизни за счет знания основных гигиенических, эргономических и технических условий безопасной эксплуатации средств ИКТ.

Обучающийся получит возможность для формирования:

готовности и способности к саморазвитию и реализации творческого потенциала в духовной и предметно продуктивной деятельности за счет развития их образного, алгоритмического и логического мышления;

готовности к повышению своего образовательного уровня и продолжению обучения с использованием средств и методов информатики и ИКТ;

интереса к информатике и ИКТ, стремление использовать полученные знания в процессе обучения другим предметам и в жизни;

стремление использовать полученные знания в процессе обучения другим предметам и в жизни;

способность увязать учебное содержание с собственным жизненным опытом и личными смыслами,

Метапредметные результаты

Регулятивные:

Обучающийся научится:

самостоятельно обнаруживать и формулировать проблему в индивидуальной учебной деятельности; выдвигать версии решения проблемы, осознавать конечный результат, выбирать средства достижения цели из предложенных или их искать самостоятельно; составлять (индивидуально или в группе) план решения проблемы (выполнения творческой задачи); подбирать к каждой проблеме (задаче) адекватную ей теоретическую модель; работая по предложенному или самостоятельно составленному плану, использовать наряду с основными и дополнительные средства (справочная литература, сложные приборы, компьютер); работать по самостоятельно составленному плану, сверяясь с ним и с целью деятельности, исправляя ошибки, используя самостоятельно подобранные средства (в том числе и Интернет); уметь оценить степень успешности своей индивидуальной образовательной деятельности; давать оценку своим личностным качествам и чертам характера («каков я»), определять направления своего развития («каким я хочу стать», «что мне для этого надо сделать»).

учитывать правило в планировании и контроле способа решения;

осуществлять итоговый и пошаговый контроль по результату;

оценивать правильность выполнения действий на уровне адекватной ретроспективной оценки;

вносить необходимые коррективы в действие после его завершения на основе учета характера сделанных ошибок; различать способ и результат действия;

Обучающийся получит возможность научиться:

выполнять действия в опоре на заданный ориентир;

воспринимать мнение и предложения (о способе решения задачи) сверстников;

в сотрудничестве с учителем, группой находить несколько вариантов решения учебной задачи;

самостоятельно оценивать правильность выполнения действия и вносить необходимые коррективы в действия с наглядно образным материалом.

Познавательные:

Обучающийся научится:

осуществлять поиск необходимой информации для выполнения учебных заданий с использованием учебной литературы;

строить речевые высказывания в устной и письменной форме;

проводить сравнение, классификацию по заданным критериям;

владеть общим приемом решения задач;

ориентироваться на разнообразие способов решения задач;

Обучающийся получит возможность научиться:

работать с дополнительными текстами и заданиями;

моделировать задачи на основе анализа жизненных сюжетов; формулировать выводы на основе аналогии, сравнения, обобщения;

пользоваться эвристическими приемами для нахождения решения алгоритмических задач.

Коммуникативные:

Обучающийся научится:

самостоятельно организовывать учебное взаимодействие в группе (определять общие цели, договариваться друг с другом и т.д.);

отстаивая свою точку зрения, приводить аргументы, подтверждая их фактами;

в дискуссии уметь выдвинуть контраргументы;

учиться критически относиться к своему мнению, с достоинством признавать ошибочность своего мнения (если оно таково) и корректировать его.

Обучающийся получит возможность научиться:

учитывать разные мнения и стремиться к координации различных позиций в сотрудничестве;

контролировать действия партнера;

договариваться и приходить к общему решению в совместной деятельности, в том числе в ситуации столкновения интересов.

проявлять инициативу в учебно-познавательной деятельности;

Обучающийся научится:

Предметные результаты:

навыкам алгоритмического мышления и пониманию необходимости формального описания алгоритмов;

понятию сложности алгоритма, знанию основных алгоритмов;

владеть стандартными приёмами написания на языке программирования программы для решения стандартной задачи с использованием основных конструкций программирования и отладки программ;

умению понимать программы, написанные на выбранном для изучения языке

Обучающийся получит возможность научиться:

новым знаниям в рамках учебного предмета, его преобразованию и применению в учебных, учебно-проектных и социально-проектных ситуациях, научным представлениям о ключевых теориях, типах и видах отношений, владению научной терминологией, ключевыми понятиями, методами и приемами.

