

**Муниципальное бюджетное общеобразовательное учреждение
Землянозаймская средняя школа
Чановского муниципального округа Новосибирской области**

СОГЛАСОВАНО

Зам. директора по УВР



Мустафина Р.М.

«17» 08. 2026 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОГО КУРСА

ОСНОВЫ IT

Составитель(и):

Федоров А.В., учитель физики

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Курс «Программирование на языке Python» предназначен для учащихся 8-11 классов, желающих изучить программирование.

Язык программирования Python – широко известный скриптовый язык высокого уровня, известный своим минимализмом и простотой. Это мощный инструмент для создания программ самого разнообразного назначения и решения задач различных типов.

В процессе обучения учащиеся получают навыки объектно-ориентированного программирования, умения решать задания олимпиадного уровня с повышенной сложностью.

В завершении обучения учащиеся смогут реализовать собственный проект в области разработки пользовательского интерфейса на базе Qt/Python или веб-программирования на базе Python/Django.

Актуальность программы «Программирование на языке Python» вызвана потребностью современного информационного общества в высокообразованных, адаптированных к изменениям специалистах в IT-сфере. Для удовлетворения данной потребности перед дополнительным образованием стоит задача развития человеческого потенциала через выявление талантливых детей, развитие их мотивации и способностей.

Дополнительная общеобразовательная программа «Программирование на языке Python» является программой технической направленности и продвинутого образовательного уровня.

Python – это современный, универсальный, интерпретируемый, мультипарадигмальный язык программирования, который позволяет совмещать процедурный подход к написанию кода с объектно-ориентированным и функциональным.

Язык программирования Python имеет много преимуществ перед другими языками программирования:

- открытая разработка;
- довольно прост в изучении, особенно на начальном этапе;
- особенности синтаксиса стимулируют программиста писать хорошо читаемый код;
- имеет большое сообщество, позитивно настроенное по отношению к новичкам;
- множество полезных библиотек и расширений языка можно легко использовать в своих проектах благодаря предельно унифицированному механизму импорта и программным интерфейсам;
- кроссплатформенность;
- механизмы модульности хорошо продуманы и могут быть легко использованы;
- бесплатные простые и удобные среды для разработки программ;
- подходит для любых решений в области программирования, будь то офисные программы, web-приложения, GUI-приложения и т.д.

Также, важно, что большинство Интернет-ресурсов для проведения олимпиад по информатике и программированию поддерживают язык Python, и что с 2015 года он прочно стал одним из пяти языков ЕГЭ по информатике.

Традиционное изучение в общеобразовательных учреждениях языка программирования Pascal не может в полной мере удовлетворить потребности школьников старших классов информационно-технологической и физико-математической направленностей в изучении новых и популярных языков программирования. Изучение языка программирования Python поможет школьнику получить более целостное представление о профессии программиста, разработчика, инженера, повысить уровень подготовки к обучению в ВУЗе на технических специальностях.

Программа предполагает участие обучающихся в интеллектуальных соревнованиях по программированию различных уровней, создание образовательных практических или научно-исследовательских проектов, что даст возможность детям полностью реализовать свой интеллектуальный и творческий потенциал.

Цель- формирование и развитие комплекса компетенций в области алгоритмизации и программирования на языке Python.

Задачи

Обучать:

базовой части математического аппарата, применяемого в современном программировании; работе в интегрированных средах разработки и в он-лайн сервисах Интернета, связанных с программированием;

навыкам алгоритмического и логического мышления, грамотной разработки программ;

методам программирования на языках, применяемых в современной вычислительной технике;

работе в интегрированных средах разработки;

навыкам разработки эффективных алгоритмов и программ на основе языка программирования Python.

возможностям получения новых знаний в области компьютерного программирования.

