


ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ НЕТИПОВОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
«САНКТ-ПЕТЕРБУРГСКИЙ ГОРОДСКОЙ ДВОРЕЦ ТВОРЧЕСТВА ЮНЫХ»

**ПРИНЯТО**

Протокол Малого педагогического совета  
№ 4 от «25» 05 2017г.

  
/М.Ю. Колганов  
руководитель структурного подразделения

**УТВЕРЖДЕНО**

Приказ № 1254-01 от «26» 06 2017г.

Генеральный директор

  
М.Р. Кагунова



Дополнительная общеобразовательная общеразвивающая программа  
«ПРОГРАММИРОВАНИЕ В СРЕДЕ PASCAL ABC И LAZARUS»

Возраст обучающихся: 12-15 лет  
Срок реализации программы: 1 год

**Разработчик:**  
Крикало Тамара Викторовна  
педагог дополнительного образования

**ОДОБРЕНО**

Протокол Методического совета  
№ 11 от «22» 06 2017г.

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

### ***Направленность программы.***

Данная дополнительная общеобразовательная общеразвивающая программа «Программирование на языке PascalABC и Lazarus» (далее Программа) имеет **техническую направленность**.

Уровень освоения программы – **базовый**. В рамках освоения дополнительной общеобразовательной общеразвивающей программы результатом является не только демонстрация собственной компьютерной разработки и представление на итоговой конференции Центра компьютерных технологий, но и участие в Городском конкурсе по программированию и других.

### ***Актуальность программы.***

Программирование является одной из самых востребованных сфер деятельности. Именно язык Pascal, как правило, изучают в школе, поэтому его углубленное изучение востребовано как учащимися, так и их родителями. Pascal ABC относится к свободному программному обеспечению (СПО). Достоинством СПО является общедоступность и бесплатность. Поэтому данная Программа находится в полном соответствии с Концепцией развития разработки и использования свободного программного обеспечения в Российской Федерации.

***Адресат программы*** Программа адресована учащимся 12-15 лет, желающим познакомиться с основами программирования на алгоритмическом языке Паскаль. На обучение по данной программе принимаются учащиеся, закончившие обучение по программе «Основы программирования Паскаль «Все очень просто» по результатам итоговой конференции и рекомендации педагогов. Группа может быть дополнена теми, кто знаком с основами программирования на языке Pascal, эти учащиеся подбираются по результатам тестирования и собеседования.

***Цель программы*** -удовлетворение потребностей учащихся в интеллектуальном совершенствовании и мотивации к творческой деятельности при получении универсальных учебных навыков в области программированию и алгоритмизации на языке Паскаль.

### ***Задачи.***

Обучающие:

- Приобретение знаний в области программирования в среде PascalABC и Lazarus.
- Формирование универсальных знаний и умений для дальнейшего изучения других языков программирования.
- Изучение основных структур данных и алгоритмов их обработки.

Развивающие:

- Развитие логического мышления, умения выполнять логические операции анализа, синтеза, сравнения, классификации, установления аналогий.
- Развитие навыков концентрации и сосредоточенности на выполнении учебной задачи и получении результата.
- Формирование мотивации к углубленному изучению предмета сейчас и в будущем.

Воспитательные:

- Формирование устойчивой мотивации к творческому труду.
- Воспитание инициативности и самостоятельности.
- Воспитание способностей к самоорганизации с целью решения поставленных задач.

## **Условия реализации программы.**

**Условия набора и формирования групп** Списочный состав формируется в соответствии с нормативно-правовыми основами проектирования общеобразовательных программ в т.ч. СанПиН 2.4.4.3172-14. из учащихся 12-15 лет.

Коллектив учащихся формируется из учащихся:

- прошедших обучение по программе «Основы программирования Паскаль «Все очень просто», показавших хорошие результаты и пожелавшие продолжить обучение в области программирования;

- на основе результатов тестирования и собеседования с родителями и детьми. Тестирование проводится согласно документу «Задачи для поступающих в Центр Компьютерных Технологий (компьютерный центр) отдела техники». Тесты позволяют проверить наличие у учащегося логических способностей, гибкости и скорости мышления. Тестирование состоит из двух блоков: 40 арифметических задач и 30 задач на поиск закономерностей. На каждый блок дается 20 минут. Результат теста – общее количество правильно решенных задач. Прходной балл зависит от количества учебных групп в текущем году.

**Объем и срок реализации программы** - 1 год ,144 академических часа.

**Особенности организации образовательного процесса** заключаются в применении современных образовательных технологий, а именно применение технологии проектного обучения в теме, связанной с разработкой собственной программы.

**Формы занятий:** лекции, практические занятия, индивидуальная работа, беседы, конкурсы.

- лекция — объяснение новой теории, во время лекции учащиеся записывают конспект в тетради;

- практическое занятие в компьютерном классе, во время которого на компьютере выполняются задания педагога или осуществляется разработка собственного проекта;

- конкурс, где демонстрируются и оцениваются работы учащихся;

конкурс «Итоговая конференция» проводится в обязательном порядке в конце учебного года;

**Формы организации деятельности на занятии:** используется традиционный способ организации занятия фронтальная – проведение лекции со всем составом учащихся, групповая – проведение занятия в малых группах при разработке собственных программ, индивидуальная – индивидуальные консультации при подготовке программ к конкурсу, коллективная - коллективное решение какой-либо задачи у доски; что позволяет учащимся формулировать и аргументировать свое мнение

**Кадровое обеспечение** – педагогический состав формируется из специалистов отдела техники, имеющих профильное образование или опыт профессиональной деятельности в области программирования.

**Материально-техническое оснащение:** компьютеры, проектор.

На компьютерах должны быть:

- Операционная система Windows 7 или выше;
- Программы Pascal ABC и Lazarus;
- Любой браузер;
- Internet.

### ***Планируемые результаты***

#### Предметные результаты:

- учащиеся получают основные знания по программированию и алгоритмизации на языке Паскаль;
- приобретут навыки работы в среде Lazarus;
- сформируют универсальные знания и умения для дальнейшего изучения других языков программирования.
- изучат основные структуры данных и алгоритмов их обработки.

#### Метапредметные результаты.

##### Учащиеся:

- разовьют творческие способности;
- разовьют интеллектуально-познавательные способности, внимание, память, воображение;
- научатся работать над постановкой и алгоритмизацией задач, планировать свое время на выполнение проекта, творчески представлять итоги своей деятельности.

У ряда учащихся сформируется устойчивая мотивация к дальнейшему изучению современных компьютерных технологий;

#### Личностные результаты.

##### Учащиеся:

- приобретут навыки работы в коллективе, в т.ч. при разработке программного обеспечения и обсуждении задач;
- сформируют ценностные отношения к современным компьютерным технологиям и постоянному повышению собственного образовательного уровня;
- воспитают в себе уважительное отношение к окружающим людям и общечеловеческим ценностям.

