

УПРАВЛЕНИЕ ОБРАЗОВАНИЯ ГОРОДА ПЕНЗЫ  
МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
«ЛИЦЕЙ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ УПРАВЛЕНИЯ № 2» Г. ПЕНЗЫ

ОДОБРЕНА  
Научно-методическим  
советом  
МБОУ ЛСТУ № 2 г. Пензы  
Протокол № 1  
от « 30 » августа 2023 г.

ПРИНЯТА  
Педагогическим советом  
МБОУ ЛСТУ № 2 г. Пензы  
Протокол № 9  
от « 29 » августа 2023 г.

УТВЕРЖДЕНА  
Приказом директора  
МБОУ ЛСТУ № 2 г. Пензы  
(*Шонкина Т.Н.*)  
Приказ № 224 от «29» 08 2023 г.



Дополнительная общеобразовательная общеразвивающая  
Программа технической направленности  
**«Программирование»**

Возраст учащихся: 15 – 17 лет

Срок реализации: 3 года.

Автор-составитель:

*Акимова Ирина Викторовна,*

педагог дополнительного образования

г. Пенза, 2023 г.

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная общеразвивающая программа «Программирование»

- по содержанию является **технической**,
- по уровню освоения – *базовой*,
- по форме организации - *очной, групповой*,
- по степени авторства – *модифицированной*.

Программа разработана в соответствии со следующими нормативно – правовыми документами:

- Федеральный Закон РФ от 29.12.2012 г. № 273 «Об образовании в РФ»;
- Федеральный Закон от 31 июля 2020 года № 304-ФЗ «О внесении изменений в Федеральный закон «Об образовании в Российской Федерации» по вопросам воспитания обучающихся»;
- Приказ Министерства образования и науки Российской Федерации от 27 июля 2022 г. № 629 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи", утвержденных постановлением Главного государственного санитарного врача Российской Федерации от 28.09.2020 г. N 28 «Об утверждении санитарных правил СП 2.4.3648-20».
- Распоряжение Правительства РФ от 31.03.2022 № 678-р «Об утверждении Концепции развития дополнительного образования детей до 2030 г.»;
- Национальный проект «Образование» (утвержден Президиумом Совета при Президенте РФ по стратегическому развитию и национальным проектам (протокол от 24.12.2018 г. № 16);
- Целевая модель развития региональной системы дополнительного образования детей (приказ Министерства просвещения РФ от 3 сентября 2019 г. № 467);
- Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы) (разработанные Минобрнауки России совместно с ГАОУ ВО «Московский государственный педагогический университет», ФГАУ «Федеральный институт развития образования», АНО ДПО «Открытое образование», 2015 г.) (Письмо Министерства образования и науки РФ от 18.11.2015 № 09-3242);
- Методические рекомендации по проектированию дополнительных общеразвивающих программ нового поколения (включая разноуровневые программы) в области физической культуры и спорта (разработаны ФГБУ «Федеральным центром организационно – методического обеспечения физического воспитания» в 2021 году). Для физкультурно-спортивной направленности.
- Устав МБОУ ЛСТУ №2 г. Пензы;
- «Положение о дополнительной общеобразовательной общеразвивающей программе МБОУ ЛСТУ №2 г. Пензы».

**Актуальность программы** обусловлена тем, что в настоящее время технологии программирования являются востребованными компетенциями как на рынке образовательных услуг, так и на важными составляющими будущей подготовки учащихся в рамках продолжения образования в высших и средне специальных учебных заведениях.

Различные рейтинги языков программирования в настоящее время продолжают ставить на лидирующие позиции язык C++, Python, что также делают рассматриваемую программу актуальной.

**Новизна и отличительные особенности** данной программы от уже существующих программ в этой области заключаются в том, что учащиеся работают не с теоретическим

материалом, который им надо изучить самостоятельно или вместе с педагогом. Обучение идет на основе использования деятельностного подхода – решениям задач различного уровня сложности.

Специфика предполагаемой деятельности обусловлена реальными потребностями современного социума и растущим спросом на информационные услуги. Практические занятия по программе связаны с использованием прикладного программного обеспечения, позволяющим осуществлять полный цикл решения задачи с помощью языка программирования.

