**Муниципальное бюджетное общеобразовательное учреждение «Северский лицей»**

**Программа дополнительного образования**

***«Юный программист»***

**(1 час х 34 нед. = 34 часа)**

|  |
| --- |
| **Составитель:**  учитель информатики  Иванова Наталья Анатольевна |

**2023-2024 учебный год**

**Пояснительная записка**

В современном образовательном процессе проблема формирования алгоритмического мышления обучающихся достаточно актуальна. Роль обучения информатике в развитии мышления обусловлена современными разработками в области методики моделирования и проектирования, особенно в объектно-ориентированном моделировании. Умение для любой предметной области выделить систему понятий, представить их в виде совокупности атрибутов и действий, описать алгоритмы действий и схемы логического вывода (т.е. то, что и происходит при информационно-логическом моделировании) улучшает ориентацию человека в этой предметной области и свидетельствует о его развитом мышлении. В программе для основной школы составленной на основе ФГОС подчеркивается необходимость в курсе информатики уделить большее внимание вопросам алгоритмизации и программирования. При этом учитывается важная роль, которую играет алгоритмическое мышление в формировании личности.

Под алгоритмическим стилем мышления подразумевается  система мыслительных действий и приемов, направленных на решение как теоретических, так и практических задач, результатом которых являются  алгоритмы как специфические продукты человеческой деятельности. Огромные возможности для развития алгоритмического стиля открываются при изучении темы алгоритмизация и программирование. Программирование помогает школьникам пройти все основные этапы формализованного решения некоторой творческой точно сформулированной задачи.

Технология решения задач на компьютере – это не только составление программы и получение загрузочного модуля, а формирование модели, составление алгоритма, отладка программы и ее тестирование. Без программирования развитие алгоритмического стиля мышления практически невозможно, так как отсутствует возможность компьютерного эксперимента проверки работоспособности алгоритма. Поэтому изучать основы алгоритмизации и программирования нужно в средней школе на базе систем программирования.

На всех этапах всероссийской олимпиады школьников большое внимание уделяется задачам, связанным с программированием. Данный авторский курс направлен на более глубокое изучение обучающимися алгоритмики и знакомство с различными средами объектно-ориентированного программирования. Курс носит практическую направленность, т.к. это необходимо для того, чтобы овладеть навыками составления алгоритмов и программ.

**Цель курса** формирование и развитие у обучающихся алгоритмического мышления.

**Задачи курса**

* создать условия для формирования и развития у обучающихся интереса к изучению информатики и информационных технологий;
* развитие алгоритмического мышления обучающихся;
* расширить спектр посильных учащимся задач из различных областей знаний, решаемых с помощью формального исполнителя.

Предполагаемый объем учебного времени — 1 часа в неделю в течение учебного года. Таким образом, весь курс рассчитан на 34 часа.

Курс поддержан программным обеспечением **КуМир** (Комплект Учебных МИРов) и **Scratch.**

**КуМир** - система программирования, предназначенная для поддержки начальных курсов информатики и программирования в средней и высшей школе.

**Особенности системы КуМир:**

* В системе КуМир используется школьный алгоритмический язык с русской лексикой и встроенными исполнителями Робот и Чертёжник.
* При вводе программы КуМир осуществляет постоянный полный контроль ее правильности, сообщая на полях программы об всех обнаруженных ошибках.
* При выполнении программы в пошаговом режиме КуМир выводит на поля результаты операций присваивания и значения логических выражений. Это позволяет ускорить процесс освоения азов программирования.
* Кумир работает в операционных системах Windows или Linux.
* Система Кумир разработана в НИИСИ РАН по заказу Российской Академии Наук и распространяется свободно на условиях лицензии GNU 2.0.
* Данная лицензия разрешает бессрочно использовать КуМир на любом количестве компьютеров в любых целях без оформления каких либо дополнительных документов.

**Scratch -** это визуальная объектно-ориентированная среда программирования для обучения школьников младших и средних классов.

**Особенности системы Scratch:**

* Программы на Scratch состоят из графических блоков, подписи к которым зависят от выбранного для интерфейса языка, в том числе русского.
* Основными компонентами Scratch -программы являются объекты-спрайты.
* Для программирования сценариев в Scratch используется drag-and-drop-подход: блоки из Окна блоков перетаскиваются в область скриптов.
* Язык Scratch оперирует числами, текстовыми строками, логическими значениями, а также списками, играющими роль динамических массивов.
* Scratch разрабатывается небольшой командой исследователей из «Lifelong Kindergarten Group» в Массачусетском технологическом институте и распространяется свободно на условиях  простой [копилефтн](http://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BF%D0%B8%D0%BB%D0%B5%D1%84%D1%82)ой лицензии, с дополнительными ограничениями по неиспользованию оригинальных [торговых марок](http://ru.wikipedia.org/wiki/%D0%A2%D0%BE%D1%80%D0%B3%D0%BE%D0%B2%D0%B0%D1%8F_%D0%BC%D0%B0%D1%80%D0%BA%D0%B0). Код может быть загружен с официального сайта.

