

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

Заведующий кафедрой
Ядерной физики
Кадменский С.Г.



30.06.2021 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Б1. В.ДВ.02.02 ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

1. Код и наименование направления подготовки/специальности: 03.03.02 Физика
 2. Профиль подготовки/специализация: Физика твердого тела
 3. Квалификация выпускника: бакалавр
 4. Форма обучения: очная
 5. Кафедра, отвечающая за реализацию дисциплины: кафедра ядерной физики
 6. Составители программы: ассистент, Работкин Владимир Александрович
-

7. **Рекомендована:** Научно-методическим советом физического факультета, протокол № 6 от 17.06.2021
РП продлена на 2022-2023 учебный год НМС физического факультета 14.06.2022, протокол №6.

отметки о продлении вносятся вручную)

8. Учебный год: 2021-2022

Семестр(ы): 2

9. Цели и задачи учебной дисциплины

Целями освоения учебной дисциплины являются:

- приобретение базовых знаний и навыков в области практики классического программирования.
- знакомство с основными принципами и подходами объектно-ориентированного программирования.
- формирование культуры разработки программных продуктов.

- обучение работе с научно-технической литературой и технической документацией по программному обеспечению ПЭВМ.
Задачи учебной дисциплины:
- получить практические навыки работы с современными визуальными средами программирования и навыки проектирования программ со сложным графическим интерфейсом.

10. Место учебной дисциплины в структуре ООП: *(обязательная часть или часть, формируемая участниками образовательных отношений (вариативная) блока Б1, к которой относится дисциплина; требования к входным знаниям, умениям и навыкам; дисциплины, для которых данная дисциплина является предшествующей)*

Дисциплина вариативной части базового цикла основной образовательной программы подготовки бакалавров направления 03.03.02 Физика.

Дисциплина базируется на курсах дисциплин, изучаемых в образовательных программах бакалавриата: «Математика», «Физика». Для освоения дисциплины необходимы знания, умения и компетенции, полученные при изучении дисциплины «Программирование» основной образовательной программы подготовки бакалавров по направления 03.03.02 Физика.

11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями) и индикаторами их достижения:

Код	Название компетенции	Код(ы)	Индикатор(ы)	Планируемые результаты обучения
ПК-3	Подготовка исходных данных, наладка экспериментальных стендов и установок для обеспечения выполнения научных исследований	ПК-3.1	Подготовка исходных данных для используемых программных кодов моделирования физических процессов в экспериментальных стендах и установках	<p>Знать: методологию объектно-ориентированного проектирования и программирования; основные средства стандартной библиотеки классов VCL Delphi и методы разработки пользовательских классов</p> <p>Уметь: методологию объектно-ориентированного проектирования и программирования; основные средства стандартной библиотеки классов VCL Delphi и методы разработки пользовательских классов</p> <p>Владеть: разработки программного обеспечения с использованием методов и технологий объектно-ориентированного проектирования и программирования</p>
		ПК-3.2	Проведение тестовых расчетов и поверочных измерений на установках и стендах	
		ПК-3.3	Применять современные математические и графические методы обработки расчетных и экспериментальных результатов	

12. Объем дисциплины в зачетных единицах/час. *(в соответствии с учебным планом)* — 3/108.

Форма промежуточной аттестации экзамен

13. Трудоемкость по видам учебной работы

Вид учебной работы	Трудоемкость
--------------------	--------------

	Всего	По семестрам		
		№ 2	№ семестра	...
Аудиторные занятия	68	68		
в том числе:	лекции	16	16	
	практические			
	лабораторные	16	16	
Самостоятельная работа	40	40		
Форма промежуточной аттестации <i>экзамен</i>	36	36		
Итого:	108	108		