**Педагогическая целесообразность** программы объясняется необходимостью углубленного изучения темы для детей, желающих заниматься разработками современных языками программирования.

#### **Адресат программы:**

Образовательная программа «Программирование» рассчитана на детей в возрасте от 15 лет до 17 лет.

#### **Объем и сроки реализации** программы:

Программа рассчитана на 3 года обучения, с общим количеством часов – 216.

**Форма реализации** образовательной программы очная. Основной формой обучения является занятие.

#### **Режим проведения занятий** соответствует возрасту учащихся:

1 раз в неделю по 2 часа (1 учебный час - 45 мин);

**Концепция представляемой программы** состоит в том, что в ее содержании предусмотрена **практико-ориентированная деятельность** учащихся по реализации программных проектов на одном из современных языках программирования.

Эта деятельность будет способствовать социализации школьников, становлению их будущей профессиональной траектории. Блочно-модульное построение учебной программы, разнообразие заданий внутри каждой темы, их различная направленность и разный уровень трудности позволяют каждому ученику реализовать свое право выбора заданий, соответствующих их возможностям и способностям, развиваться в зоне ближайшего развития и взять ответственность за создание своего индивидуального учебного плана.

При выполнении учебной программы будут созданы условия для реализации **компетентностного подхода** в образовании учащихся.

#### **Особенности организации образовательного процесса**

##### **Уровни обучения**

Ознакомительный уровень (*1 год обучения.*)

Знакомство с основами современного языка программирования (C++, Python).

Базовый уровень (*2 год обучения.*)

Решение задач углубленного уровня.

Продвинутый уровень (*3 год обучения.*)

Решение задач углубленного уровня.

**Цель курса:** освоить основы программирования на одном из современных языков программирования (Python, C++), что должно способствовать интеллектуальному и творческому развитию школьников, совершенствованию их логического и алгоритмического мышления.

#### **Задачи курса:**

##### на личностном уровне:

- совершенствованию практических навыков при проектировании программ с использованием языка высокого уровня;
- формированию образного, логического и алгоритмического мышления школьников;
- развитию навыков анализа и самоанализа;
- воспитанию целеустремленности и усидчивости в процессе творческой работы и учебной деятельности;
- формирование готовности к выбору профильного образования, связанного с профессией программиста или частично связанной с ней.

на предметном уровне:

- научить понятию «язык программирования», «программа»;
- научить понятиям об основных алгоритмических конструкциях: линейной, ветвления, цикла, вспомогательного алгоритма;
- научить работать с основными операторами языка C++: присвоение, ввода-вывода, ветвления, цикла;
- научить основным типам и структурам данных;
- научить реализовывать вспомогательные алгоритмы.

На метапредметном уровне:

- создание и преобразование моделей и схем для решения задач, построение логического рассуждения, включающее установление причинно-следственных связей;
- осуществление выбора наиболее эффективных способов решения задач на компьютере в зависимости от конкретных условий;
- овладение составляющими проектной деятельности, включая умения видеть проблему, формулировать тему и цель проекта, составлять план своей деятельности, осуществлять действия по реализации плана, результат своей деятельности соотносить с целью, классифицировать, наблюдать, проводить эксперименты, делать выводы и заключения, доказывать, защищать свои идеи, оценивать результаты своей работы;
- проектирование и организация своей индивидуальной и групповой деятельности, организация своего времени с использованием ИКТ.

**Методы обучения:** деятельностные (учебная дискуссия, диалог, конференция, деловые и ролевые игры, открытые вопросы, ситуационные задания и т.д.); практические (выполнение лабораторных работ экспериментального и исследовательского характера в реальном и виртуальном режиме, моделирование, др.); самостоятельная работа с литературой (обычной и электронной), цифровыми образовательными ресурсами, Internet-ресурсами и др.

