ГАОУ ВО «ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ НАРОДНОГО ХОЗЯЙСТВА»

Утверждены решением
Ученого совета ДГУНХ,
протокол № 9
от 24 марта 2025 г.

Кафедра «Информационные технологии и информационная безопасность»

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ

«ТЕХНОЛОГИИ И МЕТОДЫ ПРОГРАММИРОВАНИЯ»

Направление подготовки 10.03.01 Информационная безопасность,
профиль «Безопасность автоматизированных систем»

Уровень высшего образования – бакалавриат

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 00C5A9FA0FAD6D6153888A98846B419FD3
Владелец: Бучаев Ахмед Гамидович
Действителен: с 11.07.2024 до 04.10.2025

Махачкала – 2025

УДК 681.518(075.8)
ББК 32.81.73

Составитель – Савина Елена Владимировна, кандидат физико-математических наук, доцент кафедры «Информационные технологии и информационная безопасность» ДГУНХ.

Внутренний рецензент – Гасанова Зарема Ахмедовна, кандидат педагогических наук, и.о. зав. кафедрой «Информационные технологии и информационная безопасность» ДГУНХ.

Внешний рецензент – Абдурагимов Гусейн Эльдарханович, кандидат физико-математических наук, доцент кафедры "Математические методы в экономике" Дагестанского государственного университета.

Представитель работодателя – Зайналов Джабраил Тажутдинович, директор регионального экспертно-аттестационного центра «Экспертиза».

Оценочные материалы по дисциплине «Технологии и методы программирования» разработаны в соответствии с требованиями федерального государственного образовательного стандарта высшего образования по направлению подготовки 10.03.01 Информационная безопасность, утвержденного приказом Министерства образования и науки Российской Федерации от 17 ноября 2020 г., № 1427, в соответствии с приказом Министерства науки и высшего образования Российской Федерации от 6.04.2021 г. № 245 «Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры»

Оценочные материалы по дисциплине «Технологии и методы программирования» размещены на официальном сайте www.dgunh.ru

Савина Е.В. Оценочные материалы по дисциплине «Технологии и методы программирования» для направления подготовки 10.03.01 Информационная безопасность, профиль «Безопасность автоматизированных систем». – Махачкала: ДГУНХ, 2025 г. – 53 с.

Рекомендованы к утверждению Учебно-методическим советом ДГУНХ 10 марта 2025 г.

Рекомендованы к утверждению руководителем основной профессиональной образовательной программы высшего образования – программы бакалавриата по направлению подготовки 10.03.01 Информационная безопасность, профиль «Безопасность автоматизированных систем», к.пед.н., Гасановой З.А.

Одобрены на заседании кафедры «Информационные технологии и информационная безопасность» 24 февраля 2025 г., протокол № 7..

СОДЕРЖАНИЕ

Назначение оценочных материалов……..4

РАЗДЕЛ 1. Перечень компетенций с указанием видов оценочных средств в процессе освоения дисциплины …..…….…….5

1.1 Перечень формируемых компетенций……5

1.2 Перечень компетенций с указанием видов оценочных средств…….5

РАЗДЕЛ 2. Задания, необходимые для оценки планируемых результатов обучения по дисциплине……....12

РАЗДЕЛ 3. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания……..40

РАЗДЕЛ 4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков, характеризующие этапы формирования компетенций…….4

4Лист актуализации оценочных материалов по дисциплине…….53

Назначение оценочных материалов

Оценочные материалы для текущего контроля успеваемости (оценивания хода освоения дисциплин), для проведения промежуточной аттестации (оценивания промежуточных и окончательных результатов обучения по дисциплине) обучающихся по дисциплине «Технологии и методы программирования» на соответствие их учебных достижений поэтапным требованиям образовательной программы высшего образования 10.03.01 Информационная безопасность, профиль «Безопасность автоматизированных систем»

Оценочные материалы по дисциплине «Технологии и методы программирования» включают в себя: перечень компетенций с указанием видов оценочных средств в процессе освоения дисциплины; описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания; типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения ОПОП; методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Оценочные материалы сформированы на основе ключевых принципов оценивания:

- валидности: объекты оценки должны соответствовать поставленным целям обучения;
- надежности: использование единообразных стандартов и критериев для оценивания достижений;
- объективности: разные обучающиеся должны иметь равные возможности для достижения успеха.