**УЧЕБНЫЙ ПЛАН**  
**дополнительной общеобразовательной общеразвивающей программы**  
**«Программирование в среде PascalABC и Lazarus»**

№	Тема	Количество часов			Формы контроля
		Теория	Практика	Всего	
1	Инструктаж по ТБ. Повторение основных типов данных и операторов Паскаля (циклы, массивы, процедуры, файлы)	1	3	4	Проверка правильности работы программ
2	Программы «LINES» и/или «TETRIS»		6	6	Проверка правильности работы программ
3	PascalABC. Полярная система координат	2	6	8	Проверка правильности работы программ
4	PascalABC. Основы машинной графики	2	4	6	Проверка правильности работы программ
5	PascalABC. События: использование мыши, таймер	1	3	4	Проверка правильности работы программ
6	PascalABC. Рекурсия	1	11	12	Проверка правильности работы программ
7	На осенних каникулах – создание собственных программ. Корректировка индивидуальных навыков	2	2	4	Проверка правильности работы программ
8	PascalABC. Указатели. Списки. Графы. Двоичные деревья	6	12	18	Проверка правильности работы программ
9	Lazarus. Среда разработчика. Окна. Объекты. События. Свойства формы. Основные визуальные компоненты	6	8	14	Проверка правильности работы программ

10	Lazarus. Массивы.	2	2	4	Проверка правильности работы программ
11	Lazarus. Рисование по Canvas.	1	1	2	Проверка правильности работы программ
12	Lazarus. Работа с клавиатурой	1	1	2	Проверка правильности работы программ
13	Lazarus. Использование CopyRect	2	2	4	Проверка правильности работы программ
14	Lazarus. Использование таймера и мыши.	1	1	2	Проверка правильности работы программ
15	Lazarus. Закладка DIALOG	1	1	2	Проверка правильности работы программ
16	Параметры Sender и Tag	1	3	4	Проверка правильности работы программ
17	Lazarus. Множества.	1	1	2	Проверка правильности работы программ
18	Lazarus. Создание меню (MainMenu, PopUpMenu)	1	1	2	Проверка правильности работы программ
19	Lazarus. Работа с регионами.	1	1	2	Проверка правильности работы программ
20	Lazarus. Создание нескольких форм. Круглая, многоугольная форма	1	1	2	Проверка правильности работы программ
21	Lazarus. Создание базы данных	1	3	4	Проверка правильности работы программ
22	Lazarus. Создание игр «Toppler» и «Reversy»	1	7	8	Проверка правильности работы программ

23	Создание и отладка программы к итоговой конференции.		16	16	Консультации к самостоятельной работе
24	Проведение итоговой конференции		2	2	Предъявление итоговой программы
25	В каникулы . Корректировка индивидуальных навыков. Подведение итогов полугодия и года	4	6	10	Беседа. Опрос. Проверка тестовых заданий
Итого		<b>40</b>	<b>104</b>	<b>144</b>	

## **РАБОЧАЯ ПРОГРАММА**

### **к дополнительной общеобразовательной общеразвивающей программе «Программирование на языке PascalABC и Lazarus»**

#### ***Задачи.***

Обучающие:

- Приобретение знаний в области программирования в среде PascalABC и Lazarus.
- Формирование универсальных знаний и умений для дальнейшего изучения других языков программирования.
- Изучение основных структур данных и алгоритмов их обработки.

Развивающие:

- Развитие логического мышления, умения выполнять логические операции анализа, синтеза, сравнения, классификации, установления аналогий.
- Развитие навыков концентрации и сосредоточенности на выполнении учебной задачи и получении результата.
- Формирование мотивации к углубленному изучению предмета сейчас и в будущем.

Воспитательные:

- Формирование устойчивой мотивации к творческому труду.
- Воспитание инициативности и самостоятельности.
- Воспитание способностей к самоорганизации с целью решения поставленных задач.

#### ***Планируемые результаты***

Предметные результаты:

- учащиеся получают основные знания по программированию и алгоритмизации на языке Паскаль;
- приобретут навыки работы в среде Lazarus;
- сформируют универсальные знания и умения для дальнейшего изучения других языков программирования.
- изучат основные структуры данных и алгоритмов их обработки.

Метапредметные результаты.

- разовьют творческие способности;
- разовьют интеллектуально-познавательные способности, внимание, память, воображение;
- научатся работать над постановкой и алгоритмизацией задач, планировать свое время на выполнение проекта, творчески представлять итоги своей деятельности.

У ряда учащихся сформируется устойчивая мотивация к дальнейшему изучению современных компьютерных технологий;

Личностные результаты.

- приобретут навыки работы в коллективе, в т.ч. при разработке программного обеспечения и обсуждении задач;
- сформируют ценностные отношения к современным компьютерным технологиям и постоянному повышению собственного образовательного уровня;
- воспитают в себе уважительное отношение к окружающим людям и общечеловеческим ценностям.

#### ***Содержание программы***



- Тема 1. Инструктаж по ТБ. Повторение основных типов данных и операторов Паскаля (циклы, массивы, процедуры, файлы)  
Теория. PascalABC. Повторение основных понятий языка Паскаль.  
Практика. Отработка основных типов данных и операторов.
- Тема 2. Программы «LINES» и/или «TETRIS»  
Практика. Решение задач ЕГЭ предыдущих лет. Программы «Морфинг», «Lines» или «Тетрис».
- Тема 3. PascalABC. Полярная система координат.  
Теория. Полярная система координат. Тригонометрические функции .  
Практика. Движение по окружности, эллипсу, синусоиде. Поворот фигуры относительно произвольной точки.
- Тема 4. PascalABC. Основы машинной графики  
Теория. Масштабирование координат.  
Практика. Построение графиков функций.
- Тема 5. PascalABC. События: использование мыши, таймер  
Теория. События. Прерывания.  
Практика. Программа «Увернись».
- Тема 6. PascalABC. Рекурсия.  
Теория. Создание рекурсивных процедур.  
Практика. Программы «Снежинка», «Веточка».  
Программа «Сапер» . Программа «Шарики» .
- Тема 7. На осенних каникулах – создание собственных программ. Корректировка индивидуальных навыков  
Теория и практика – индивидуальная работа
- Тема 8. PascalABC. Указатели. Списки. Графы  
Теория. Динамическое распределение памяти. Указатели. Списки. Графы  
Практика. Создание списков. Графы Эйлера и Дейкстры.
- Тема 9. Lazarus. Среда разработчика. Окна. Объекты. События.  
Теория. Среда разработчика. Окна. Объекты. События.  
Практика. Освоение среды Lazarus. Работа с формой.
- Тема 10. Lazarus. Массивы.  
Теория. Работа с массивами и файлами.  
Практика. Использование компоненты TabGrid.
- Тема 11. Lazarus. Рисование по Canvas.  
Теория. Работа с Canvas.  
Практика. Использование графических примитивов.
- Тема 12. Lazarus. Работа с клавиатурой  
Теория. Прерывания от клавиатуры  
Практика. Программа «Таракан»
- Тема 13. Lazarus. Использование CopyRect  
Теория. Использование приема.

Практика. Копирование части изображения

Тема 14. Lazarus. Использование таймера и мыши.

Теория. Использование таймера и мыши.

Практика. Программа «Тир».

Тема 15. Lazarus. Закладка DIALOG

Теория. Закладки

Практика. Работа с компонентами закладки Dialog

Тема 16. Параметры Sender и Tag

Теория. Применение параметров.

Практика. Создание программы с использованием параметров.

Тема 17. Lazarus. Множества.

Теория. Множества.

Практика. Игра «Пятнашки»

Тема 18. Lazarus. Создание меню (MainMenu, PopUpMenu)

Теория. Компоненты Label, Edit, Button, Image.