**Технологии обучения:**

- **технология интегрированного обучения** – через объединение знаний и действий из разных предметных областей обеспечивает формирование личностных, регулятивных, познавательных и коммуникативных универсальных учебных действий, устойчивой мотивации учебной деятельности, включая учебные и познавательные мотивы;
- **технология личностно-ориентированного обучения** - обеспечивает развитие личности, творческого и интеллектуального потенциала, формирование его ценностной ориентации в процессе обучения через реализацию принципов сотрудничества и свободы выбора, при этом процессы обучения и учения взаимно согласовываются с учётом механизмов познания, особенностей мыслительных и поведенческих стратегий учащихся;
- **технология проектно-исследовательской деятельности** – обеспечивает формирование и развитие специфических умений и навыков: умение самостоятельно формулировать проблему и ставить задачи для ее решения; осуществлять самоанализ и рефлексию по поводу успешности своей деятельности, находить и отбирать актуальную информацию; представлять результаты работы перед аудиторией, др.
- **информационно-коммуникационные технологии** – реализация и совершенствование умений работать с информацией (учебник, журнал, газета, виртуальные источники), использовать Internet-ресурсы, различные офисные приложения и др. для формирования информационной культуры и компьютерной грамотности.

Структура программы внеурочной деятельности «Программирование» предполагает, как изучение теоретического материала, так и выполнение практических работ, разработкой исследовательского проекта. .

На занятиях используется групповая работа обучающихся с презентацией полученных группами результатов и выводов для всего класса. При такой организации обучающимся предоставляется **право выбора** практических работ согласно их запросам, что позволяет охватить значительный объем экологического материала, развить общеучебные и специальные

умения (работа с приборами, постановка опытов и т.п.), коммуникативные умения старшеклассников, научить эффективно работать в команде.

### Тематическое планирование 216 часов

№	Название темы	Кол-во часов	Форма контроля
<b>1 год (Ознакомительный уровень)</b>			
1	Основы языка программирования C++	6	Практическая работа
2	Типы данных языка C++	6	Практическая работа
3	Основные операторы: операторы присвоения, ввода-вывода	6	Практическая работа
4	Основные операторы: операторы ветвления	6	Практическая работа
5	Основные операторы: операторы цикла	8	Практическая работа
6	Структурированные типы данных: массивы	8	Практическая работа
7	Структурированные типы данных: структуры	8	Практическая работа
8	Работа с файлами	8	Практическая работа
9	Структурированные типы данных: объединения	8	Практическая работа
10	Структурированные типы данных: классы	8	Практическая работа
<b>Итого за год:</b>		<b>72</b>	
<b>2 год (Базовый уровень)</b>			
13	Структурированные типы данных: строки. Применимые действия.	8	Практическая работа
14	Реализация вспомогательного алгоритма: функции	8	Практическая работа
15	Типовые алгоритмы обработки двумерных массивов	8	Практическая работа
16	Типовые алгоритмы обработки двумерных массивов по строкам и по столбцам	8	Практическая работа
17	Решение задач с использованием дополнительного	10	Практическая

	массива «флажков»		работа
18	Примеры решения задач ЕГЭ по информатике с использованием языка программирования	10	Практическая работа
19	Подготовка индивидуального проекта	16	Защита проекта
20	Резерв	4	
<b>Итого за год:</b>		<b>72</b>	
<b>3 год (Продвинутый уровень)</b>			
<b>№</b>	<b>Название темы</b>	<b>Кол-во часов</b>	<b>Форма контроля</b>
1	Работа в режиме WindowsForms	8	Практическая работа
2	Работа с формами в VisualStudio 2022	8	Практическая работа
3	Основные компоненты в VisualStudio 2022	8	Практическая работа
4	Разработка проекта «Текстовый редактор»	8	Практическая работа
5	Разработка проекта «Шифровальщик сообщений»	10	Практическая работа
6	Разработка проекта «СУБД»	10	Практическая работа
7	Разработка проекта «Планирование бюджета»	16	Практическая работа
8	Выполнение индивидуального проекта	4	Практическая работа
9	Защита проекта		Практическая работа
10	Резерв		Практическая работа
<b>Итого за год:</b>		<b>72</b>	

## Содержание программы

### Основы языка программирования C++

Алфавит, зарезервированные слова, идентификаторы, константы, структура программы на языке Паскаль, типы данных. Структура программы: основные разделы.

### Типы данных языка C++

Понятие типа данных, простые и структурированные типы. Типы данных `int`, `float`, `double`, `char`, `bool`.