13.1. Содержание дисциплины

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК *
1. Лекции			
1.1	Базовые принципы объектно-ориентированного программирования	Абстрагирование. Иерархия (наследование и включение). Полиморфизм. Инкапсуляция. Дополнительные принципы.	
1.2	Основы синтаксиса	Объявление класса, директивы видимости. Поля и методы различных типов. Разработка простой иерархии классов	
	Свойства	Объявление свойств. Методы записи и чтения. Индексные методы доступа. Объявления нового типа. Объявление и реализация свойств пользовательских типов.	
	Методы и события	Процедуры и функции как методы класса. Объявление и диспетчеризация события. Реализация события пользовательского типа. Обработка сообщений операционной системы. Объявление и реализации методов и событий пользовательских типов. Регистрация компонента.	
	Базовые компонентные классы и часто используемые классы.	Иерархия базовых классов VCL. TComponent. TCustomControl. TWinControl. TCustomControl Диалоги. TIniFile и TRegIniFile. TStringList. Другие Persistent классы	
	Графические элементы управления	Базовый класс графических элементов управления TGraphicControl. Использование ресурсов	
	Невизуальные компоненты и другие классы VCL.	Таймер. Компоненты для работы с ini-файлами и реестром. Организация многопоточных приложений. Объекты синхронизации: события, мьютексы, семафоры и критические секции. Стандартные диалоговые компоненты и функции	
	Интерфейсы и основы COM технологии	Интерфейсы и основы COM – технологии.	
2. Лабораторные занятия			
2.1	Базовые принципы объектно-ориентированного программирования	Проектирование простой иерархии классов для хранения и обработки данных.	
2.2	Основы синтаксиса	Реализация простой иерархии классов для хранения и обработки данных.	

2.3	Свойства	Реализация свойств различного типа.	
2.4	Методы и события	Реализация свойств, методов и событий пользовательских типов. Наследование существующего компонента. Реализация свойств, методов и событий пользовательских типов. Регистрация и тестирование компонента во время проектирования.	
2.5	Базовые компонентные классы и часто используемые классы.	Расширение существующего компонента. Наследование существующего компонента, перекрытие методов, добавление свойств, методов и событий.	
2.6	Графические элементы управления	Разработка графического элемента управления	
2.7	Невизуальные компоненты и другие классы VCL.	Разработка многопоточного приложения	
2.8	Интерфейсы и основы СОМ технологии	Сборка и тестирование всей библиотеки компонентов	

* *заполняется, если отдельные разделы дисциплины изучаются с помощью онлайн-курса. В колонке Примечание необходимо указать название онлайн-курса или ЭУМК. В других случаях в ячейки ставятся прочерки.*

13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела) дисциплины	Виды занятий (количество часов)				Всего
		Лекции	Практические	Лабораторные	Самостоятельная работа	
1	Базовые принципы объектно-ориентированного программирования	2		2	4	8
2	Основы синтаксиса	2		2	4	8
3	Свойства	2		2	4	8
4	Методы и события	2		2	4	8
5	Базовые компонентные классы и часто используемые классы.	2		2	5	9
6	Графические элементы управления	2		2	5	9
7	Невизуальные компоненты и другие классы VCL.	2		2	5	9
8	Интерфейсы и основы СОМ технологии	2		2	5	9
	Итого:	16		16	36	68

14. Методические указания для обучающихся по освоению дисциплины

(рекомендации обучающимся по освоению дисциплины: указание наиболее сложных разделов, работа с конспектами лекций, презентационным материалом, рекомендации по выполнению курсовой работы, по организации самостоятельной работы по дисциплине и др)

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины *(список литературы оформляется в соответствии с требованиями ГОСТ и используется общая сквозная нумерация для всех видов источников)*

а) основная литература:

№ п/п	Источник
1	Бабушкина И.А. Практикум по объектно-ориентированному программированию / И.А. Бабушкина, С.М. Окулов. – Издательство: "БИНОМ. Лаборатория знаний", ISBN: 978-5-9963-0954-2, 2012, 366 с // Издательство «Лань»: электронно-библиотечная система. – URL: https://e.lanbook.com/book/8781#book_name .
2	Лафоре, Р. Объектно-ориентированное программирование в С++ = Object-Oriented Programming in C++ / Р. Лафоре; [пер. с англ. А. Кузнецова, М. Назарова, В. Шрага]. — 4-е изд. — Санкт-Петербург : Питер, 2015. — 923 с.