Развивать:

тип мышления, направленный на выбор оптимальных решений;

навыки инженерного мышления и умения работать по предложенным инструкциям;

навыки программирования, проектирования и эффективного использования электронного вычислительного оборудования;

внимательность, аккуратность и изобретательность при выполнении учебных проектов;

творческое мышление и воображение, умение излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений и информационного поиска.

Воспитывать:

правильный методологический подход к познавательной и практической деятельности;

мотивацию учащихся к изобретательству, созданию собственных программных реализаций;

стремление к овладению техникой исследования;

трудолюбие, инициативность и настойчивость в преодолении трудностей;

стремление к получению качественного законченного результата в проектной деятельности;

навыки сознательного и рационального использования компьютера в своей учебной, а впоследствии и в профессиональной деятельности;

навыки проектного мышления, работы в команде, эффективно распределять обязанности при реализации проекта, требующего от участников знаний и умений из различных направлений.

Возраст детей, участвующих в освоении данной образовательной программы: программа ориентирована на учащихся 14-18 лет (учеников 8-11х классов), желающих освоить язык программирования Python.

Сроки реализации программы: образовательный процесс длится в течение учебного года, объем 34 часов.

Срок реализации программы – 1 год.

Количество обучающихся в группе: 15 человек.

Оптимальный режим занятий – 1 раз в неделю по 45 минут с 5-минутным перерывом. Во время выполнения практической работы за компьютером предусмотрено проведение физкультминутки (приложение 1).

Формы организации деятельности на занятии: программа предусматривает использование фронтальной, индивидуальной и групповой формы учебной работы учащихся:

- фронтальная форма - для изучения нового материала, информация подаётся всей группе из 10-15 человек;

- индивидуальная форма - самостоятельная работа учащихся, педагог может направлять процесс в нужную сторону;

- групповая форма помогает педагогу сплотить группу общим делом, способствует качественному выполнению задания, для реализации проектной деятельности в малых группах (3-4 человека).

• Помимо основных занятий, программа включает в себя и культурно-массовые мероприятия, такие как: экскурсии, конкурсы, выставки.

Программой предусмотрены следующие **формы занятий**: лекционно-практические занятия, проблемные лекции, тренинги, мастер-классы.

Планируемые результаты освоения программы

Предметные компетенции:

закрепить знания синтаксиса, основных алгоритмических конструкций и парадигм программирования языка программирования Python;

изучить и реализовать основные подходы к созданию программ на высоко-уровневом языке программирования Python;

научить распознавать необходимость применения той или иной алгоритмической конструкции при решении задач;

научить организовывать данные для эффективной алгоритмической обработки;

совершенствовать навыки решения олимпиадных задачи реализовать их в олимпиадах по программированию различного уровня;

уметь разрабатывать программы и приложения в стиле ООП на языке программирования Python.

реализовать полученные знания с помощью проектной работы, на-правленной на решение прикладной задачи в интересующей учащегося научной области.

Метапредметные результаты:

сформировать представления о роли компьютерного программирования в развитии общества, изменении содержания и характера деятельности чело-века.

научить применять полученные знания в области прикладной математики и информатики;

дать опыт дистанционного обучения с помощью современных онлайн-сервисов Интернета;

научить проводить научное исследование и получить новый научный или прикладной результат. под руководством преподавателя

Личностные результаты:

развить абстрактное, аналитическое, алгоритмическое и логическое мышление, гибкость ума;

сформировать готовность к саморазвитию, самореализации.

научиться работать в группе, быть требовательным к себе, ответственным и добросовестным;

научиться выстраивать индивидуальную образовательную траекторию для непрерывного самостоятельного повышения уровня знаний в области IT;

совершить осознанный выбор высшего образовательного учреждения и специальности в информационно-технологической или инженерно- технической областях для продолжения обучения;

развить способности к абстрактному мышлению, анализу и синтезу;

сформировать способность самостоятельного приобретения новых знаний, умений и навыков с помощью IT технологий и использования их в практической деятельности;

расширить и углубить личное мировоззрение.