2. Содержание программы и тематическое планирование занятий

№ п/п	Название тем	Кол-во часов
1	Техника безопасности.	1
2	Ознакомление с олимпиадными заданиями по информатике.	2
3	Решение простых задач и тренировка в использовании разных стратегий.	2
4	Основные математические концепции для олимпиадных задач.	2
5	Разбор примеров из предыдущих олимпиад для понимания подходов к решению задач.	4
6	Развитие навыков решения задач на математическую логику и доказательства.	2
7	Изучение принципов логического мышления и техник доказательства.	1
8	Решение сложных логических задач и анализ их решений.	2
9	Подготовка к заданиям на чтение и анализ текстов на олимпиадах.	2
10	Тренировка в скорочтении и поиске ключевой информации в тексте.	2
11	Решение задач, основанных на прочитанных текстах.	2
12	Разбор примеров задач на предмет интеграции различных алгоритмических концепций.	4
13	Тренировка в решении сложных задач, требующих применения нескольких методов одновременно.	4
14	Анализ ошибок и обратная связь для улучшения навыков	2

	участников.	
15	Подготовка к командным олимпиадам.	4
16	Формирование команд и развитие командного взаимодействия.	2
17	Решение командных задач и тренировка в соревновательной обстановке.	4
18	Проведение тестов, симулирующих реальные олимпиады.	4
19	Анализ результатов тестирования и обратная связь для каждого участника.	2
20	Индивидуальное консультирование и планирование индивидуальной работы.	4
21	Подготовка к финальным испытаниям и олимпиадам.	4
22	Прохождение полноценных олимпиадных испытаний в классе.	4
23	Проведение тренировочных олимпиад для подготовки к школьному этапу ВСОШ	10
24	Проведение тренировочных олимпиад для подготовки к муниципального этапа ВСОШ	10
25	Проведение тренировочных олимпиад для подготовки к региональному этапу ВСОШ	10
26	Проведение тренировочных олимпиад для подготовки к заключительному этапу ВСОШ	10
27	Обсуждение стратегий решения, анализ результатов и награждение участников.	2
	Итого:	102

Структура программы

Программа состоит из 13 теоретических разделов и 3 проектных. В конце каждого года обучения предполагается что обучающиеся выполняют небольшой проект. Данный проект будет показывать степень облученности учащегося и позволить сформировать дифференцируемый подход к его дальнейшему обучению

Разделы:

1. Знакомство с C++. Введение.

Рассматривается процесс написания простых программ на языке C++. Учащиеся узнают о роли функции `main ()` ознакомятся с базовой структурой программы и основными принципами синтаксиса. С некоторыми разновидностями операторов, используемых в программах на C++. Для операций ввода-вывода в программах вы будете использовать predefined объекты `cout` и `cin`, а еще вы научитесь создавать и использовать переменные. В конце главы вы познакомитесь с функциями — программными модулями языка C++.

2. Типы данных

Изучения принципа хранения двух разновидностей данных — целых чисел (чисел без дробной части) и чисел с плавающей точкой (числа с дробной частью) — в языке программирования C++ предусмотрены встроенные типы. Чтобы удовлетворить разнообразные требования программистов, язык C++ предлагает по несколько типов в каждой категории данных. Об этих типах, а также о создании переменных и написании констант различных типов, речь пойдет в разделе 3. Вы узнаете также о том, как в языке C++ осуществляется явное и неявное преобразование одного типа в другой.

3. Составные типы данных

На основе базовых встроенных типов данных в языке программирования C++ можно создавать более совершенные типы. В этом разделе будут рассмотрены другие формы, включая массивы, содержащие несколько однотипных значений; структуры, хранящие несколько разнотипных значений; указатели, которые идентифицируют ячейки памяти. Учащиеся узнают о том, как создаются и хранятся текстовые строки и как выполняется обработка операций текстового ввода-вывода за счет использования символьных массивов, присущих языку C, и класса `string` языка C++.

4. Указатели

В этом разделе учащиеся узнают о том, как в языке C++ осуществляется распределение памяти, включая операции `new` и `delete` явного управления памятью.

5. Проектная деятельность

В качестве подведения итога пройденного материала за год учащимся на выбор будут предложены проекты, а также индивидуальное задание в соответствии с пройденным материалом. После выполнения всех работ, а также полный их разбор первый год обучения считается окончанным.

6. Циклы

Часто бывает так, что в программе необходимо выполнить некоторые повторяющиеся действия, и для этих целей в языке C++ предусмотрены три циклических структуры: цикл `for`, цикл `while` и цикл `do while`. Завершение цикла осуществляется по заранее определенному условию, поэтому для создания циклических структур и их управления в языке C++ используются операции отношения. В этом разделе учащиеся узнают, как создавать циклы, в которых входные данные считываются и обрабатываются посимвольно. В конце раздела учащиеся овладеют навыком создавать двумерные массивы и использовать вложенные циклы для их обработки.