### Основные операторы: операторы присвоения, ввода-вывода

Понятие о простом и составном операторе. Операторные скобки `begin-end`. Оператор присвоить, оператор ввода, оператор вывода. Примеры решения задач.

### Основные операторы: операторы ветвления

Оператор ветвления, полная и сокращенная форма. Примеры решения задач.

### Основные операторы: операторы цикла

Понятие циклического алгоритма, три вида цикла в языке C++. Цикл с параметром `for`, цикл с предусловием `while`, цикл с постусловием `do`. Примеры решения задач.

### Структурированные типы данных: массивы

Понятие массива в языке программирования, способы описания массива, основные возможности в программе: различные способы ввода массива, вывод массива на экран, первичная обработка элементов массива. Примеры решения задач.

### Структурированные типы данных: структуры

Понятие структуры в языке программирования, способы описания структур, основные возможности в программе. Примеры решения задач.

### Работа с файлами

Определение файла. Файлы прямого и последовательного доступа. Описание файлового типа. Последовательность работы с файлами в C++

### Структурированные типы данных: объединения

Понятие объединения в языке программирования, способы описания объединения, основные возможности в программе. Примеры решения задач.

### Структурированные типы данных: классы

Парадигмы программирования. Процедурное программирование. Модульное программирование. Объектно-ориентированное программирование. Понятия класс, объект (экземпляр), ссылка. Определение объектов и типов. Инкапсуляция. Наследование. Полиморфизм. Создание объекта – выделение памяти. Уничтожение объекта – освобождение памяти. Работа с объектом (экземпляром).

### Структурированные типы данных: строки. Применимые действия

Описание типа строка, основные процедуры и функции по работе со строками. Примеры решения задач.

### Реализация вспомогательного алгоритма: функции

Понятие вспомогательного алгоритма. Описание функций в C++. Примеры решения задач.

### Типовые алгоритмы обработки двумерных массивов

Задачи заполнения и вывода элементов двумерного массива, нахождение суммы элементов, выбор по условию, поиск минимального и максимального элемента, вставка и удаление элементов.

### Типовые алгоритмы обработки двумерных массивов по строкам и по столбцам

Поиск минимального и максимального элементов по строкам и по столбцам. Вставка и удаление строк, столбцов двумерного массива. Сортировка строк и столбцов.

### Решение задач с использованием дополнительного массива «флажков»

Проблема решения задачи, в которой необходимо отметить элементы, удовлетворяющие условиям. Использование дополнительных переменных и «флаговых массивов».

### Примеры решения задач ЕГЭ по информатике

Использование элементов программирования при решении задач компьютерного ЕГЭ по информатике.

### Подготовка индивидуального проекта

Выбор темы проекта, разработка плана проекта, реализация проекта.

### **Практическая работа № 1.** Специфика влияния факторов городской среды на растения

Рассмотри решение следующих задач:

1. Составить программу нахождения периметра и площади квадрата (значения длин сторон вводятся с клавиатуры).

2. Проверить, является ли введенное с клавиатуры число четным.

Решение.

Задание 1.

```
#include "stdafx.h"
#include <iostream>
using namespace std;
int main()

{setlocale(LC_ALL,"Russian");
  double a,s,p;
  cin>>a;
  s=a*a; p=4*a;
  cout<<"периметр= "<<p<<endl;
  cout<<"площадь= "<<s<<endl;
  system("pause");
  return 0;
}
```

Задание 2.

```
#include "stdafx.h"
#include <iostream>
using namespace std;
int main()
{setlocale(LC_ALL,"Russian");
  int p;
  cin>>p;
  if (p%2==0) cout<<"четное"<<endl; else cout<<"нечетное"<<endl;
  system("pause");
  return 0;
}
```

### **Практическая работа № 2.**

Рассмотри решение следующих задач:

1. Найти сумму k чисел, введенных с клавиатуры.