б) дополнительная литература:

№ п/п	Источник
3	Скрипченко, Ю.С. Объектно-ориентированное программирование в примерах и задачах: учебное пособие / Ю.С. Скрипченко, Н.А. Тюкачев, В.Г. Хлебостроев. — Воронеж : Воронеж. гос. ун-т, 2006. — 160 с.
4	Дарахвелидзе П. Программирование в Delphi 7 / П. Дарахвелидзе, Е. Марков. — СПб.: БХВ-Петербург, 2005.— 781 с.
5	Введение в DELPHI: Методические указания к спец. курсу "Разработка Windows-приложений" для студ. 3 к. д/о и 4 к. в/о фак. ПММ / Воронеж. гос. ун-т. Каф. техн. кибернетики и автомат. регулирования; Сост. В. Г. Рудалев, А. И. Кремер. — Воронеж, 2000.— 36 с. <URL: http://www.lib.vsu.ru/elib/texts/method/vsu/m088.pdf >.

в) информационные электронно-образовательные ресурсы (официальные ресурсы интернет) *:

№ п/п	Ресурс
1.	www.lib.vsu.ru Источник
2.	http://www.delphimaster.ru/
3.	http://delphiworld.narod.ru/
4.	Новиков П.В. Объектно-ориентированное программирование [Электронный ресурс] : учебно-методическое пособие к лабораторным работам / П.В. Новиков. — Электрон. текстовые данные. — Саратов: Вузовское образование, 2017. — 124 с. — 978-5-4487-0011-8. — Режим доступа: http://www.iprbookshop.ru/64650.html

* Вначале указываются ЭБС, с которыми имеются договора у ВГУ, затем открытые электронно-образовательные ресурсы, онлайн-курсы, ЭУМК

Пример:

1. 3D-modeling / Трехмерное моделирование / Ф.П. Перепелица, В.А. Коснырев. — Открытое образование. — Режим доступа: <https://openedu.ru/>

16. Перечень учебно-методического обеспечения для самостоятельной работы (учебно-методические рекомендации, пособия, задачки, методические указания по выполнению практических (контрольных), курсовых работ и др.)

№ п/п	Источник
12	Объектно-ориентированное программирование [Электронный ресурс] : учебно-методическая разработка для вузов : [для студ. 2 к. д/о фак. приклад. математики, информатики и механики; для направлений 010503 - Мат. обеспечение и администрирование информ. систем, профиля 010500.62 - Теоретические основы информатики] / Воронеж. гос. ун-т ; сост.: Е.Е. Михайлова, Г.Э. Вошинская, К.С. Рыбак. — Электрон. текстовые дан. — Воронеж : Издательско-полиграфический центр Воронежского государственного университета, 2013. — Загл. с титул. экрана. — Свободный доступ из интранета ВГУ. — Текстовый файл. — Windows 2000 ; Adobe Acrobat Reader. — <URL: http://www.lib.vsu.ru/elib/texts/method/vsu/m13-172.pdf >
13	Николаев Е.И. Объектно-ориентированное программирование [Электронный ресурс]: учебное пособие/ Николаев Е.И.— Электрон. текстовые данные. — Ставрополь: Северо-Кавказский федеральный университет, 2015.— 225 с.— Режим доступа: http://www.iprbookshop.ru/62967.html . — ЭБС «IPRbooks»
14	Николаев Е.И. Объектно-ориентированное программирование. Часть 1 [Электронный ресурс]: лабораторный практикум/ Николаев Е.И.— Электрон. текстовые данные. — Ставрополь: Северо-Кавказский федеральный университет, 2015.— 183 с.— Режим доступа: http://www.iprbookshop.ru/62966.html . — ЭБС «IPRbooks»
15	Николаев Е.И. Объектно-ориентированное программирование. Часть 2 [Электронный ресурс]: лабораторный практикум/ Николаев Е.И.— Электрон. текстовые данные. — Ставрополь: Северо-Кавказский федеральный университет, 2015.— 156 с.— Режим доступа: http://www.iprbookshop.ru/63218.html . — ЭБС «IPRbooks»

17. Образовательные технологии, используемые при реализации учебной дисциплины, включая дистанционные образовательные технологии (ДОТ), электронное обучение (ЭО), смешанное обучение):

При реализации дисциплины могут проводиться различные типы лекций (вводная, обзорная и т.д.), семинарские занятия (проблемные, дискуссионные и т.д.), применяться дистанционные образовательные технологии в части освоения лекционного материала, проведения текущей аттестации, самостоятельной работы по дисциплине или отдельным ее разделам и т.д. При применении ЭО и ДОТ необходимо в п.15 в) указать используемые ресурсы (см. пример выше)

18. Материально-техническое обеспечение дисциплины: *(при использовании лабораторного оборудования указывать полный перечень, при большом количестве оборудования можно вынести данный раздел в приложение к рабочей программе)*

Компьютерный класс - 15 комп. III поколения, объединенных в сеть с выходом в Интернет. Сервер, рабочая станция.