СОДЕРЖАНИЕ ПРОГРАММЫ
УЧЕБНЫЙ (ТЕМАТИЧЕСКИЙ) ПЛАН

№ п\п	Наименование раздела	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
1	Знакомство с языком программирования Python	2	1	1	учет решенных задач
2	Подпрограммы	2	1	1	учет решенных задач
3	Множества и словари	2	1	1	учет решенных задач
4	Рекурсия	2	1	1	учет решенных задач
5	Работа с файлами	2	1	1	учет решенных задач
6	Исключения и их обработка	2	1	1	учет решенных задач
7	Принципы объектно-ориентированного программирования	2	1	1	учет решенных задач
8	Алгоритмы сортировки и учет поиска	2	1	1	учет решенных задач
9	Длинная арифметика	2	1	1	учет решенных задач
10	Динамическое программирование	2	1	1	учет решенных задач
11	Структуры данных	6	4	2	учет решенных задач
12	Комбинаторика	4	2	2	учет решенных задач
13	Подготовка и защита проекта	2		4	Защита проекта
	Всего по программе за год:	34	16	18	

Тема 1. Знакомство с языком программирования Python – 2 часа.

Основные понятия: Условный оператор, цикл, строка, список, кортеж, двумерный массив.

Лекции: условный оператор, условное и альтернативное исполнение алгоритма, оператор цикла, ввод-вывод строки, форматирование строки, методы работы со строкой представление списка и кортежа в памяти компьютера, сходства и различия, способы заполнения списка (с клавиатуры, из файла, случайным образом, по формуле), методы работы со списком и кортежем. описание массива, размерность массива, ввод и вывод значений массива, вложенные генераторы двумерных массивов, действия над элементами массива.

Практика: решение задач по теме.

Тема 2. Подпрограммы – 2 часа

Основные понятия: функция, аргумент функции, lambda-функция, локальные и глобальные переменные

Лекции: функция: определение, вызов, аргументы, передача аргументов, возвращаемые значения, области видимости имен, встроенные функции lambda-функции, декораторы.

Практика: решение задач по теме.

Тема 3. Множества и словари – 2 часа

Основные понятия: словарь, ассоциативный массив, ключ, значение, множество, кодирование.

Лекции: словарь, преимущества и недостатки, методы работы со словарем, словари со смешанными значениями, кодирование и декодирование текста, множество, преимущества и недостатки, методы работы с множеством.

Практика: решение задач по теме.

Тема 4. Рекурсия – 2 часа

Основные понятия: рекурсия, глубина рекурсии, прямая и косвенная рекурсия.

Лекции: понятие рекурсии, основные концепции, реализация в языке Python, преимущества и недостатки рекурсии.

Практика: решение задач по теме.

Тема 5. Работа с файлами – 2 часа

Основные понятия: файл, чтение, запись, открытие и закрытие файла.

Лекции: операции с файлами и директориями, работа с файловыми объектами: функция open, чтение и запись в файлы, особенности работы с файлами, обработка файлов в формате Unicode, сохранение базовых объектов языка Python в файл, текстовые и бинарные файлы.

Практика: решение задач по теме.

Тема 6. Исключения и их обработка – 2 часа

Основные понятия: исключение, системная ошибка, Exception.

Лекции: исключения, иерархия исключений, использование исключений: обработка и вызов, синтаксис, собственные исключения.

Практика: решение задач по теме.

Тема 7. Принципы объектно-ориентированного программирования – 2 часа

Основные понятия: класс, метод INIT, экземпляр, наследование, полиморфизм, виджет, интерфейс.

Лекции: основные понятия ООП и их реализация в Python, классы, экземпляры, методы и атрибуты, конструкторы, наследование, множественное наследование, композиция при разработке классов, перегрузка операций и функций.

Практика: создание собственного класса.

Тема 8. Алгоритмы сортировки и поиска – 2 часа

Основные понятия: трудоёмкость алгоритма, скорость роста функции, сложность алгоритма, оценка Q (тетта), оценка O (О большое), сортировка.