7. Ветвление и логика

Поведение программы будет “интеллектуальным”, если она сможет адаптироваться к различным ситуациям. Раздел повествует об управлении ходом выполнения программ с помощью операторов `if`, `if else` и `switch` и условные операции. Учащиеся узнают о том, как с помощью условных операций можно выразить проверку для принятия решения. В конце раздела учащиеся ознакомятся с обзором процессов файлового ввода-вывода.

8. Создание программных модулей C++

Функции являются основными стандартными программными блоками в языке C++. Главное внимание в этом разделе уделено возможностям, характерным как для функций языка C++, так и для функций языка C. В частности, здесь будет представлен общий формат описания функций и рассказано о том, как с помощью прототипов функций можно повысить надежность создаваемых программ. Учащиеся научатся создавать функции для обработки массивов данных, символьных строк и структур. Будет рассмотрена рекурсия, возникающая при вызове функцией самой себя; учащиеся узнают о том, как с помощью функции можно реализовать стратегию “разделяй и властвуй”. В конце раздела познакомятся с указателями на функции, благодаря которым одна функция может с помощью аргумента вызывать другую функцию.

9. Проектная деятельность

В качестве подведения итога пройденного материала за год учащимся на выбор будут предложены проекты, а также индивидуальное задание в соответствии с пройденным материалом. После выполнения всех работ, а также полный их разбор первый год обучения считается оконченным.

10. Дополнительные сведения о функциях

В разделе будет рассказано о новых возможностях функций в C++. Мы рассмотрим подставляемые функции, с помощью которых можно ускорить выполнение программ за счет увеличения их размера. Учащиеся будут работать со ссылочными переменными, которые предлагают альтернативный способ передачи информации в функции. Благодаря аргументам, используемым по умолчанию, функции могут задавать значения автоматически тем аргументам функций, которые не указываются при вызове. Перегрузка функций позволяет создавать функции с одинаковыми именами, но принимающие разные списки аргументов. Все эти особенности часто применяются при создании классов. Учащиеся познакомятся с шаблонами функций, благодаря которым можно создавать целые семейства связанных функций.

11. Модель памяти и пространство имен

В разделе рассмотрены вопросы объединения программ, состоящих из множества файлов. В ней будут представлены варианты распределения памяти, рассмотрены способы управления памятью; учащиеся узнают, что такое область видимости, компоновка и пространства имен, определяющие, в каких частях программы будут известны те или иные переменные.

12. Объекты и классы

Класс представляет собой тип, определенный пользователем, а объект (например, переменная) является экземпляром класса. В данном разделе учащиеся узнают о том, что такое объектно-ориентированное программирование, и как создаются классы. В объявлении класса указывается информация, хранящаяся в объекте класса, и операции (методы класса), разрешенные над объектами класса. Некоторые части объекта будут видимыми для внешнего мира (общедоступная часть), а другие будут скрытыми (приватная часть). В момент создания и уничтожения объектов инициируются специальные методы класса, называемые конструкторами и деструкторами. Учащиеся узнают об этих и других особенностях классов и получают представление о вариантах использования классов для реализации абстрактных типов данных, таких как стек.

13. Использование классов

В разделе продолжается знакомство с классами. Прежде всего, будет рассмотрен механизм перегрузки операций, посредством которой программист определяет, как операция (например, +) будет работать с объектами класса. Мы поговорим о дружественных функциях, позволяющих обращаться к данным класса, которые в общем случае являются недоступными. Учащиеся узнают о том, как некоторые конструкторы и функции-члены перегруженных операций могут быть использованы для управления преобразованием одного типа класса в другой.

14. Классы и динамическое использование памяти

Зачастую бывает полезным применение указателя на член класса для динамического распределения памяти. Если в конструкторе класса вы будете использовать операцию new для динамического распределения памяти, то на этот случай вы должны предусмотреть соответствующий деструктор, явный конструктор копирования и явную операцию присваивания. В разделе мы рассмотрим, как это можно сделать, и расскажем о поведении функций-членов, которые генерируются неявным образом, если явные описания не предусмотрены. Опыт работы с классами учащиеся закрепят на примерах использования указателей на объекты и моделировании очереди.