Программное обеспечение: Turbo Delphi или Lazarus (свободно распространяемая среда разработки).

19. Оценочные средства для проведения текущей и промежуточной аттестаций

Порядок оценки освоения обучающимися учебного материала определяется содержанием следующих разделов дисциплины:

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
1	1-8	ПК-3	ПК-3.1	Вопросы КИМ
			ПК-3.2	
			ПК-3.3	
Промежуточная аттестация форма контроля - экзамен				КИМ

20 Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

20.1 Текущий контроль успеваемости

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств:

Описание технологии проведения

Требования к выполнению заданий (или шкалы и критерии оценивания)

Оценка знаний, умений и навыков, характеризующая этапы формирования компетенций в рамках изучения дисциплины осуществляется в ходе, текущей и промежуточной аттестаций.

Текущая аттестация проводится в соответствии с Положением о текущей аттестации обучающихся по программам высшего образования Воронежского государственного университета. Текущая аттестация проводится в формах: *устного опроса (индивидуальный опрос, фронтальная беседа); выполнение практико-ориентированных заданий, лабораторные работы, тестирования;*

20.2 Промежуточная аттестация

Описание технологии проведения

Промежуточная аттестация по дисциплине осуществляется с помощью следующих оценочных средств:

Для оценивания результатов обучения на экзамене/зачете используются следующие показатели (ЗУНЫ из 19.1):

(как пример):

- 1) знание учебного материала и владение понятийным аппаратом дисциплины;
- 2) умение связывать теорию с практикой;
- 3) умение иллюстрировать ответ примерами,
- 4) умение проектировать программы со сложным графическим интерфейсом.
- 5) владение способами современными методами обработки, анализа и синтеза информации, навыками работы с современными визуальными средами программирования.

Для оценивания результатов обучения на зачете используется – количественная шкала (зачет с оценкой)

Соотношение показателей, критериев и шкалы оценивания результатов обучения.

Критерии оценивания компетенций	Уровень сформиров	Шкала оценок
---------------------------------	-------------------	--------------

	анности компетенци й	
Полный ответ на вопросы билета и дополнительные вопросы, оформленная и выполненная лабораторная работа.	Повышенный уровень	Отлично
Для полного ответа требуются наводящие вопросы. Неполные ответы на дополнительные вопросы. Оформленная и выполненная лабораторная работа с незначительными ошибками.	Базовый уровень	Хорошо
Неполный ответ на вопросы. Неполные ответы на дополнительные вопросы. Общий объем продемонстрированных знаний при этом не менее 75%. Лабораторная работа со значительными ошибками.	Пороговый уровень	Удовлетворительно
Неправильный ответ на вопросы билета. Неполные ответы на дополнительные вопросы. Общий объем продемонстрированных знаний при этом менее 75. Отсутствие оформленной лабораторной работы	-	Неудовлетворительно

Требования к выполнению заданий, шкалы и критерии оценивания

Промежуточная аттестация проводится в соответствии с Положением о промежуточной аттестации обучающихся по программам высшего образования.

Контрольно-измерительные материалы промежуточной аттестации включают в себя теоретические вопросы, позволяющие оценить уровень полученных знаний. При оценивании используется количественная шкала оценок. Критерии оценивания приведены выше.