Лекции: асимптотический анализ алгоритмов, асимптотические обозначения в анализе функции трудоемкости алгоритма, определение сложности алгоритма, основные оценки в асимптотическом анализе алгоритмов, виды сортировки (сортировка выбором, сортировка вставками, сортировка «методом пузырька», сортировка Шелла, быстрая сортировка).

Практика: решение задач по теме.

Тема 9. Длинная арифметика – 2 часа

Основные понятия: длинное число, тип данных, длинная арифметика.

Лекции: встроенная поддержка длинной арифметики в Python, сложение, умножение, вычитание, деление, возведение в степень и другие операции с длинными числами.

Практика: решение задач по теме.

Тема 10. Динамическое программирование – 2 часа

Основные понятия: динамическое программирование, жадный алгоритм.

Лекции: Одномерное динамическое программирование, двумерное динамическое программирование, жадные алгоритмы.

Практика: решение задач по теме.

Тема 11. Структуры данных – 6 часов

Основные понятия: массив, список, очередь, стек, дек, граф, дерево, хеш-таблица.

Лекции: основные структуры данных, графы, виды графов и пути в графах, хранение графов в программах, ориентированный, граф неориентированный, вершина (узел), путь, простой путь, длина пути, компонентная связанность, виды и реализация деревьев, список ребер, матрица смежности, список смежности. алгоритмы на графах, обход в глубину и в ширину, хеширование,

Практика: решение задач по теме.

Тема 12. Комбинаторика – 4 часа

Основные понятия: комбинаторика, элементы комбинаторики, перестановка, факториал, сочетание, размещение.

Лекции: правило сложения, правило умножения, принцип Дирихле, сочетания и размещения, бином Ньютона.

Практика: решение задач по теме.

Тема 13. Подготовка и защита проекта – 4 часа

СРОКИ И ФОРМЫ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ И ИТОГОВОЙ АТТЕСТАЦИИ

Предметом контроля являются знания, умения и навыки учащихся, их творческие достижения, полученные в течение курса обучения, а также их внутренние личностные результаты (освоенные способы деятельности, знания, умения, готовность к саморазвитию и самоопределению), обозначенные целеполаганием курса.

Виды контроля, используемые в курсе:

- текущий (самостоятельные работы, текущие практические работы);
- промежуточный (выполнение индивидуальных практических заданий);
- итоговый (проектная работа).

Результатом освоения программы можно считать успешную разработку и защиту проекта, а также, в случае перехода в 11 класс школы, выбор обучающимся на следующий учебный год дополнительной общеобразовательной программы научно-технической направленности.

ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

Методическое обеспечение

1. Гуриков С.Р. Основы алгоритмизации и программирования на Python / С.Р. Гуриков. – М.: «Форум», 2018 – 343 с.
2. Доусен М. «Программируем на Python» (PythonProgrammingfortheAbsoluteBeginner) / М. Доусен. – СПб: «Пи-тер», 2016. – 416 с.
3. Дронов В.А. Django: Практика создания Web-сайтов на Python / В. Дронов, СПб: «БХВ-Петербург», 2016. – 528 с.
4. Лутц М. «Изучаем Python», 4 издание, – Пер. с англ. / М. Лутц – СПб: «Символ-Плюс», 2011. – 1280 с.
5. Любанович Б. «Простой Python. Современный стиль программирования» / Б. Любанович. – СПб: «Питер», 2016. – 480 с.
6. Прохоренок Н. «Python 3 и PyQt 5. Разработка приложений» / Н. Прохоренок, В. Дронов. СПб: «БХВ-Петербург», 2019. – 832 с.
7. Форсье Дж., Биссекс П. и др. Django. Разработка веб-приложений на Python / Дж Форсье, СПб.: «Символ-Плюс», 2014. – 456с.

Условия реализации программы (оборудование, инвентарь, специальные помещения, ИКТ и др.)