15. Наследование классов

Одной из самых замечательных особенностей объектно-ориентированного программирования является наследование, благодаря которому производный класс наследует особенности базового класса, что открывает возможности для повторного использования кода базового класса. В разделе обсуждается общедоступное наследование, моделирующее отношения is-a, при которых производный объект рассматривается как частный случай базового объекта. Здесь в качестве примера можно упомянуть физика, который является частным случаем ученого. Некоторые отношения при наследовании являются полиморфными, что означает возможность писать код, используя несколько связанных между собой классов, для которых метод с одним и тем же именем может реализовывать поведение, зависящее от типа объекта. Чтобы такое поведение было возможным, необходима особая функция-член, называемая виртуальной функцией. Нередко для отношений наследования лучше всего использовать абстрактные базовые классы. В этой главе будут рассмотрены все эти вопросы с описанием ситуаций, в которых общедоступное наследование будет иметь смысл.

16. Создание программы в ООП

В качестве подведения итога пройденного материала за все время обучения учащимся на выбор будут предложены проекты, а также индивидуальное задание в соответствии с пройденным материалом. В результате у учащегося который полностью прошел данный курс, на выходе должно полноценное приложение позволяющие демонстрировать его работу над проектом. Данный проект и будет говорить о качестве обученности за все годы обучения.

3. Формы контроля и оценочные материалы

Итоговое оценивание будет проводиться с учетом качества практических работ.

Формы предъявления и демонстрации образовательных результатов открытое занятие, итоговое занятие, создание программы.

4. Организационно-педагогические условия реализации программы.

Особенности работы по программе

Основными формами организации учебно-познавательной деятельности, учащихся являются:

- изложение узловых вопросов курса (лекционный метод),
- собеседования (дискуссии),
- тематическое комбинированное занятие,
- решение нестандартных задач;
- участие в олимпиадах,
- знакомство с литературой по программированию
- самостоятельная работа,
- работа в парах, в группах,
- творческие практические работы.

Методы обучения:

- словесный (урок рассуждение)
- частично поисковый, исследовательский,
- объяснительно-иллюстративный.

Меж предметные связи: темы тесно связаны с уроками экономики, математики, предметами естественного цикла.

ВВ программе базовыми являются следующие технологии, основанные на:

- уровневой дифференциации обучения,
- реализации деятельностного подхода,
- реализации проектной деятельности

5. Учебно - методическое и материально – техническое обеспечение образовательного процесса

Материально – техническое обеспечение

- ✓ оборудование для демонстрации мультимедийных программ и презентаций (компьютер, мультимедийный проектор).
- ✓ ПК ученика
- ✓ ПК учителя
- ✓ Локальная сеть

Программное обеспечение:

Dev C++
Notepad++
MS Office 2013
C++ Builder 2019

Электронные ресурсы

Форматы электронной поддержки и сопровождения программы включают в себя ресурсы для совместной или самостоятельной работы учащихся, позволяющие осуществлять автоматизированную проверку решения задач по программированию.

Дистанционный портал обучения Sirius.online	Sirius.online
Сайт дистанционной подготовки по информатике	https://informatics.msk.ru/
Портал соревнований по программированию	http://codeforces.com/
Школа программиста	https://acmp.ru/

При проведении занятий используются компьютеры с установленным бесплатным ПО Dev C++ 5.11 (TDM-GCC 4.9.2), а так же используются электронные материалы, разработанные автором данной программы.

Формы проведения занятий: беседы, практические занятия, самостоятельная работа, проектная деятельность.

Использование метода проектов позволяет обеспечить условия для развития у обучающихся навыков самостоятельной постановки задачи, выбора оптимального варианта их решения, самостоятельного достижения цели, анализа полученных результатов, с точки зрения решения поставленной задачи.

Программой предусмотрены методы обучения: объяснительно-иллюстративные, частично-поисковые (вариативные задания), творческие, практические.

Список использованных источников

1. Стивен Прата, Язык программирования С++ Лекции и упражнения 5-е изд.: Пер. с англ. – М.: ООО “И. Д. Вильямс”, 2007. – 1184 с.
2. Меньшиков Ф. В. Олимпиадные задачи по программированию (+CD). –СПб.: Питер, 2006 – 315 с.: ил
3. Учебник «Информатика» для 7 класс: в 2 ч. Ч. 1 /К. Ю. Поляков, Е.А. Еремин. – М.: БИНОМ. Лаборатория знаний
4. Учебник «Информатика» для 7 класс: в 2 ч. Ч. 2 /К. Ю. Поляков, Е.А. Еремин. – М.: БИНОМ. Лаборатория знаний
5. Учебник «Информатика» для 8 класс /К. Ю. Поляков, Е.А. Еремин. – М.: БИНОМ. Лаборатория знаний
6. Учебник «Информатика» для 9 класс /К. Ю. Поляков, Е.А. Еремин. – М.: БИНОМ. Лаборатория знаний