Основными параметрами и свойствами оценочных материалов являются:

- предметная направленность (соответствие предмету изучения конкретной дисциплины);
- содержание (состав и взаимосвязь структурных единиц, образующих содержание теоретической и практической составляющих дисциплины);
- объем (количественный состав оценочных материалов);
- качество оценочных материалов в целом, обеспечивающее получение объективных и достоверных результатов при проведении контроля с различными целями.

РАЗДЕЛ 1. Перечень компетенций с указанием видов оценочных средств в процессе освоения дисциплины

1.1 Перечень формируемых компетенций

Код компетенции	Формулировка компетенции
ОПК-7	Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности

1.2. Перечень компетенций с указанием видов оценочных средств

Формируемые компетенции	***Код и наименование индикатора достижения компетенции***	***Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций***	***Уровни освоения компетенций***	***Критерии оценивания сформированности компетенций***	***Виды оценочных средств***
ОПК-7. Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности	**ИОПК-7.2**. Выполняет отладку и тестирование программ	**Знать:** современные средства разработки и анализа программного обеспечения; основные принципы отладки и тестирования ПО.	Пороговый уровень	Обучающийся слабо (частично) знает современные средства разработки и анализа программного обеспечения; основные принципы отладки и тестирования ПО	**Блок А –** задания репродуктивного уровня – вопросы для обсуждения
			Базовый уровень	Обучающийся с незначительными ошибками и отдельными пробелами знает современные средства разработки и анализа программного обеспечения; основные принципы отладки и	

Формируемые компетенции	*Код и наименование индикатора достижения компетенции*	*Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций*	*Уровни освоения компетенций*	*Критерии оценивания сформированности компетенций*	*Виды оценочных средств*
				тестирования ПО	
			Продвинутый уровень	Обучающийся с требуемой степенью полноты и точности знает современные средства разработки и анализа программного обеспечения; основные принципы отладки и тестирования ПО	
		<u>Уметь</u>: выбирать необходимые инструментальные средства для отладки ПО; выбирать правильные алгоритмы для тестирования ПО; разбирать программные коды на ассемблере	Пороговый уровень	Обучающийся слабо (частично) умеет выбирать необходимые инструментальные средства для отладки ПО; выбирать правильные алгоритмы для тестирования ПО; разбирать программные коды на ассемблере	**Блок В** – задания реконструктивного уровня: – тестовые задания; – лабораторная работа; – тематика презентаций; – тематика рефератов
			Базовый уровень	Обучающийся с незначительными затруднениями умеет выбирать необходимые инструментальные средства для отладки ПО;	

Формируемые компетенции	*Код и наименование индикатора достижения компетенции*	*Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций*	*Уровни освоения компетенций*	*Критерии оценивания сформированности компетенций*	*Виды оценочных средств*
				выбирать правильные алгоритмы для тестирования ПО; разбирать программные коды на ассемблере	
			Продвинутый уровень	Обучающийся умеет выбирать необходимые инструментальные средства для отладки ПО; выбирать правильные алгоритмы для тестирования ПО; разбирать программные коды на ассемблере	
		Владеть: навыками отладки и тестирования программного обеспечения различного назначения; элементарными навыками работы с отладчиками и дизассемблерами	Пороговый уровень	Обучающийся слабо (частично) владеет навыками отладки и тестирования программного обеспечения различного назначения; элементарными навыками работы с отладчиками и дизассемблерами	**Блок С** – задания практико-ориентированного уровня – решение кейс-задач – тематика курсовых проектов
			Базовый уровень	Обучающийся с небольшими затруднениями	

Формируемые компетенции	*Код и наименование индикатора достижения компетенции*	*Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций*	*Уровни освоения компетенций*	*Критерии оценивания сформированности компетенций*	*Виды оценочных средств*
				владеет навыками отладки и тестирования программного обеспечения различного назначения; элементарными навыками работы с отладчиками и дизассемблерами	
			Продвинутый уровень	Обучающийся свободно владеет навыками отладки и тестирования программного обеспечения различного назначения; элементарными навыками работы с отладчиками и дизассемблерами	
	ИОПК-7.3. Разрабатывает и реализовывает на языке высокого уровня алгоритмы решения профессиональных задач	**Знать:** – современные средства разработки и анализа программного обеспечения на языках высокого уровня; – базовые алгоритмы работы с	Пороговый уровень	Обучающийся слабо (частично) знает современные средства разработки и анализа программного обеспечения на языках высокого уровня; базовые алгоритмы	**Блок А –** задания репродуктивного уровня – вопросы для обсуждения; – тестовые задания.