Практика. Программа «калькулятор». Программа «лепестки розы».

Тема 19. Lazarus. Работа с регионами.

Теория. Понятие региона и работа с ним.

Практика. Создание регионов на фотографии.

Тема 20. Lazarus. Создание нескольких форм. Круглая, многоугольная форма

Теория. Круглая, многоугольная форма.

Практика. Создание нескольких форм.

Тема 21. Lazarus. Создание базы данных

Теория. Базы

Практика. Создание базы данных.

Тема 22. Lazarus.

Создание игр «Toppler» и «Reversy»

Теория. Основные понятия.

Практика. Создание игры.

Тема 23. Создание и отладка программы к итоговой конференции

Практика. Создание программы по самостоятельно разработанному алгоритму.

Тема 24. Проведение итоговой конференции

Практика. Защита разработанных программ

Тема 25. В каникулы. Корректировка индивидуальных навыков. Подведение итогов полугодия и года

Теория и практика – индивидуальные консультации. Проведение текущего и итогового контроля.

## ОЦЕНОЧНЫЕ И МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

### Оценочные материалы:

В процессе обучения используются следующие оценочные материалы:

- Тест «Задачи для поступающих в Центр Компьютерных Технологий (компьютерный центр) отдела техники» – один раз в год
- Информационная карта освоения дополнительной образовательной программы – по окончании каждой темы
- Информационная карта оценки метапредметных и личностных результатов освоения образовательной программы – два раза в год
- Анкета выпускника – один раз в год.
- Протокол итоговой конференции – один раз в год.

### Виды и формы контроля:

**Входной контроль** осуществляется с помощью тестирования при приеме в коллектив.

**Текущий контроль** за учащимися осуществляется на каждом практическом занятии, проверяется правильность работы всех выполненных учащимися программ. Знания учащихся оцениваются по пяти-бальной системе по каждой учебной теме, оценки фиксируются в информационной карте освоения дополнительной образовательной программы.

**Итоговый контроль** осуществляется в конце учебного года. Проводится итоговое занятие в виде конференции на которой учащиеся представляют свои собственные разработанные программы.

В ходе обучения учащиеся участвуют в конкурсах, проводимых как внутри учебных групп, так и между группами, а также в конференциях, проводимых Компьютерным центром ГДТЮ. Также учащиеся могут принять участие в Городском конкурсе школьников по программированию и компьютерным работам.

Общую оценку работы коллектива учащихся, отражающую достижение цели дополнительной общеобразовательной и общеразвивающей программы (удовлетворение потребностей учащихся в интеллектуальном совершенствовании и мотивации к творческой деятельности) предлагаю оценивать по следующим параметрам:

- есть учащиеся, разрабатывающие в течение учебного года самостоятельные программы, не предусмотренные учебным процессом;
- большинство учащихся выполнило самостоятельно работу к итоговой конференции;
- есть учащиеся, которые продолжают обучение программированию в ЦКТ или специализированных школах (№30, №239, ФТШ);
- есть учащиеся, выступившие на олимпиадах, конкурсах и конференциях по программированию районного и городского уровня.

В процессе реализации программы используются **современные образовательные технологии**, а именно технология проектного обучения в темах «Создание новогодних программ», «Создание и отладка программы к итоговой конференции», что обеспечивает моделирование предметного и социального содержания профессии программиста. Таким образом, проектные технологии значительно увеличивают интерес обучающихся как к отдельным областям знаний, так и к образованию в целом.

Дидактические материалы: представляет собой раздаточные материалы в виде распечаток с задачами. Для дополнения теоретической части в среде PascalABC существует встроенные учебник и справочник по PascalABC и Lazarus.

### Приложения

- Информационная карта освоения дополнительной образовательной программы.

- Протокол итоговой конференции.
- Информационная карта освоения оценки метапредметных результатов образовательной программы.
- Анкета выпускника.
- Примеры раздаточных материалов.

Информационная карта  
освоения дополнительной образовательной программы.

Группа \_\_\_\_\_

Программа «Программирование в среде PascalABC и Lazarus»

Педагог Крикало Т.В.

	Фамилия / тема	T1	T2	T3	T4	T5	T6	T7	T8
1									
2									
3									
4									
5									
6									
7									
8									
9									
10									

5 – «отлично», учащийся полностью освоил тему, выполнил все предложенные программы с минимальной помощью преподавателя.

4 – «хорошо», учащийся освоил тему, выполнил предложенные программы с подсказками преподавателя.

3 – «посредственно», учащийся частично освоил тему, выполнил необходимый минимум программ с большим количеством подсказок преподавателя.

2 – «неудовлетворительно», учащийся посещал занятия, но тему не освоил.

« - » - учащийся не присутствовал на занятиях.

T1 . . T8 - номера тем

Информационная карта  
оценки метапредметных результатов образовательной программы.

Группа \_\_\_\_\_

Программа «Программирование в среде PascalABC и Lazarus»

Педагог Крикало Т.В.

	Фамилия	Самостоятельная работа в течение года	Создание итоговой программы	Продолжение обучения программированию	Участие в олимпиадах и конкурсах
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					

Выполнение учащимся соответствующего пункта фиксируется знаком «+»

Перечень олимпиад и конкурсов дается под таблицей.

## Анкета выпускника

1. Что являлось главной причиной выбора Центра компьютерных технологий как объединения дополнительного образования ?

- удобство расположения
- близость к месту проживания
- хорошая репутация
- его посещают знакомые (одноклассники)
- другое:

2. Оцени уровень своего интереса к программе дополнительного образования

- всегда с удовольствием посещал занятия
- временами интерес к занятиям снижался
- родители часто уговаривали меня посещать занятия

3. Как, по твоему мнению, влияет обучение программированию на твою успеваемость в школе?

- позитивно влияет (повышает школьную успеваемость)
- влияет скорее позитивно, чем негативно
- никак не влияет
- влияет скорее негативно, чем позитивно
- негативно влияет (снижает школьную успеваемость)
- затрудняюсь ответить

4. Какие на твой взгляд черты личности развивают занятия программированием

- развитие личностных качеств (дисциплина, внимательность, аккуратность и т.п.)
- приобретение полезных навыков и умений расширение кругозора,
- повышение эрудированности
- развитие интеллектуальных способностей (четкость мышления, способность делать выводы и т. п.)
- развитие навыков общения и самопрезентации
- повышение интереса к учебной деятельности
- ничего из перечисленного
- затрудняюсь ответить

5. Доволен ли ты морально-психологическим климатом в коллективе

- да
- нет

6. Нашел ли ты друзей и единомышленников

- да
- нет

## Протокол итоговой конференции.

\_\_\_\_\_ 201\_\_ года

№	Фамилия	класс	группа	название программы	Балл
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					

Педагогическое жюри:

- 1
- 2
- 3

Работы оцениваются по 10-бальной системе

- 10 — диплом 1 степени, программа достойна быть представлена на городском конкурсе;
- 9 — диплом 2 степени, при доработке программа может быть выставлена на городской конкурс;
- 8 — программа хорошая, возможен диплом 3 степени;
- 6-7 — программа удовлетворительная, учащийся освоил темы;
- 1-5 — программа слабая, учащийся освоил не все темы.