**Ожидаемые результаты**.

После изучения курса учащиеся должны знать:

* что такое исполнитель,
* что такое среда исполнителя,
* конструкции, команды исполнителя,
* состояние исполнителя,
* что такое алгоритм,
* типы алгоритмов (линейный, разветвляющийся, циклический),
* логические операции,
* эффективность и сложность алгоритма,
* координаты на плоскости,
* преобразование программ,
* параллельное программирование.

И уметь:

* составлять алгоритмы любого типа,
* оформлять алгоритмы в изучаемых средах объектно-ориентированного программирования,
* составлять программы на языке Pascal.

**Методы и средства обучения**

При изложении курса рекомендуется применять мини-лекции и компьютерный практикум. На мини-лекциях: учитель выдает необходимый минимум теоретического материала (назначение и форматы операторов, особенности применяемых алгоритмов, способы их построения), снабженного демонстрационными примерами. Далее учащиеся получают задания (некоторые из которых, по возможности, должны были выполняться ими ранее в изучаемых средах программирования) и приступают к их выполнению на компьютерном практикуме. Подобный подход реализует один из важных методологических принципов — параллельное изложение со сравнением, что позволяет ученикам глубже постигать суть выполняемых операций.

**Особенности организации занятий.**

Учебный процесс носит интерактивный характер: обучающиеся участвуют в практических занятиях, выполняют индивидуальные задания.

Лекции используются для ознакомления детей с новым теоретическим материалом.

Интерактивные практические занятияпозволяют приобрести ученикам практические умения и навыки по каждой теме.

Выполнение заданий и упражнений индивидуально и по парам используется для расширения возможностей по применению на практике усвоенной теории. Занятия проводятся также с целью синтезировать полученные знания и умения по теме и предоставляют возможность преподавателю оценить эффективность изучения каждой темы.

**Требования к технической оснащенности занятий**

Для проведения занятий требуется компьютерный класс (каждый ученик должен быть обеспечен компьютером).

**Содержание курса**

**Исполнитель. Алгоритмы (10ч)**

Среда исполнителя. Конструкции и команды исполнителя. Алгоритм и его формальное исполнение. Свойства алгоритмов. Основные типы алгоритмических структур. Основы объектно-ориентированного визуального программирования.

**Среда КуМир Чертежник (10 ч)**

Исполнитель Чертежник в системе “КуМир”. Список команд исполнителя

Практические работы:

1. Команда повтора с условием

2. Команда повтора n-раз

3. Команда повтора с параметром

4. Решение задач

**Среда КуМир Робот (10 ч)**

Исполнитель Роботв системе “КуМир”. Список команд исполнителя

Практические работы:

5. Команда повтора с условием

6. Команда повтора n-раз

7. Вложенные циклы

8. Решение задач

**Среда объектно-ориентированного программирования Scratch (20ч)**

Интерфейс Scratch и основы работы в нем. Скрипты, костюмы, звуки, движения.

Практические работы:

9. «Управление несколькими объектами»

10. «Последовательное и одновременное выполнение»

11. «Интерактивность, условия, переменные»

12. «Случайные числа»

13. «Рисование в Scratch»

14. «Диалог с программой»

15. «Создание объектов и костюмов»

16. «Использование библиотеки объектов»

17. «Смена фона»

**Среда объектно-ориентированного программирования Pascal (18ч)**

Этапы разработки программы. Характеристики программы. Данные. Результаты. Типы данных. Алфавит языка. Структура программы. Операторы.

Практические работы:

18. Операторы (ввод данных, вывод данных, присваивание).

19. Условный оператор и оператор выбора.