Решение.

```
#include <iostream>
using namespace std;
int main(int argc, char** argv) {
  int k,s=0;
  cout<<"k= ";
  cin>>k;
  int c;
  for (int i=1; i<=k;i++)
  {
    cin>>c;
    s+=c;
  }
  cout<<"sum= "<<s<<endl;
  system("pause");
}
```



```

    return 0;
}
2. Вычислить n!.
Решение.
#include <iostream>
using namespace std;
int main(int argc, char** argv) {
    int n,p=1;
    cout<<"n= ";
    cin>>n;
    for (int i=2; i<=n;i++)
        p=p*i;
    cout<<"n!= "<<p<<endl;
    system("pause");
    return 0;
}

```

3. Вывести на экран все трехзначные числа, в которых две одинаковые цифры.  
Решение.

```

#include <iostream>
using namespace std;
int main()
{setlocale(LC_ALL,"Russian");
  for (int a=1; a<=9; a++)
    for (int b=0; b<=9;b++)
      for (int c=0; c<=9;c++)
        if ((a==b)||(a==c)||(b==c))
          cout<<a<<b<<c<<endl;
  system("pause");
  return 0;
}

```

### Практическая работа № 3.

Рассмотри решение следующих задач:

1. Найти сумму положительных элементов массива.

Решение.

```

#include <iostream>
#include<time.h>
using namespace std;
int main()
{
    srand(time(NULL));
    const int N = 10;
    int Arr[N];
    for (int i=0; i<N; i++){
        Arr[i] = rand()%100 -50;
        cout<<Arr[i]<<endl;
    }
    int s=0;
    for (int i=0; i<N; i++)
        if (Arr[i]>0) s+=Arr[i];
    cout<<"sum= "<<s<<endl;
    system("pause");
    return 0;
}

```

2. Проверить, встречается ли в массиве введенное с клавиатуры число.

Решение.

```
#include <iostream>
#include <cstdlib>
#include <ctime>
using namespace std;
int main()
{ srand(time(NULL));
  const int N = 10;
  int Arr[N];
  for (int i=0; i<N; i++){
    Arr[i] = rand()%100 +1;
    cout<<Arr[i]<<endl;
  }
  int k;
  cout<<"k= "; cin>>k;
  bool f=0;
  for (int i=0; i<N; i++)
    if (Arr[i]==k) f=1;
  if (f) cout<<"yes"<<endl;
  else cout<<"no"<<endl;
  system("pause");
  return 0;
}
```

3. Найти в массиве максимальный элемент.

Решение.

```
#include <iostream>
#include <cstdlib>
#include <ctime>
using namespace std;
int main()
{ srand(time(NULL));
  const int N = 10;
  int Arr[N];
  for (int i=0; i<N; i++){
    Arr[i] = rand()%100 +1;
    cout<<Arr[i]<<endl;
  }
  int max=Arr[0];
  for (int i=0; i<N; i++)
    if (Arr[i]>max) max=Arr[i];
  cout<<"max= "<<max<<endl;
  system("pause");

  return 0;
}
```

4. Найти в массиве номер минимального элемента.

```
#include <iostream>
#include <cstdlib>
#include <ctime>
using namespace std;
int main()
{ srand(time(NULL));
  const int N = 10;
  int Arr[N];
```

```

for (int i=0; i<N; i++){
    Arr[i] = rand()%100 +1;
    cout<<Arr[i]<<endl;
}
int nmin=0;
for (int i=0; i<N; i++)
    if (Arr[i]<Arr[nmin]) nmin=i;
cout<<"nmin= "<<nmin+1<<endl;
system("pause");
return 0;
}

```

## **Проекты по программированию**

Проекты по программированию представляют собой проекты, результатами которых является программа для решения той или иной задачи. Особенностью является то, что одна и та же задача в зависимости от уровня проработки, может быть решена как начинающим, так и опытным программистом.

При выполнении проекта по программированию учащиеся имеют следующие возможности: получить умения самостоятельно формулировать цели и задачи проекта, планировать свою деятельность, повысить уровень программирования на языке Python, получить умение представления результатов своей деятельности.

Проект может разрабатываться индивидуально или группой учащихся. Если задача достаточно сложная, то проекта может быть разбит на подзадачи, подпроекты. Каждую подзадачу будут выполнять различные группы участников проекта. Например, одна группа занимается разработкой алгоритма, другая группа – непосредственно написанием и отладкой кода на языке Python, третья – подготовкой к презентации проекта.

В помощь участникам проекта можно предложить заполнить следующий учетный лист.