Компьютерный класс с установленным программным обеспечением Python 3.7, PyCharm, PyQt5, QtDesigner, Django, Notepad++;
локальная сеть с доступом в Интернет;
интерактивная доска/панель.

Планируемые результаты освоения программы

Содержание программы позволяет формировать ряд компетентностей как предметных, так и метапредметных, в числе которых: ИКТ-компетентность, коммуникативная компетентность, учебно-познавательная компетентность и др.

В результате изучения программы учащиеся должны обладать следующими предметными результатами:

- знать историю, эволюцию и место языка Python среди языков программирования высокого уровня;
- знать синтаксис, основные алгоритмические конструкции и парадигмы программирования языка программирования Python;
- знать основные подходы к созданию программ на высокоуровневом языке программирования Python;
- уметь распознавать необходимость применения той или иной алгоритмической конструкции при решении задач;
- уметь организовывать данные для эффективной алгоритмической обработки;

- уметь реализовывать полученные навыки для решения задач в олимпиадах по программированию различного уровня;
- уметь решать типовые задачи ЕГЭ по информатике из раздела «Программирование»;
- разрабатывать программы и консольные приложения на языке программирования Python;
- уметь реализовать полученные знания с помощью проектной работы, направленной на решение прикладной задачи в интересующей учащегося научной области.

Метапредметные результаты:

- представление о роли компьютерного программирования в развитии общества, изменении содержания и характера деятельности человека.
- применение полученных знаний в области прикладной математики и информатики;
- получение опыта дистанционного обучения с помощью современных онлайн-сервисов Интернета;
- владение навыками познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем;
- стремление к самостоятельной работе, усовершенствованию известных моделей и алгоритмов, созданию творческих проектов.

Личностные результаты:

- готовность к саморазвитию, самореализации;
- развитие способностей к абстрактному мышлению, анализу и синтезу;
- требовательность к себе, ответственность и добросовестность;
- сформированность коммуникативных навыков: сотрудничество и работа в команде, успешное распределение ролей;
- определение индивидуальной образовательной траектории для непрерывного самостоятельного повышения уровня знаний в области IT;
- самостоятельность в приобретении новых знаний, умений и навыков с помощью IT технологий и использования их в практической деятельности;
- расширение личного мировоззрения;
- осознанный выбор высшего образовательного учреждения и специальности в информационно-технологической или инженерно-технической областях для продолжения обучения.

Также результатом освоения программы будет сформированность следующих ключевых компетенций.

1. Учебно-познавательные компетенции:

- 1.1. способность разрабатывать методы решения нестандартных задач и новые методы решения традиционных задач;
- 1.2. способность воспринимать математические, естественно-научные и профессиональные знания, умение самостоятельно приобретать, развивать и применять их для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте;
- 1.3. организация взаимодействия учащегося и преподавателя, принятие управленческих решений в условиях различных мнений;
- 1.4. умение осуществлять планирование, анализ, рефлексию, самооценку своей деятельности (планирование собственной деятельности по разработке программы, владение технологией решения задач с помощью компьютера, компьютерным моделированием);
- 1.5. умение выдвигать гипотезы, ставить вопросы к наблюдаемым фактам и явлениям, оценивать начальные данные и планируемый результат (моделирование и формализация, численные методы решения задач, компьютерный эксперимент, ит.п.);
- 1.6. умение работать со справочной литературой, инструкциями (знакомство с новыми видами ПО, устройствами, анализ ошибок в программе ит.п.);
- 1.7. умение оформить результаты своей деятельности, представить их на современном уровне (построение диаграмм и графиков, средства создания презентаций);
- 1.8. создание целостной картины мира на основе собственного опыта.