20. Составление программ с условным оператором

21. Операторы повтора (циклы)

22. Составление программ, использующих цикл REPEAT

23. Составление программ, использующих цикл WHILE

24. Составление программ, использующих цикл FOR

25. Зачетная работа

**Тематический план**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Тема** | **Количество часов** | **теория** | **практика** |
| 1 | Исполнители | 2 | 2 |  |
| 2 | Алгоритмы | 3 | 3 |  |
| 3 | Среда КуМир - Чертежник | 5 | 1 | 4 |
| 4 | Среда КуМир - Робот | 5 | 1 | 4 |
| 5 | Среда объектно-ориентированного программирования Scratch | 10 | 1 | 9 |
| 6 | Среда объектно-ориентированного программирования Pascal | 9 | 2 | 7 |
|  | Итого | 34 | 10 | 24 |

**Примерное тематическое планирование курса   
1 часа в неделю, всего 34 часа**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Тема** | **Дата** | **Примечание** |
| **Исполнители. Алгоритмы (5ч)** | | | |
| 1-2 | Исполнители. Среда исполнителя |  |  |
| 3-4 | Список команд исполнителя. Конструкции исполнителя |  |  |
| 5-6 | Алгоритм. Свойства алгоритмов |  |  |
| 7-8 | Типы алгоритмических структур |  |  |
| 9-10 | Основы объектно-ориентированного визуального программирования |  |  |
| **Среда КуМир Чертежник (5 ч)** | | | |
| 11-12 | Среда исполнителя Чертежник. Список команд исполнителя |  |  |
| 13-14 | Практическая работа № 1. Команда повтора с условием |  |  |
| 15-16 | Практическая работа № 2. Команда повтора n-раз |  |  |
| 17-18 | Практическая работа № 3. Команда повтора с параметром |  |  |
| 19-20 | Практическая работа № 4. Решение задач |  |  |
| **Среда КуМир Робот (5 ч)** | | | |
| 21-22 | Среда исполнителя Робот. Список команд исполнителя |  |  |
| 23-24 | Практическая работа № 5. Команда повтора с условием |  |  |
| 25-26 | Практическая работа № 6. Команда повтора n-раз |  |  |
| 27-28 | Практическая работа № 7. Вложенные циклы |  |  |
| 29-30 | Практическая работа № 8. Решение задач |  |  |
| **Среда объектно-ориентированного программирования Scratch (10ч)** | | | |
| 31-32 | Знакомство со Scratch |  |  |
| 33-34 | Практическая работа № 9 «Управление несколькими объектами» |  |  |
| 35-36 | Практическая работа № 10 «Последовательное и одновременное выполнение» |  |  |
| 37-38 | Практическая работа № 11 «Интерактивность, условия, переменные» |  |  |
| 39-40 | Практическая работа № 12 «Случайные числа» |  |  |
| 41-42 | Практическая работа № 13 «Рисование в Scratch» |  |  |
| 43-44 | Практическая работа № 14 «Диалог с программой» |  |  |
| 45-46 | Практическая работа № 15 «Создание объектов и костюмов» |  |  |
| 47-48 | Практическая работа № 16 «Использование библиотеки объектов» |  |  |
| 49-50 | Практическая работа № 17 «Смена фона» |  |  |
| **Среда объектно-ориентированного программирования Pascal (9 ч)** | | | |
| 51-52 | Этапы разработки программы. Характеристики программы. Данные. Результаты. |  |  |
| 53-54 | Типы данных. Алфавит языка. Структура программы. |  |  |
| 55-56 | Практическая работа № 18. Операторы (ввод данных, вывод данных, присваивание) |  |  |
| 57-58 | Практическая работа № 19. Условный оператор и оператор выбора. |  |  |
| 59-60 | Практическая работа № 20. Составление программ с условным оператором |  |  |
| 61-62 | Практическая работа № 21. Операторы повтора (циклы) |  |  |
| 63-64 | Практическая работа № 22. Составление программ, использующих цикл REPEAT |  |  |
| 65-66 | Практическая работа № 23. Составление программ, использующих цикл WHILE |  |  |
| 67 | Практическая работа № 24. Составление программ, использующих цикл FOR |  |  |
| 68 | Практическая работа № 25. Итоговое занятие |  |  |

**Список литературы**

1. Житкова О.А., Кудрявцева Е.К. Алгоритмы и основы программирования. М.: «Интеллект-центр», 2001.
2. Житкова О.А., Кудрявцева Е.К. Справочные материалы по программированию на языке Pascal. М.: «Интеллект-центр», 2001.
3. Немнюгин С.А. Turbo Pascal учебник. Издательство «Питер», 2001.
4. Рапаков Г.Г. Ржецкая С.Ю. Turbo Pascal для студентов и школьников. – СПб.: БХВ-Петербург, 2011
5. <http://ido.tsu.ru/other_res/school2/osn/metod/prog/p1.html>
6. <http://scratch.mit.edu/>
7. <http://younglinux.info/scratch>
8. <http://zarapina.blogspot.ru/>
9. <http://ips.ifmo.ru/courses/pascal/>