### ***Проект по программированию***

*Тема проекта:*

*Творческое название (при наличии):*

*Основополагающий вопрос:*

*Авторы:*

- 1.
- 2.
- 3.

...

*Предметная область:*

*Краткая аннотация:*

*Этапы выполнения проекта:*

При подготовке к защите проекта учащимся необходимо подготовить презентацию и доклад, в котором отражаются основные этапы разработки программы, представлен алгоритм решения задачи, листинг программы, основные результаты работы. Можно предложить в помощь учащимся заполнить следующий чек-лист:

1. Аннотация.
2. Содержание.
3. Постановка задачи:
  - a. Возможности использования программы;
  - b. Описание интерфейса;
4. Формализация алгоритма:
  - a. Перечень подпрограмм (при наличии);
  - b. Описание алгоритма (блок-схема или подробное словесное описание алгоритма);
5. Листинг программы (текст программы).
6. Тестовые примеры

- а. Результаты работы;
  - б. Скрин-шоты результатов работы.
7. Описание размещения.
  8. Требования к программным и аппаратным средствам.
  9. Для оценивания проекта могут быть разработаны специальные оценочные листы.
- Ниже представлен пример оценочного листа:

### **Комплекс организационно-педагогических условий**

#### **Формы аттестации и система оценки результативности обучения по программе**

Формы аттестации: наблюдение, опрос, проектная работа, практическая работа.

Для оценивания результативности обучения по программе используются следующие показатели: теоретическая подготовка учащихся, практическая подготовка, общеучебные умения и навыки (метапредметные результаты), личностное развитие учащихся в процессе освоения дополнительной образовательной программы.

Применяется 5- бальная шкала (низкий уровень: 1-2 балла, средний уровень: 3 балла, высокий уровень: 4-5 баллов).

#### **Условия реализации программы**

##### **Материально-технические ресурсы:**

<b>№</b>	<b>Название</b>	<b>Количество</b>
1	Учебная аудитория (групповые занятия)	1
2	Доска школьная (магнитно-маркерная)	1
3	Стол письменный	1
4	Стул ученический	По количеству учащихся
5	Стол ученический (парта)	По количеству учащихся
6	Раковина для мытья рук	1
7	Колонки (звуковые)	1 комплект
8	Проектор	1

**Информационные ресурсы:** оргтехника, интернет-ресурсы.

<b>№</b>	<b>Название</b>	<b>Количество</b>
1	Рабочее место ученика (стул, стол, персональное компьютерное устройство)	По количеству учащихся
2	Локальная сеть с выходом в сеть Интернет	1
3	Сеть wi-fi с выходом в сеть Интернет	1
4	Компьютерное устройство с сетевой ролью сервера	1
5	Компьютер учительский	1
6	Принтер	1

**Расходные материалы** (приобретаются учащимися самостоятельно)

<b>№</b>	<b>Название</b>	<b>Количество</b>
1	Тетрадь	1 на учащегося

#### **Контрольно- измерительные материалы**

## 1. Оценочные материалы

### Оценивание предметных результатов обучения по программе:

Показатели (оцениваемые параметры)	Критерии	Методы диагностики	Степень выраженности оцениваемого качества		
			Низкий уровень (1-2 балла)	Средний уровень (3-4 баллов)	Высокий уровень (5 баллов)
Теоретические знания по основным разделам программы	Соответствие теоретических знаний учащегося программным требованиям	Наблюдение, тестирование, контрольный опрос и др.	Учащийся овладел менее чем половиной знаний, предусмотренных программой	Объем усвоенных знаний составляет более 50%	Учащийся освоил практически весь объем знаний, предусмотренных программой за конкретный период
Практические умения и навыки, предусмотренные программой	Соответствие практических умений и навыков программным требованиям	Контрольное задание	Практические умения и навыки неустойчивые, требуется постоянная помощь по их использованию	Овладел практическими умениями и навыками, предусмотренными программой, применяет их под руководством педагога	Учащийся овладел в полном объеме практическими умениями и навыками, практические работы выполняет самостоятельно, качественно