2. Коммуникативные компетенции:

2.1. ведение диалога «человек» - «техническая система» (понимание принципов построения интерфейса, работа с диалоговыми окнами ит.д.);

2.2. понимание факта многообразия языков, владение языковой, лингвистической компетенцией (в том числе – формальных языков, систем кодирования, языков программирования; владение ими на соответствующем уровне);

2.3. владение телекоммуникациями для организации общения с удаленными собеседниками (понимание возможностей разных видов коммуникаций, нюансов их использования ит.д.);

2.4. умение работать в группе, искать и находить компромиссы (работа над совместным программным проектом, взаимодействие в сети ит.д.);

2.5. умение строить общение с представителями других взглядов (существование в сетевом сообществе, телекоммуникации с удаленными собеседниками ит.п.);

2.6. иметь позитивные навыки общения.

2.7. умение представить себя устно, владение стилевыми приемами оформления текста (монолог, диалог, умение задать вопрос, привести довод при устном ответе, защите проекта и т.п.);

3. Информационные компетенции:

3.1. знакомство с компьютером как с устройством по работе с информацией, получение технических навыков по работе с различными устройствами и приборами (наушники, колонки, принтер, сканер, web-камера ит.п.);

3.2. способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности;

3.3. владение способами работы с информацией:

- ♣ поиск в каталогах, поисковых системах, иерархических структурах;

- ♣ извлечение информации с различных носителей;

- ♣ систематизация, анализ и отбор информации (разные виды сортировки, фильтры, запросы, структурирование файловой системы, проектирование баз данных ит.д.);

- ♣ технические навыки сохранения, удаления, копирования информации ит.п.

- ♣ преобразование информации (из графической – в текстовую, из аналоговой – в цифровую ит.п.);

3.4. владение навыками работы с различными носителями и источниками информации (книгами, учебниками, мультимедийные справочники, электронные учебники, Интернет ресурсы, и. т.п.);

3.5. критическое отношение к получаемой информации, умение выделять главное, оценивать степень достоверности (релевантность запроса, сетевые мистификации, и т.п.);

3.6. умение применять информационные и телекоммуникационные технологии для решения широкого класса учебных задач.

Перечисленные компетенции являются ориентирами для конкретизации педагогом целей проводимых занятий и контроля результативности обучения.

КАЛЕНДАРНО-УЧЕБНЫЙ ГРАФИК

№ п/п	Дата проведения по плану	Дата фактическая	Форма занятия	Кол-во часов	Тема занятия	Форма контроля
1.			Лекция	1	Установка и настройка среды программирования Python, Типы данных и составные объекты в Python	учет решенных задач
2.			Лекция	1	Управляющие конструкции ходом выполнения программ на Python. Пользовательские функции и основы функционального программирования на языке Python	учет решенных задач
3.			Лекция	1	Модульное программирование. Стандартные и нестандартные модули. Модуль чисел с плавающей точкой Decimal. Модуль Fractions.	учет решенных задач
4.			Практ.	1	Модуль стандартных математических функций Math. Модуль	учет решенных задач

					абстрактных базовых классов Numbers	
5.			Лекция	1	Модуль псевдослучайных чисел Random. Модуль для работы с массивами Array	учет решенных задач
6.			Лекция	1	Модуль сортировки списков Bisect. Численные методы решения алгебраических уравнений.	
7.			Практ.	1	Численные методы интерполирования. Статистическая обработка данных методом наибольших квадратов	учет решенных задач
8.			Практ.	1	Стандартные модули обработки строк String, Codes	учет решенных задач
9-10.			Практ.	2	Решение задач	учет решенных задач
11			Лекц.	1	Словари и их методы в Python	
12-13.			Практ.	2	Решение задач	учет решенных задач
14.			Лекц.	1	Рекурсия в Python.	
15-16.			Практ.	2	Решение задач	Практ. задание
17.			Лекц.	1	Работа с файлами в Python	