Формируемые компетенции	*Код и наименование индикатора достижения компетенции*	*Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций*	*Уровни освоения компетенций*	*Критерии оценивания сформированности компетенций*	*Виды оценочных средств*
		данными; – основные понятия обратного программирования.		работы с данными; основные понятия обратного программирования	
			Базовый уровень	Обучающийся с незначительными ошибками и отдельными пробелами знает современные средства разработки и анализа программного обеспечения на языках высокого уровня; базовые алгоритмы работы с данными; основные понятия обратного программирования	
			Продвинутый уровень	Обучающийся с требуемой степенью полноты и точности знает современные средства разработки и анализа программного обеспечения на языках высокого	

Формируемые компетенции	*Код и наименование индикатора достижения компетенции*	*Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций*	*Уровни освоения компетенций*	*Критерии оценивания сформированности компетенций*	*Виды оценочных средств*
				уровня; базовые алгоритмы работы с данными; основные понятия обратного программирования	
		<u>Уметь:</u> – выбирать необходимые инструментальные средства для разработки программ в различных операционных системах и средах; – составлять, тестировать, отлаживать и оформлять программы на языках высокого уровня, включая Delphi / Lazarus.	Пороговый уровень	Обучающийся слабо (частично) умеет выбирать необходимые инструментальные средства для разработки программ в различных операционных системах и средах; составлять, тестировать, отлаживать и оформлять программы на языках высокого уровня, включая Delphi / Lazarus	**Блок В** – задания реконструктивного уровня: – тестовые задания; – лабораторная работа; – тематика презентаций; – тематика рефератов; – вопросы для коллоквиума
			Базовый уровень	Обучающийся с незначительными затруднениями умеет выбирать необходимые инструментальные средства для разработки программ в различных операционных	

Формируемые компетенции	*Код и наименование индикатора достижения компетенции*	*Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций*	*Уровни освоения компетенций*	*Критерии оценивания сформированности компетенций*	*Виды оценочных средств*
				системах и средах; составлять, тестировать, отлаживать и оформлять программы на языках высокого уровня, включая Delphi / Lazarus	
			Продвинутый уровень	Обучающийся умеет выбирать необходимые инструментальные средства для разработки программ в различных операционных системах и средах; составлять, тестировать, отлаживать и оформлять программы на языках высокого уровня, включая Delphi / Lazarus	
		Владеть: – навыками подбора алгоритмов и разработки программ для решения практических задач на языке программиров	Пороговый уровень	Обучающийся слабо (частично) владеет навыками подбора алгоритмов и разработки программ для решения	**Блок С** – задания практико-ориентированного уровня – решение кейс-задач – тематика курсовых проектов

Формируемые компетенции	*Код и наименование индикатора достижения компетенции*	*Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций*	*Уровни освоения компетенций*	*Критерии оценивания сформированности компетенций*	*Виды оценочных средств*
		ания высокого уровня.		практических задач на языке программирования высокого уровня	
			Базовый уровень	Обучающийся с небольшими затруднениями владеет навыками подбора алгоритмов и разработки программ для решения практических задач на языке программирования высокого уровня	
			Продвинутый уровень	Обучающийся свободно владеет навыками подбора алгоритмов и разработки программ для решения практических задач на языке программирования высокого уровня	

РАЗДЕЛ 2. Задания, необходимые для оценки планируемых результатов обучения по дисциплине

Для проверки сформированности компетенции ОПК-7

Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности

ИОПК-7.2. Выполняет отладку и тестирование программ

Блок А. Задания репродуктивного уровня («знать»)