Перечень вопросов к зачету:

1. Краткая история развития технологии программирования. Причины возникновения и основные отличительные черты объектно-ориентированное программирования.
2. Общая характеристика объектной декомпозиции. Общее определения и возможные атрибуты объекта.
3. Характеристика принципов объектно-ориентированного программирования. Схема организации объекта.
4. Характеристика принципов объектно-ориентированного программирования. Сущность принципа абстрагирования.
5. Характеристика принципов объектно-ориентированного программирования. Сущность принципа иерархии (наследования и включения).
6. Характеристика принципов объектно-ориентированного программирования. Сущность принципа полиморфизма.
7. Определение объекта в терминах языка программирования. Общий синтаксис объявления класса, директивы видимости.
8. Определение объекта в терминах языка программирования. Общий синтаксис объявления класса, конструкторы и деструкторы.
9. Определение объекта в терминах языка программирования. Общий синтаксис объявления класса, виды полиморфизма и виды методов.
10. Синтаксис объявления свойства. Пример объявления свойства простого типа с методами чтения и записи.
11. Синтаксис объявления свойства. Пример объявления свойства типа «класс», реализации конструктора и деструктора.
12. Синтаксис объявления свойства. Пример объявления свойства перечислимого типа.
13. Синтаксис объявления свойства. Пример объявления свойства типа «множество».

14. Синтаксис объявления свойства. Пример объявления свойства типа ссылка на компонент, реализации метода уведомления об удалении.
15. Синтаксис объявления свойства. Пример объявления свойства типа ссылка на компонент, реализации метода уведомления об удалении.
16. Иерархия классов VCL. Общая характеристика класса TComponent.
17. Иерархия классов VCL. Общая характеристика класса TControl.
18. Иерархия классов VCL. Общая характеристика класса TWinControl.
19. Иерархия классов VCL. Общая характеристика класса TCustomControl.
20. Основные виды ресурсов. Приемы работы со строковыми ресурсами.
21. Основные виды ресурсов. Приемы работы с ресурсами курсоров.
22. Основные виды ресурсов. Приемы работы с ресурсами битовых изображений.
23. Основные виды ресурсов. Приемы работы с бинарными ресурсами.
24. Общая характеристика системных сообщения в Windows. Способы отправки сообщений.
25. Общая характеристика системных сообщения в Windows. Способы приема и обработки сообщений.
26. Общая характеристика системных сообщения в Windows. Определение и обработка пользовательских сообщений.
27. Диалоги. Функции вызова диалогов и способы анализа ответа пользователя.
28. Диалоги. Основные приемы создания и использования пользовательского диалога, способы анализа ответа пользователя.
29. Обзор системных диалогов VCL. Основные свойства, методы и обработчики событий.
30. Невизуальные компоненты. Основные причины создания и подходы к реализации.
31. Компоненты, работающие с данными. Общие принципы создания компонента, работающего с данными. Основные приемы создания компонента, отображающего данные.

19.3.2 Перечень практических заданий к зачету:

1. Создать компонент наследник класса TImage. Объявить и реализовать свойство, возвращающее площадь, занимаемую компонентом в мегапикселях. Разработать тестовое приложение, демонстрирующее использование нового свойства.
2. Создать графический элемент управления, визуально представляющий собой окружность (с диаметром, равным меньшей стороне компонента), расположенную по центру компонента.
3. Создать компонент-наследник TLabel, добавить свойство типа ссылка на компонент, реализовать метод уведомления об удалении компонента.
4. Создать компонент-наследник TEdit, который при нажатии клавиши Enter предпринимает попытку преобразовать введенный текст в целое число и при успешном преобразовании вызывает событие пользовательского типа, одним из аргументов которого является полученное целое число.
5. Создать приложение, выдающее подтверждение завершения работы программы в виде диалога. Текст сообщения должен быть строковым ресурсом.
6. Создать невидуальный компонент, выдающий при вызове определенного метода диалог, содержащий сведения о форме, на которой он расположен (ширины, высота, заголовок, количество компонентов)
7. Создать невидуальный компонент, выдающий при вызове определенного метода диалог, содержащий сведения о форме, на которой он расположен (ширины, высота, заголовок, количество компонентов)
8. Создать визуальный компонент, добавить свойство перечислимого типа.