## ПРИМЕРЫ РАЗДАТОЧНЫХ МАТЕРИАЛОВ

Классами в Object Pascal называются специальные типы, которые содержат поля, методы и свойства. Как и любой другой тип, класс служит лишь образцом для создания конкретных экземпляров реализации, которые называются объектами.

. В основе классов лежат три фундаментальных принципа, которые называются *инкапсуляция, наследование и полиморфизм*.

### **1. ОСНОВНЫЕ ПОНЯТИЯ**

#### **1.1 Инкапсуляция**

Класс представляет собой единство трех сущностей - полей, методов и свойств. Объединение этих сущностей в единое целое и называется инкапсуляцией. Инкапсуляция позволяет во многом изолировать класс от остальных частей программы, сделать его "самодостаточным" для решения конкретной задачи. В результате класс всегда несет в себе некоторую функциональность. Например, класс `TForm` содержит (инкапсулирует в себе) все необходимое для создания Windows-окна, класс `TMemo` представляет собой полнофункциональный текстовый редактор, класс `TTimer` обеспечивает работу программы с таймером и т. д.

Инкапсуляция представляет собой мощное средство обмена готовыми к работе программными заготовками. Библиотека классов Delphi - это фактически набор "кирпичиков", созданных программистами Borland для построения ваших программ.

#### **1.2 Наследование**

Любой класс может быть порожден от другого класса. Для этого при его объявлении указывается имя класса-родителя:

```
TChildClass = class (TParentClass)
```

Порожденный класс автоматически наследует поля, методы и свойства своего родителя и может дополнять их новыми. Таким образом, принцип наследования обеспечивает поэтапное создание сложных классов и разработку собственных библиотек классов.

Все классы Object Pascal порождены от единственного родителя класса `TObject`. Этот класс не имеет полей и свойств, но включает в себя методы самого общего назначения, обеспечивающие весь жизненный цикл любых объектов - от их создания до уничтожения. Программист не может создать класс, который не был бы дочерним классом `TObject`. Следующие два объявления идентичны:

```
TaClass = class(TObject) TaClass = class
```

Принцип наследования приводит к созданию ветвящегося дерева классов, постепенно разрастающегося при перемещении от `TObject` к его потомкам. Каждый потомок дополняет возможности своего родителя новыми и передает их своим потомкам.

Для примера на рисунке показан небольшой фрагмент дерева классов Delphi. Класс `TPersistent` обогащает возможности своего родителя `TObject`: он "умеет" сохранять данные в файле и получать их из него, в результате это умеют делать и все его потомки. Класс `TComponent`, в свою очередь, умеет взаимодействовать со средой разработчика и передает это умение своим потомкам. `TControl` не только способен работать с файлами и средой разработчика, но он еще умеет создавать и обслуживать видимые на экране изображения, а его потомок `TWinControl` может создавать Windows-окна и т. д.

#### **1.3 Полиморфизм**

Полиморфизм - это свойство классов решать схожие по смыслу проблемы разными способами. В рамках Object Pascal поведенческие свойства класса определяются набором входящих в него методов. Изменяя алгоритм того или иного метода в потомках класса, программист может придавать этим потомкам отсутствующие у родителя специфические свойства. Для изменения метода необходимо перекрыть его в потомке, т. е. объявить в потомке одноименный метод и реализовать в нем нужные действия. В результате в объекте-родителе и объекте-потомке будут действовать два одноименных метода, имеющих разную алгоритмическую основу и, следовательно, придающих объектам разные свойства. Это и называется полиморфизмом объектов.

## 2. СОСТАВЛЯЮЩИЕ КЛАССА

### 2.1 Поля

Полями называются инкапсулированные в классе данные. Поля могут быть любого типа, в том числе - классами, например:

```
type TMyClass = class  
  aIntField: Integer;  
  aStrField: String;  
  aObjField: TObject;  
end;
```

Каждый объект получает уникальный набор полей, но общий для всех объектов данного класса набор методов и свойств. Фундаментальный принцип инкапсуляции требует обращаться к полям только с помощью методов и свойств класса. Однако в Object Pascal разрешается обращаться к полям и напрямую:

```
Form2.Caption:= Ssender.ClassName;
```

Инкапсулированные в классе процедуры и функции называются методами. Они объявляются так же, как и обычные подпрограммы:

```
type  
TMyClass = class  
Function MyFunc(aPar: Integer): Integer;  
Procedure MyProc;  
end;
```

Доступ к методам класса, как и к его полям, возможен с помощью составных имен:

```
var  
aObject: TMyClass;  
begin  
aObject.MyProc;  
end;
```

Экземпляры разных классов могут выполнять сходную по названию процедуру, но, в общем случае, будут это делать по-разному..

### ОБЪЯВЛЕНИЕ КЛАССА

Любой вновь создаваемый класс может содержать секции (разделы), определяемые зарезервированными словами **published** (опубликованные), **private** (закрытые), **protected** (защищенные), **public** (доступные) и **automated** (автоматизированные). Внутри каждой секции вначале определяются поля, а затем - методы и свойства.

Секции определяют области видимости элементов описания класса. Секция **public** накладывает ограничений на область видимости перечисляемых в ней полей, методов и свойств - их можно вызывать в любом другом модуле программы. Секция **published** также не ограничивает область видимости, однако в ней перечисляются свойства, которые должны быть доступны не только на этапе исполнения, но и на этапе конструирования программы (т. е. в окне Инспектора объектов). Секция **published** используется только при разработке нестандартных компонентов. Замечу, что среда Delphi помещает описания компонентов, вставленных в форму, в специальную секцию без названия, которая располагается сразу за заголовком класса и продолжается до первой объявленной секции. Эта секция - **published**. Программисту не следует помещать в нее собственные элементы описания класса или удалять из нее элементы, вставленные средой. Секция **private** сужает область видимости до минимума: закрытые элементы описания доступны только внутри методов данного класса и подпрограммах, находящихся в том же модуле, где описан класс. Элемент, объявленный в секции **private**, становится недоступным даже ближайшим потомкам класса, если они размещаются в других модулях. Секция **protected** доступна только методам самого класса, а также любым его потомкам, независимо от того, находятся ли они в том же модуле или нет. Наконец, секция **automated** используется только для объявления свойств и методов, которые

будут добавлены к так называемому интерфейсу OLE-объектов Автоматизации; область видимости членов этой секции не ограничена.

В Object Pascal разрешается сколько угодно раз объявлять любую секцию, причем порядок следования секций не имеет значения. Любая секция может быть пустой.