А1. Вопросы для обсуждения

1. Основные понятия объектно-ориентированного программирования
2. Понятия проекта и приложения в Delphi / Lazarus
3. Классы и объекты в разных программных средах
4. Визуальная среда программирования
5. Типы данных и основные управляющие структуры
6. Графические возможности Delphi / Lazarus
7. Обратное программирование, функции и возможности
8. Архитектура процессора
9. Основные понятия дизассемблирования
10. Понятие технологии программирования

А2. Фонд тестовых заданий по дисциплине

1. К базовым понятиям ООП относятся

1) алгоритм	2) объект	3) оператор
4) полиморфизм	5) процедура	6) инкапсуляция
7) класс	8) наследование	9) компиляция

2. Установить соответствие

	Свойство		***Описание***
1	Полиморфизм	A	строит иерархию объектов
2	Инкапсуляция	B	использует виртуальные или перегружаемые элементы
3	Наследование	C	скрывает внутренние данные объекта

3. Процесс преобразования исходного кода в исполняемый называется _____

4. Описание класса производится в разделе

1) `program`	2) `unit`	3) `uses`	4) `type`	5) `var`

5. Совокупность данных, свойств и методов называется _____

6. Главное окно Delphi включает

1) дерево объектов	2) главное меню	3) инспектор объектов
4) панели инструментов	5) палитру компонентов	
6) редактор кода	7) окно формы	

7. Заголовки процедур и функций объявляются в разделе
   1) `implementation` 2) `type` 3) `interface`
   4) `var` 5) `initialization`

8. Ошибками времени компиляции называют
   1) синтаксические ошибки 2) алгоритмические ошибки
   3) логические ошибки 4) грамматические ошибки

9. Исключениями называют
   1) ошибки времени компиляции 2) алгоритмические ошибки
   3) синтаксические ошибки 4) ошибки времени выполнения

10. Участок кода, который может вызвать ошибку времени выполнения, следует располагать внутри конструкции
   1) `repeat … until` 2) `with … do` 3) `try … except`
   4) `for … do` 5) `try … finally`

11. Поток это
   1) объект, получающий время процессора
   2) активное приложение
   3) процесс в реальном времени
   4) каждая из задач, решаемых в одно и то же время

12. Порция времени, выделяемая процессором потоку, называется _____

13. При работе с потоками могут возникнуть проблемы
   1) гонок и пересечений 2) столкновений и тупиков
   3) тупиков и гонок 4) пересечений и тупиков

14. Ситуация, когда один поток ожидает ресурс, который в данный момент принадлежит другому потоку, и наоборот, называется _____

15. Файл модуля потока начинается словом
   1) `Thread` 2) `Project` 3) `Unit` 4) `Library`

Блок В. Задания реконструктивного уровня («уметь»)

В.1 Фонд тестовых заданий по дисциплине

1. Переменная `s` имеет тип `String`, а переменная `i` – `Integer`. В результате выполнения кода:

```
s:= '10';
i:= 5;
```

```
s:= s + IntToStr(i);
```
в переменной s будет находиться
1) строка ‘15’
2) произойдет ошибка в результате несовместимости типов
3) строка ‘105’
4) строка ‘10’

2. Выберите все верные утверждения
1) Выражение `ln(StrToInt(Edit1.Text))` содержит ошибку
2) Следующий фрагмент кода не содержит ошибок:
```
var a, b: integer; s1: real;
begin
  s1:= a*b – b;
end.
```
3) Результат действия оператора присваивания
`s12:= Edit1.Text` имеет строковый тип данных
4) Функция `ord(x)` применима только к данным символьного типа

3. Выберите все верные утверждения
1) При возникновении ошибки в программе, запущенной из `Windows`, на экране появляется сообщение об ошибке без указания типа ошибки
2) Алгоритмические ошибки выявляются на этапе компиляции
3) Структура `try..finally` предназначена для непосредственного перехвата исключительных ситуаций и их исправления
4) Операции ввода/вывода могут привести к возникновению исключительной ситуации

Задания 4-6: изучите программный код и ответьте на вопросы

```
1   procedure TForm1.btnShowClick(Sender: TObject);
  2   var MyPicture: TBitmap;
  3   begin
  4    try
  5      MyPicture:= TBitmap.Create;
  6      MyPicture.LoadFromFile ('photo.bmp');
  7      Form1.Canvas.Draw(10, 20, MyPicture);
  8    finally
  9      MyPicture.Free;
  10   end;
  11  end;
```