### Оценивание метапредметных результатов обучения по программе:

Показатели (оцениваемые параметры)	Критерии	Методы диагностики	Степень выраженности оцениваемого качества		
			Низкий уровень (1-2 балла)	Средний уровень (3-4 баллов)	Высокий уровень (5 баллов)
Учебно-познавательные умения	Самостоятельность в решении познавательных задач	Наблюдение	Учащийся испытывает серьезные затруднения в работе, нуждается в постоянной помощи и контроле педагога	Учащийся выполняет работу с помощью педагога	Учащийся выполняет работу самостоятельно, не испытывает особых затруднений

Учебно-организационные умения и навыки	Умение планировать, контролировать и корректировать учебные действия, осуществлять самоконтроль и самооценку	Наблюдение	Учащийся испытывает серьезные затруднения в анализе правильности выполнения учебной задачи, собственные возможности оценивает с помощью педагога	Учащийся испытывает некоторые затруднения в анализе правильности выполнения учебной задачи, не всегда объективно осуществляет самоконтроль	Учащийся делает осознанный выбор направления учебной деятельности, самостоятельно планирует выполнение учебной задачи и самостоятельно осуществляет самоконтроль
Учебно-коммуникативные умения и навыки	Самостоятельность в решении коммуникативных задач	Наблюдение	Учащийся испытывает серьезные затруднения в решении коммуникативных задач, нуждается в постоянной помощи и контроле педагога	Учащийся выполняет коммуникативные задачи с помощью педагога и родителей	Учащийся не испытывает трудностей в решении коммуникативных задач, может организовать учебное сотрудничество
Личностные качества	Сформированность моральных норм и ценностей, доброжелательное отношение к окружающим, мотивация к обучению	Наблюдение	Сформировано знание на уровне норм и правил, но не использует на практике	Сформированы, но не достаточно актуализированы	Сформированы в полном объеме

### Список литературы

1. Акимова И.В. Лабораторный практикум по программированию. – Пенза: ПГПУ, 2011. – 44 с.
2. Голощапов А. Microsoft VisualStudio 2010. – СПб.: БХВ-Петербург, 2010. – 544 с.
3. Загребальный, С.Л. Основы визуального программирования в среде VisualStudio 2010. – Краматорск, ДГМА, 2012. – 160 с.
4. Лаптев В.В., Морозов А.В. С++. Объектно-ориентированное программирование. Задачи и упражнения. – СПб.: Питер. – 2007. – 288 с.
5. Павловская, Т.А. С/С++. Программирование на языке высокого уровня. – СПб.: Питер, 2004.
6. Пауэрс, Л. Microsoft VisualStudio 2008. – СПб.: БХВ-Петербург, 2009. – 1200 с.
7. Пахомов, Б.И. С/С++ и MSVisualC++ 2010 для начинающих. – СПб.: БХВ-Петербург, 2011. – 736 с.
8. Павловская, Т.А. С/С++. Программирование на языке высокого уровня. – СПб.: Питер, 2004.
9. Поляков К. Ю. Информатика. 10 класс. Базовый и углубленный уровни : учебник : в 2 ч. Ч. 2 / К. Ю. Поляков, Е. А. Еремин. – М. : БИНОМ. Лаборатория знаний, 2016. – 352 с.
10. Поляков К. Ю. Информатика. 11 класс. Базовый и углубленный уровни : учебник: в 2 ч. Ч. 2 / К. Ю. Поляков, Е. А. Еремин. – М. : БИНОМ. Лаборатория знаний, 2016. – 304 с.
11. Прата С. Язык программирования С++. Лекции и упражнения. – М.: Диалектика-Вильямс, 2018. – 1244 с.
12. Страуструп, Б. Язык программирования С++. Специальное издание. – М. : Бином, 2011. – 1136 с.

### Internet – ресурсы:

1. <https://visualstudio.microsoft.com/ru/> Последние сведения о загрузках, новости и советы по следующему выпуску VisualStudio
2. <http://www.intuit.ru/department/sc/ooepe/> Учебный курс по объектно-ориентированному программированию
3. <https://visualstudio.microsoft.com/ru/> Последние сведения о загрузках, новости и советы по следующему выпуску VisualStudio
4. <http://www.intuit.ru/department/sc/ooepe/> Учебный курс по объектно-ориентированному программированию