9. Создать невидимый компонент, выдающий при вызове определенного метода диалог, содержащий сведения о форме, на которой он расположен (ширины, высота, заголовок, количество компонентов)

19.3.3 Перечень тестовых заданий для текущей аттестации:

1. Событие в объектно-ориентированном программировании – это:
1) свойство, являющееся указателем на процедуру; 2) метод; 3) объект; 4) процедура.
2. Как соотносятся выражения “TPrim=class” и “TPrim=class(Tobject)”?
1) противоречат друг другу; 2) эквивалентны; 3) дополняют друг друга; 4) описывают разные классы.
3. Значения свойств компонентов могут изменяться динамически?
1) да; 2) нет; 3) только для визуальных компонентов; 4) только для не визуальных компонентов.
4. Главная форма проекта- это форма,
1) всегда отображаемая поверх других форм; 2) отображаемая в модальном окне; 3) закрытие которой приводит к окончанию работы проекта; 4) все вместе.
5. Возможно ли размещение таких графических объектов как линия, окружность и др. непосредственно на форме?
1) невозможно; 2) возможно без ограничений; 3) возможно только в клиентской ее части; 4) возможно только в не клиентской ее части;
6. Какой из приведенных компонентов является невидимым?
1) TMainMenu; 2) TMemo; 3) TButton; 4) TStringList
7. Какое свойство компонента TCheckBox определяет состояние флажка?
1) Enabled 2) Visible 3) Checked 4) Hint 5) State
8. Для каких из приведенных событий в обработчике события можно использовать координаты мыши?
1) OnClick; 2) OnDbClick; 3) OnMouseDown; 4) OnMouseMove
9. Выберите правильные фрагменты программы:
1) Edit1.Text:=12; 2) Edit1.Caption:=12; 3) Edit1.Text:="12"; Label2.caption:='12';
10. Выберите правильные фрагменты программы:
1) Edit1.Text:='12,2'; 2) Edit1.Text:=IntToStr(12,2); 3) Edit1.Text:=FloatToStr(12,2); 4) Edit1.Text:=IntToStr (12/2);
11. Функция Execute диалогового компонента TOpenDialog принимает значение True в случае
1) успешного открытия диалогового окна; 2) выбора пользователем файла в диалоговом окне;
3) закрытия диалогового окна; 4) загрузки выбранного файла.
12. Какие из приведенных компонентов не могут быть использованы для ввода текстовой информации
1) компонент TEdit; 2) функция InputBox; 3) процедура ShowMessage; 4) функция InputQuery.
13. Какое из приведенных выражений обеспечивает доступ к строковому значению ячейки таблицы StringGrid1
1) str:=StringGrid1.ColCount; 2) str:=StringGrid1.Objects[3,4];
3) str:=StringGrid1.CellRect[3,4]; 4) str:=StringGrid1.Cells[3,4].

14. Текст всплывающей подсказки визуального компонента содержится в его свойстве
1) Hint; 2) Tag; 3) HelpContext; 4) HelpType.
15. Для создания контекстного меню используется компонент класса:
1) TButton; 2) TEdit; 3) TCheckBox; 4) TPopupMenu; 5) TBitBtn
16. Для создания многострочного редактируемого текстового поля используется компонент класса:
1) TLabel 2) TMainMenu 3) TMemo 4) TEdit 5) TForm
17. Свойство, позволяющее поместить на графическую кнопку класса TBitBtn, растровое изображение:
1) Caption 2) Picture 3) Image 4) Text 5) Glyph
18. Событие OnResize генерируется, когда:
1) Окно стало активным 2) Изменяется размер окна 3) Окно закрывается
4) Двигается мышка 5) Нет такого события
19. Как называется принцип создания класса на базе существующего:
1) Полиморфизм 2) Наследование 3) Перегрузка 4) Инкапсуляция 5) Делегирование
20. При уничтожении объекта класса автоматически вызывается специальный метод класса:
1) Деструктор 2) Член – функции 3) Член – данные 4) Конструктор 5) Разрушитель
21. Метод, вызываемый автоматически при создании объекта класса называется:
1) Деструктор 2) Делегирование 3) Член-данные 4) Конструктор 5) Член-функции
22. Вкладка Events в Object Inspector предназначена для...
1) Изменения свойств компонентов 2) Редактирования кода программы
3) Поиска компонентов 4) Просмотра структуры проекта 5) Определения событий
23. Событие OnCreate генерируется, когда...
1) Окно закрывается 2) Окно создается 3) Окно изменяет размеры
4) Пользователь щелкнул по форме 5) Окно меняет цвет
24. Скрытие полей объекта с целью обеспечения доступа к ним только посредством методов класса называют...
1) Наследованием 2) Делегированием 3) Полиморфизмом
4) Виртуализацией 5) Инкапсуляцией