18.			Практ.	1	Решение задач	учет решенных задач
19.			Лекц.	1	Исключения в языках Программирования. Ошибки и исключения в Python. Обработка исключений	
20.			Лекц.	1	Наследование. Полиморфизм и абстракция	
21.			Практ.	1	Создание класса и его экземпляров	учет решенных задач
22.			Лекц.	1	Асимптотический анализ алгоритмов. Алгоритмы сортировки и поиска	
23.			Практ.	1	Решение задач	учет решенных задач
24.			Лекц.	1	Жадные алгоритмы	
25- 32.			Практ.	8	Разработка и программирование проекта.	
33- 34.			Практ.	2	Защита проекта в аудитории.	Защита проекта

Упражнения, уменьшающие усталость при работе за компьютером

Упражнения для улучшения осанки

«Глядя в небо»

Цель этого упражнения - устранение вредных эффектов от неподвижного сидения в течение длительного периода времени и профилактика грыжи межпозвоночных дисков поясничного отдела.

Упражнение: стоя, руки лежат на бедрах. Медленно отклоняться назад, глядя в небо. Вернуться в исходное положение.

«Египтянин»

Цель - укрепление мышц задней стороны шеи для улучшения осанки и предотвращения болей в области шеи. Упражнение способствует предотвращению:

- синдрома запястного канала;
- вытягиванию шеи вперед;
- дисфункции височно-нижнечелюстного сустава;
- грыжи межпозвоночных дисков шейного отдела;
- синдрома верхней апертуры грудной клетки;

Упражнение: сидя или стоя, взгляд направлен прямо, а не вверх и не вниз. Надавив указательным пальцем на подбородок, сделать движение шеей назад. В этом положении следует оставаться в течение 5 секунд.

«Абра-кадабра»

Цель упражнения: усиление кровотока к ладоням; снятие напряжения в запястьях и ладонях; удаление продуктов распада из области запястного канала и ладоней.

Упражнение: сидя, руки лежат на предплечьях, запястья должны быть вытянуты ладонями вниз.

Абра: медленно сжать ладони в кулак.

Кадабра: медленно разжать кулаки.

Для достижения желаемого результата эти упражнения следует повторять не менее 10 раз.

Упражнения для глаз

Эффективная профилактическая мера - зрительная гимнастика. Ее проводят дважды: через 7-8 минут от начала работы ребенка на компьютере и после ее окончания. Непродолжительная гимнастика - около одной минуты - проста и доступна каждому. Например, сидя за компьютером, ребенок поднимает глаза кверху и, представив летящего там мотылька или бабочку, следит за их полетом из одного угла комнаты в другой, не поворачивая при этом головы - двигаться должны только глаза!

Есть, конечно, и другие нехитрые правила:

1. На счет 1-4 закрыть глаза, не напрягая глазные мышцы, на счет 1-6 широко раскрыть глаза и посмотреть вдаль. Повторить 4-5 раз.

2. Посмотреть на кончик носа на счет 1-4, а потом перевести взгляд вдаль на счет 1-6. Повторить 4-5 раз.

3. Не поворачивая головы, медленно делать круговые движения глазами вверх-вправо-вниз-влево и в обратную сторону: вверх-влево-вниз-вправо. Затем посмотреть вдаль на счет 1-6. Повторить 4-5 раз.

4. Держа голову неподвижно, перевести взор, зафиксировав его, на счет 1-4 вверх, на счет 1-6 прямо; затем аналогично вниз-прямо, вправо-прямо, влево-прямо. Прodelать движение по диагонали в одну и другую стороны, переводя глаза прямо на счет 1-6. Повторить 3-4 раза.

5. Не поворачивая головы, закрытыми глазами "посмотреть" направо на счет 1-4 и прямо на счет 1-6. Поднять глаза вверх на счет 1-4, опустить вниз на счет 1-4 и перевести взгляд прямо на счет 1-6. Повторить 4-5 раз.

6. Посмотреть на указательный палец, удаленный от глаз на расстоянии 25-30 см, и на счет 1-4 приблизить его к кончику носа, потом перевести взор вдаль на счет 1-6. Повторить 4-5 раз