## **Объект TForm**

### **Свойства**

ActiveControl - компонент формы, который имеет фокус ввода при ее запуске

Align - выравнивание

alLeft, alTop, alRight, alBottom - касается одной из сторон экрана (левой, верхней, правой, нижней), при этом растягивается по всей длине этой стороны

alClient - занимает весь экран

alNone - форма запускается в таком виде, как задано при проектировании

AutoScroll

true - автоматически показывает полосы прокрутки, если форма уменьшена, и не все ее компоненты видны

AutoSize

true - размер формы автоматически меняется по содержимому (даже на этапе проектирования)

BorderIcons - кнопки в заголовке (в системном меню) формы

biSystemMenu - наличие системного меню (кнопок в заголовке формы)

biMinimize - кнопка минимизации формы

biMaximize - кнопка максимизации формы

biHelp - если свойство BorderStyle имеет значение bsDialog, либо отсутствуют кнопки biMinimize или biMaximize, то в главном меню появляется кнопка помощи (знак вопроса)

BorderStyle - поведение рамки окна

bsDialog - стандартное окно; размер окна менять нельзя

bsSingle - стандартное окно; размер окна менять нельзя, но доступна минимизация и максимизация

bsSizeable - стандартное окно; размер окна можно менять

bsNone - заголовок и рамка окна отсутствуют

BorderWidth - ширина рамки окна (на рамке не могут находиться компоненты)

Caption - заголовок

ClientHeight - высота рабочей области окна (на которой могут быть компоненты). Не включает в себя ширину рамки

ClientWidth - ширина рабочей области окна (на которой могут быть компоненты). Не включает в себя ширину рамки

Color - цвет

Constraints - ограничители ширины и высоты окна, минимум и максимум. Если установлены нули, то эти свойства не используются

Cursor - тип курсора мыши в момент, когда он находится над окном

Enabled

true - окно отвечает на события от мыши, клавиатуры и таймера

Font - шрифт. Параметры шрифта устанавливаются через стандартное окно шрифтов, либо по строкам

FormStyle

fsNormal - стандартное окно

fsStayOnTop - окно всегда «наверху», перекрывает собой остальные окна.

Height - общая высота окна

Hint - текст всплывающей подсказки, когда курсор мыши находится над формой

HorzScrollBar - параметры горизонтальной полосы прокрутки

Icon - «иконка» в заголовке формы

Left - координата левого верхнего угла формы относительно экрана

Menu - главное меню окна (для работы с меню на форму нужно поместить компонент TMainMenu)

Name - внутреннее имя формы, имя переменной

Position - расположение формы

poDesigned - как при разработке

poDefault - размер и расположение определяется Windows

poDefaultPosOnly - Windows определяет только расположение

poDefaultSizeOnly - Windows определяет только размер

poScreenCenter, poDesktopCenter - по центру экрана

poMainFormCenter - по центру главной формы (если данная форма не является главной формой приложения), а для главной формы - по центру экрана

ShowHint

true - наличие всплывающей подсказки (текст задается в свойстве Hint)

Top - координата левого верхнего угла формы относительно экрана

VertzScrollBar - параметры вертикальной полосы прокрутки

Visible - видимость окна (для главной формы приложения не отключается)

Width - общая ширина окна

WindowState wsNormal, wsMinimized, wsMaximized

Сочетание свойств BorderStyle = bsNone, Align = alClient можно использовать для приложения с окном, постоянно развернутым на весь экран

## **Объект TForm**

### **События**

OnActivate - форма отрисована и готова к работе

OnClick - сделан щелчок мышью

OnClose - получена команда закрыть окно

OnCreate - форма создается в памяти

OnDblClick - сделан двойной щелчок мышью

OnDestroy - форма уничтожается в памяти

OnHide - (свойство Visible = false)

OnKeyDown - нажата клавиша (еще не отпущена)

OnKeyPress - нажата и отпущена клавиша

OnKeyUp - отпущена клавиша

OnMouseDown - нажата кнопка мыши (курсор находится над окном)

OnMouseMove - передвинут курсор мыши (курсор находится над окном)

OnMouseUp - отпущена кнопка мыши (курсор находится над окном)

OnPaint - команда перерисовки окна (от пользователя либо от операционной системы - например, при разворачивании окна)

OnResize - изменение размеров окна

OnShow - отрисовка окна (в начале работы)

Порядок возникновения событий при запуске формы:

OnCreate

OnShow

OnActivate

OnPaint

Порядок возникновения событий при закрытии формы:

OnClose

OnHide

OnDestroy

## **Объект TButton**

## Свойства

### anchors

true - привязка положения и размеров к родительскому компоненту. При изменении размеров родительского компонента расстояние до его краев неизменно. Если установлена привязка одновременно к противоположным сторонам родительского компонента, то данный компонент (кнопка) растягивается.

### cancel

true - при нажатии пользователем клавиши Escape выполняется обработчик события OnClick данной кнопки, если она имеет фокус ввода. Если есть несколько кнопок, у которых свойство Cancel=true, то выполняется обработчик события той кнопки, у которой значение свойства TabOrder меньше.

### caption

Constraints - ограничители ширины и высоты кнопки, минимум и максимум. Если установлены нули, то эти свойства не используются. Если установлены привязки к родительскому компоненту (anchors), и ограничения по размеру кнопки (constraints), то constraints приоритетнее (вся форма не может быть увеличена больше, чем позволяют ограничения по размеру кнопки).

cursor - тип курсора мыши в момент, когда он находится над окном

### default

true - при нажатии пользователем клавиши Enter выполняется обработчик события OnClick данной кнопки

### enabled

true - кнопка отвечает на события от мыши, клавиатуры и таймера

font - шрифт. Параметры шрифта устанавливаются через стандартное окно шрифтов, либо по строкам

height - высота кнопки

hint - текст всплывающей подсказки, когда курсор мыши находится над кнопкой

left - координата левого верхнего угла относительно родительского компонента

name - внутреннее имя кнопки, имя переменной

### parentfont

true - использовать тот же шрифт, что у родительского компонента

popupMenu - контекстное меню, появляющееся при нажатии правой кнопки мыши (когда курсор мыши находится над данной кнопкой). Для создания контекстного меню нужно поместить на форм компонент TPopupMenu.

### showHint

true - наличие всплывающей подсказки (текст задается в свойстве hint)

tabOrder - номер компонента в списке «Tab Order» родительского компонента. Этот список определяет, в каком порядке между компонентами передается фокус ввода при нажатии пользователем клавиши Tab

### tabStop

true - кнопка может получать фокус ввода при нажатии пользователем клавиши Tab.

top - координата левого верхнего угла кнопки относительно родительского компонента

visible - видимость кнопки

width - ширина кнопки

## События

OnClick - сделан щелчок левой кнопкой мыши

OnContextPopup - сделан щелчок правой кнопкой мыши (и вызвано контекстное меню, при его наличии)

OnEnter - получен фокус ввода

OnExit - потерял фокус (перешел к другому компоненту этой формы)

OnKeyDown - нажата клавиша (еще не отпущена)  
OnKeyPress - нажата и отпущена клавиша  
OnKeyUp - отпущена клавиша  
OnMouseDown - нажата кнопка мыши (курсор находится над окном)  
OnMouseMove - передвинут курсор мыши (курсор находится над окном)  
OnMouseUp - отпущена кнопка мыши (курсор находится над окном)

## Объект TLabel

### Свойства

Align - выравнивание

alLeft, alTop, alRight, alBottom - касается одной из сторон компонента-контейнера (левой, верхней, правой, нижней), при этом растягивается по всей длине этой стороны

Alignment - выравнивание текста в пределах метки

Anchors

true - привязка положения и размеров к родительскому компоненту. При изменении размеров родительского компонента расстояние до его краев неизменно. Если установлена привязка одновременно к противоположным сторонам родительского компонента, то данный компонент (метка) растягивается.

AutoSize - изменение размеров метки в соответствии с расположенным на ней текстом

true - при нажатии пользователем клавиши Esc выполняется обработчик события OnClick данной кнопки, если она имеет фокус ввода. Если есть несколько кнопок, у которых свойство Cancel==true, то выполняется обработчик события той кнопки, у которой значение свойства TabOrder меньше.

Caption - текст метки

Color - цвет

Constraints - ограничители ширины и высоты кнопки, минимум и максимум. Если установлены нули, то эти свойства не используются.

Cursor - тип курсора мыши в момент, когда он находится над меткой

Enabled

true - обычный вид

false - серый цвет

FocusControl - указание оконного компонента (способного получать фокус - кнопка, поле ввода), которое получает фокус ввода при нажатии пользователем горячей клавиши, отмеченной знаком & в поле Caption метки. Например, если Caption == "&Exit", а FocusControl == Button2, то при нажатии клавиши E фокус ввода получит Button2.

Font - шрифт. Параметры шрифта устанавливаются через стандартное окно шрифтов, либо поэлементно

Height - высота метки

Hint - текст всплывающей подсказки, когда курсор мыши находится над меткой

Layout - вертикальное выравнивание текста на метке

Left - координата левого верхнего угла относительно родительского компонента

Name - внутреннее имя метки, имя переменной

ParentFont

true - использовать тот же шрифт, что у родительского компонента

ParentShowHint -

true - использовать то же значение свойства Hint, что и родительский компонент

PopupMenu - контекстное меню, появляющееся при нажатии правой кнопки мыши (когда курсор мыши находится над данной кнопкой). Для создания контекстного меню нужно поместить на форм компонент TPopupMenu.

ShowAccelChar - способ отображения символа &

true - символ & не отображается; следующий за ним символ отображается подчеркнутым и может использоваться в качестве горячей клавиши (кроме пробела и самого символа &)

false - символ & отображается как есть; горячая клавиша для метки не используется  
ShowHint

true - наличие всплывающей подсказки (текст задается в свойстве Hint)

TabOrder - номер компонента в списке «Tab Order» родительского компонента. Этот список определяет, в каком порядке между компонентами передается фокус ввода при нажатии пользователем клавиши Tab

Top - координата левого верхнего угла кнопки относительно родительского компонента

Transparent - прозрачность метки

Visible - видимость метки

Width - ширина метки

WordWrap -

true - возможен перенос слов на следующую строку (если ширина текста превышает ширину метки)

## **События**

OnClick - сделан щелчок левой кнопкой мыши

OnContextPopup - сделан щелчок правой кнопкой мыши (и вызвано контекстное меню, при его наличии)

OnDbClick - сделан двойной щелчок левой кнопкой мыши

OnMouseDown - нажата кнопка мыши (курсор находится над окном)

OnMouseMove - передвинут курсор мыши (курсор находится над окном)

OnMouseUp - отпущена кнопка мыши (курсор находится над окном)

## **Объект TCheckBox**

### **Свойства**

Alignment - выравнивание текста

Anchors

true - привязка положения и размеров к родительскому компоненту. При изменении размеров родительского компонента расстояние до его краев неизменно. Если установлена привязка одновременно к противоположным сторонам родительского компонента, то данный компонент (метка) растягивается.

AutoSize - изменение размеров метки в соответствии с расположенным на ней текстом

true - при нажатии пользователем клавиши Escape выполняется обработчик события OnClick данной кнопки, если она имеет фокус ввода. Если есть несколько кнопок, у которых свойство Cancel==true, то выполняется обработчик события той кнопки, у которой значение свойства TabOrder меньше.

Caption - текст

Checked

true - если поле отмечено

Color - цвет

Constraints - ограничители ширины и высоты кнопки, минимум и максимум. Если установлены нули, то эти свойства не используются.

Cursor - тип курсора мыши в момент, когда он находится над меткой

Enabled

true - обычный вид

false - серый цвет

Font - шрифт

Height - высота текстового поля

Hint - текст всплывающей подсказки, когда курсор мыши находится над меткой

Left - координата левого верхнего угла относительно родительского компонента

Name - внутреннее имя

PopupMenu - контекстное меню, появляющееся при нажатии правой кнопки мыши (когда курсор мыши находится над данной кнопкой). Для создания контекстного меню нужно поместить на форм компонент TPopupMenu.

ShowHint

true - наличие всплывающей подсказки (текст задается в свойстве Hint)

State -

cbChecked - поле отмечено

cbGrayed - поле отмечено, но галка серого цвета

cbUnchecked - поле не отмечено

TabOrder - номер компонента в списке «Tab Order» родительского компонента. Этот список определяет, в каком порядке между компонентами передается фокус ввода при нажатии пользователем клавиши Tab

TabStop

true - может получать фокус ввода при нажатии пользователем клавиши tab

Top - координата левого верхнего угла кнопки относительно родительского компонента

Visible - видимость метки

Width - ширина метки

## События

OnClick - сделан щелчок левой кнопкой мыши

OnContextPopup - сделан щелчок правой кнопкой мыши (и вызвано контекстное меню, при его наличии)

OnEnter - получен фокус ввода

OnExit - потерял фокус (перешел к другому компоненту этой формы)

OnKeyDown - нажата клавиша (еще не отпущена)

OnKeyPress - нажата и отпущена клавиша

OnKeyUp - отпущена клавиша

OnMouseDown - нажата кнопка мыши (курсor находится над окном)

OnMouseMove - передвинут курсор мыши (курсor находится над окном)

OnMouseUp - отпущена кнопка мыши (курсor находится над окном)

## TMainMenu

Поместите на форму компонент TMainMenu. Это невидимый компонент, поэтому его расположение на форме при проектировании не имеет значения. Сделайте двойной щелчок на компоненте TMainMenu. Должно открыться окно редактирования меню.

Слева в Инспекторе объектов впишите название первого пункта меню &File в поле Caption.

Знак & перед буквой означает, что такая клавиша может использоваться для быстрого доступа к меню (Alt-F). В приложении эта буква будет подчеркнутой. Автоматически будет создан объект с именем File1::TMenuItem, связанный с этим пунктом меню. Если название пункта меню вписано на русском языке, то имя объекта не будет совпадать с названием пункта, а будет предложено имя N1 (его можно сменить).

Один раз щелкните на слове File в окне редактирования меню. Ниже появится пустой пункт меню. Один раз щелкните на пустом пункте, впишите &Exit в поле Caption.

Создайте обработчик события TForm1::Exit1Click. Для этого:

1) дважды щелкните на пункте меню &Exit, или



2) убедитесь, что объект Exit1 активен в инспекторе объектов, перейдите в нем на вкладку Events и сделайте двойной щелчок справа в строке OnClick.

В окне редактирования кода должен создаться обработчик события Exit1Click. Отредактируйте тело функции, вписав команду закрытия окна формы. Должно получиться следующее:

```
//-----  
void __fastcall TForm1::Exit1Click(TObject *Sender)  
{  
Form1->Close();  
}
```

Добавлять и удалять строки в меню можно с помощью правой кнопки мыши или клавиатуры (Insert, Delete).

Для вставки разделительной строки между пунктами создайте пустой пункт, напишите минус (дефис) в строке Caption и нажмите Enter.

### **Свойства TMainMenu**

#### **AutoHotKeys**

maAutomatic - при конфликте горячих клавиш (несколько пунктов меню с одинаковой буквой) C++Builder сам определяет, какой из пунктов будет выбран при нажатии горячей клавиши

maManual - при конфликте горячих клавиш все конфликтующие пункты меню будут доступны (для доступа к первому - нажатие Alt и однократное нажатие горячей клавиши, ко второму - Alt и двукратное нажатие горячей клавиши и т.д.)

#### **AutoLineReduction** -

true - автоматическое удаление разделительных строк в начале, конце меню

Images - компонент TImageList, содержащий bitmap-изображения для вывода возле строк меню. Выбор картинки для пунктов осуществляется для каждого пункта отдельно, в его свойстве ImageIndex

Items - ссылка на подпункты

### **Событие TMainMenu**

OnChange - вызывается при программном изменении или создании в памяти объекта меню

## **УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС**

**к дополнительной общеобразовательной общеразвивающей программе  
«Программирование на языке PascalABC и Lazarus »**

<b>Направленность</b>	Техническая
<b>Продолжительно</b>	1 год

сть освоения				
Возраст детей	12-15 лет			
Нормативное обеспечение	<p>Образовательная программа Рабочая программа План воспитательной работы (план мероприятий) Инструкции по технике безопасности Нормативная документация:</p> <ul style="list-style-type: none"> <li>• <a href="#">Федеральный закон Российской Федерации №273-ФЗ "Об образовании в Российской Федерации" от 29.12.2012</a></li> <li>• <a href="#">Концепция развития дополнительного образования детей в Российской Федерации</a> Распоряжение Правительства РФ от 04.09.2014 №1726-р</li> <li>• <a href="#">Стратегия развития системы образования Санкт-Петербурга на 2011–2020 гг. «Петербургская Школа 2020»</a> // Совет по образовательной политике Комитета по образованию Правительства Санкт-Петербурга, 2010</li> <li>• <a href="#">Стратегия развития воспитания в Российской Федерации на период до 2025 года</a> // Распоряжение Правительства РФ от 29.05.2015 №996-р</li> <li>• <a href="#">Об утверждении СанПиН 2.4.4.3172-14 "Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательной организации дополнительного образования детей"</a> // Постановление Главного санитарного врача РФ от 04.07.2014 №41</li> <li>• <a href="#">Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам</a> // Приказ Министерства образования и науки РФ от 29.08.2013 г. №1008</li> <li>• Об утверждении Методических рекомендаций по проектированию дополнительных общеразвивающих программ в государственных образовательных организациях Санкт-Петербурга, находящихся в ведении Комитета по образованию от 01.03.2017 г. №617-Р</li> </ul>			
Разделы /темы дополнительной общеобразовательной программы	Учебно-методические пособия для педагогов	Учебно-методические пособия для детей	Диагностические и контрольные материалы	Средства обучения
Тема 2	- электронный сборник программ:	- раздаточные материалы;-пособие «Основы программирования Паскаль. Сложные структуры данных и алгоритмы. 2 год обучения»	- критерии оценки работы учащихся в течение учебного года; - таблица фиксации образовательных результатов	- <i>персональные компьютеры;</i> - <i>интерактивная доска;</i>
Тема 3	- электронный сборник программ:	- раздаточные материалы;-пособие «Основы программирования Паскаль. Сложные структуры данных и алгоритмы. 2 год обучения»	- критерии оценки работы учащихся в течение учебного года; - таблица фиксации образовательных результатов .	- <i>персональные компьютеры;</i> - <i>интерактивная доска;</i>
Тема 4	- электронный	- раздаточные	- критерии	- <i>персональные</i>

	сборник программ:	материалы;-пособие «Основы программирования Паскаль. Сложные структуры данных и алгоритмы. 2 год обучения»	оценки работы учащихся в течение учебного года; - таблица фиксации образовательных результатов	<i>компьютеры;</i> - <i>интерактивная доска;</i>
Тема 5	- электронный сборник программ:	- раздаточные материалы;-пособие «Основы программирования Паскаль. Сложные структуры данных и алгоритмы. 2 год обучения»	- критерии оценки работы учащихся в течение учебного года; - таблица фиксации образовательных результатов обучающихся.	- <i>персональные компьютеры;</i> - <i>интерактивная доска;</i>
Тема 6	- электронный сборник программ:	- раздаточные материалы;-пособие «Основы программирования Паскаль. Сложные структуры данных и алгоритмы. 2 год обучения»	- критерии оценки работы учащихся в течение учебного года; - таблица фиксации образовательных результатов	- <i>персональные компьютеры;</i> - <i>интерактивная доска;</i>
Тема 7	- электронный сборник программ:	- раздаточные материалы;-пособие «Основы программирования Паскаль. Сложные структуры данных и алгоритмы. 2 год обучения»	- критерии оценки работы учащихся в течение учебного года; - таблица фиксации образовательных	- <i>персональные компьютеры;</i> - <i>интерактивная доска;</i>
Тема 8		- раздаточные материалы;	- критерии оценки работы учащихся в течение учебного года; - таблица фиксации образовательных результатов	- персональные компьютеры; - интерактивная доска; - раздаточные материалы; -
Тема 9	- электронный сборник программ:- литература : 1. Фаронов В.В. « <a href="#">Delphi. Программирование на языке высокого</a>	- раздаточные материалы;- литература : 1. Фаронов В.В. «Delphi. Программирование на языке высокого уровня»	- критерии оценки работы учащихся в течение учебного года; - таблица фиксации образовательных результатов	- персональные компьютеры; - интерактивная доска; - раздаточные материалы; -

	<u>уровня»</u>			
Тема 10	- электронный сборник программ:- литература : 1. А.Чиртик «Программирование в <a href="#">Delphi</a> » .	- раздаточные материалы; - литература : 1. Сухарев М. «Золотая 1. Фаронов В.В. «Delphi. Программирование на языке высокого уровня»книга Delphi»	- критерии оценки работы учащихся в течение учебного года; - таблица фиксации образовательных результатов	- персональные компьютеры; - интерактивная доска;
Тема 11	- электронный сборник программ:- литература : 1. Сухарев М. «Золотая книга Delphi» .	- раздаточные материалы; - литература : 1. Сухарев М. «Золотая книга Delphi»	- критерии оценки работы учащихся в течение учебного года; - таблица фиксации образовательных результатов	- персональные компьютеры; - интерактивная доска; - электронные учебники;
Тема 12	- электронный сборник программ:- литература : 1. Сухарев М. «Золотая книга Delphi» .	- раздаточные материалы; - литература : 1. Сухарев М. «Золотая книга Delphi»	- критерии оценки работы учащихся в течение учебного года; - таблица фиксации образовательных результатов	- персональные компьютеры; - интерактивная доска;
Тема 13	- электронный сборник программ:- литература : 1. Сухарев М. «Золотая книга Delphi» .	- раздаточные материалы; - литература : 1. Фаронов В.В. « <a href="#">Delphi. Программирование на языке высокого уровня</a> » Электронные учебники; 1. <a href="http://intuit.valrkl.ru/course-1265/index.html">http://intuit.valrkl.ru/course-1265/index.html</a> («Программирование на Lazarus» Ачкасов В.Ю.)	- критерии оценки работы учащихся в течение учебного года; - таблица фиксации образовательных результатов	- персональные компьютеры; - интерактивная доска;
Тема 14	- электронный	- раздаточные	- критерии	- персональные

	<p>сборник программ:- литература :</p> <p>1. Фаронов В.В. «<a href="#">Delphi. Программирование на языке высокого уровня</a>»</p>	<p>материалы;- литература :</p> <p>1. Фаронов В.В. «Delphi. Программирование на языке высокого уровня»</p> <p>Электронные учебники;</p> <p>1. <a href="http://intuit.valrkl.ru/course-1265/index.html">http://intuit.valrkl.ru/course-1265/index.html</a> («Программирование на Lazarus» Ачкасов В.Ю.)</p>	<p>оценки работы учащихся в течение учебного года;</p> <p>- таблица фиксации образовательных результатов</p>	<p>компьютеры;</p> <p>- интерактивная доска;</p>
Тема 15	<p>- электронный сборник программ:- литература :</p> <p>Сайт <a href="#">Lazarus Documentation.ru</a> <a href="#">высокого уровня</a>»</p>	<p>Сайт Lazarus Documentation.ru</p>	<p>- критерии оценки работы учащихся в течение учебного года;</p> <p>- таблица фиксации образовательных результатов</p>	<p>- персональные компьютеры;</p> <p>- интерактивная доска;</p>
Тема 16	<p>- электронный сборник программ:- литература :</p> <p>Сайт <a href="#">Lazarus Documentation.ru</a> <a href="#">высокого уровня</a>»</p>	<p>Сайт Lazarus Documentation.ru</p>	<p>- критерии оценки работы учащихся в течение учебного года;</p> <p>- таблица фиксации образовательных результатов</p>	<p>- персональные компьютеры;</p> <p>- интерактивная доска;</p>
Тема 17	<p>- электронный сборник программ:</p> <p>- литература :</p> <p>1. Фаронов В.В. «Delphi. Программирование на языке высокого уровня»</p> <p>2. Сухарев М. «Золотая книга Delphi»</p> <p>3. Фленов М. «Библия Delphi»</p> <p>4. Хомоненко А., Гофман В.,</p>	<p>- раздаточные материалы;</p> <p>- литература :</p> <p>1. Фаронов В.В. «Delphi. Программирование на языке высокого уровня»</p> <p>2. Сухарев М. «Золотая книга Delphi»</p> <p>3. Бобровский С. «Delphi 7. Учебный курс»</p> <p>электронные учебники;</p> <p>1. <a href="http://intuit.valrkl.ru/">http://intuit.valrkl.ru/</a></p>	<p>- критерии оценки работы учащихся в течение учебного года;</p> <p>- таблица фиксации образовательных результатов обучающихся.</p>	<p>- персональные компьютеры;</p> <p>- интерактивная доска;</p> <p>- электронные учебники;</p> <p>- адреса сайтов в Интернете: 1. <a href="http://intuit.valrkl.ru/course-1265/index.html">http://intuit.valrkl.ru/course-1265/index.html</a> («Программирование на Lazarus» Ачкасов В.Ю.)</p> <p>2. <a href="http://www.freepascal.ru/download/pdf/osnovy_programmirovaniya_v_sre">http://www.freepascal.ru/download/pdf/osnovy_programmirovaniya_v_sre</a></p>

<p>Мещеряков В., Никифоров В. «Delphi 7. Наиболее полное руководство» 5. Чиртик А. <a href="http://intuit.valrkl.ru/course-1265/index.html">http://intuit.valrkl.ru/course-1265/index.html</a> «Программирование в Delphi» 6. Бобровский С. «Delphi 7. Учебный курс» - электронные учебники; 1. («Программирование на Lazarus» Ачкасов В.Ю.) 2. <a href="http://www.free-pascal.ru/download/pdf/osnovy_programmirovaniya_v_srede_lazarus.pdf">http://www.free-pascal.ru/download/pdf/osnovy_programmirovaniya_v_srede_lazarus.pdf</a></p> <p>( «Основы программирования в среде Lazarus». Мансуров К.Т.) Сайт <a href="http://LazarusDocumentation.ru">Lazarus Documentation.ru</a></p>	<p><a href="http://course-1265/index.html">course-1265/index.html</a> («Программирование на Lazarus» Ачкасов В.Ю. ) 2. <a href="http://teacher.ucoz.net/Lecture/Pascal/Lazarus.pdf">http://teacher.ucoz.net/Lecture/Pascal/Lazarus.pdf</a></p> <p>(«Самоучитель по программированию на Free Pascal и Lazarus». Алексеев Е.Р., Чеснокова О.В., Кучер Т.В.) Сайт Lazarus Documentation.ru</p>	<p><a href="http://de_lazarus.pdf">de_lazarus.pdf</a></p> <p>( «Основы программирования в среде Lazarus». Мансуров К.Т.) 3. <a href="http://teacher.ucoz.net/Lecture/Pascal/Lazarus.pdf">http://teacher.ucoz.net/Lecture/Pascal/Lazarus.pdf</a></p> <p>(«Самоучитель по программированию на Free Pascal и Lazarus». Алексеев Е.Р., Чеснокова О.В., Кучер Т.В.)</p>
---	---	--

## ИНФОРМАЦИОННЫЕ ИСТОЧНИКИ

Список литературы для педагога:

1. Бобровский С. «Delphi 7. Учебный курс» 2008 г.
2. Гусева А.И. Учимся программировать Pascal 7.0 .- Москва: «Диалог МИФИ»,1998 г.
3. Кетков Ю.Л., Кетков А.Ю. Свободное программное обеспечение. Free Pascal для студентов и школьников. БХВ-Петербург. 2011г.
4. Меньшиков Ф. Олимпиадные задачи по программированию., ПИТЕР, 2006
5. Мозговой М. Занимательное программирование. СПб, ПИТЕР 2004 г.
6. Сухарев М TURBO PASCAL 7.0 Теория и практики программирования.. Наука и техника СПб 2007 г.
7. Потопахин В. Решение сложных задач БХВ-Петербург, 2006 г
8. Рубанцев В Занимательные уроки с Паскалем, или PascalABC.NET для начинающих.
9. . Я+R. 2012г.
10. Суханов М. «Золотая книга Delphi».Наука и техника. 2008 г.
11. Фаронов В.В. «Delphi. Программирование на языке высокого уровня».Питер. 2009г.
12. Фаронов Н.В. TURBO PASCAL. Учебный курс СПб ПИТЕР 2002 г.
13. Фленов М. «Библия Delphi». БХВ-Петербург. 2011г.
14. Хомоненко А., Гофман В., Мещеряков В., Никифоров В. «Delphi 7/ Наиболее полное руководство»
15. Чиртик А. «Программирование в Delphi” Питер 2010г.
16. Шень А. Программирование. Теоремы и задачи. МЦНМО, Москва,2004 г.

Интернет-источники:

1. Электронный учебник «Программирование на Lazarus» Ачкасов В.Ю., <http://intuit.valrkl.ru/course-1265/index.html>
2. Электронный учебник «Самоучитель по программированию на Free Pascal и Lazarus». Алексеев Е.Р., Чеснокова О.В., Кучер Т.В., <http://teacher.ucoz.net/>
3. [Lecture/Pascal/Lazarus.pdf](#)
4. [course-1265/index.html](#) ( «Программирование на Lazarus» Ачкасов В.Ю.
5. Сайт Lazarus Documentation.ru