4. Действие, которое будет выполнено в любом случае
1) загрузка картинки из файла
2) создание объекта `MyPicture`

3) присваивание значения объекту `MyPicture`
4) уничтожение объекта `MyPicture`

5 Операторы, которые могут вызвать исключение, находятся в строках _____

6. Классы исключений, которые может инициировать выполнение этого кода
1) `EZeroDivide` 2) `EOverFlow` 3) `EConvertError`
4) `EFOpenError` 5) `ElnvalidGraphicOperation`

7. Выберите верное утверждение
1) процедура `Resume` приостанавливает выполнение потока
2) `TThread` – абстрактный класс, его можно использовать только через классы-потомки
3) функция `WaitFor()` предназначена для определения количества запусков потока

8. Изучите программный код и ответьте на вопросы:
1) на каком этапе работы приложения произойдут изменения, заданные процедурой?
2) как изменится вид формы после исполнения процедуры? Опишите все изменения.

```
procedure TForm1.Button1Click(Sender: TObject);
     var
       r1, r2: TRect;
     begin
       r1:= Rect(20, 20, 100,100);
       r2:= Rect(20, 120, 100, 200);
       with form1.Canvas do
       begin
         brush.Color:= clRed;
         fillRect(r1);
         brush.Color:= clgreen;
         frameRect(r2)
       end;
     end;
```

9. Определите, какие действия выполняет процедура и перечислите их в порядке исполнения

```
procedure TForm1.ButtonClick(Sender: TObject);
begin
  try
    Image1.Picture.LoadFromFile('D:\Pictures\photo.bmp');
  except
    Beep;
    ShowMessage('Невозможно …!');
  end;
```

```
end;
```
1) генерирует звуковой сигнал в случае отсутствия требуемого файла
2) выводит окно с сообщением об ошибке
3) пытается загрузить текстовый файл
4) загружает видеоклип из файла
5) пытается загрузить файл с изображением

10. В результате выполнения фрагмента программы

```
function sept(a, b: real; n: integer): real;
   begin
     sept:= exp(n*ln(a)) + b;
   end;
…
write(sept(3, 8, 2);
```

на экран будет выведен результат ___

11. К принципам структурного подхода относятся
1) модульность программ
2) обязательное использование рекурсии
3) организация структуры модуля в виде набора процедур
4) нисходящее проектирование рациональной иерархии модулей программ
5) сквозной структурный контроль программных комплексов в целом и составляющих их модулей

B2. Лабораторные работы

Лабораторная работа 1. Обработка исключительных ситуаций.

Цель работы*: изучение возможностей предотвращения ошибок специфическими средствами Delphi.*

Порядок работы приложения: в текстовые поля Edit1 и Edit2 вводятся значения электрического напряжения и сопротивления, соответственно, при нажатии на кнопку Button1 по закону Ома вычисляется значение силы тока и результат выводится на метку Label5.

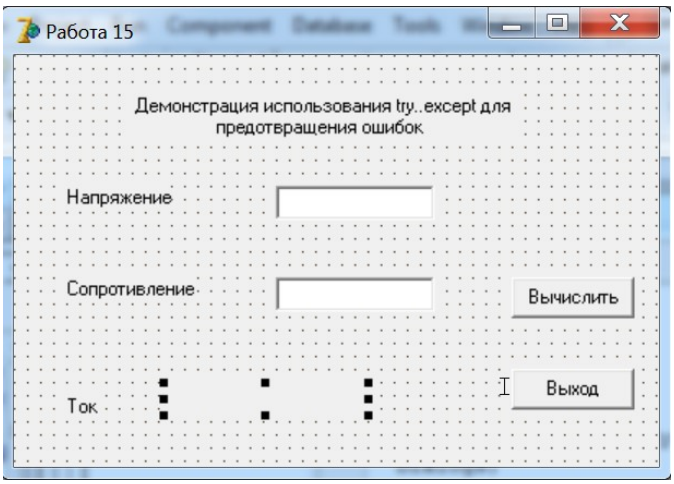

Рис. 1. Окно приложения

Расположите компоненты на форме и определите для них свойство Caption, как показано на рис. 1. Для того чтобы надпись на компоненте Label1 (заголовок на форме) можно было разместить в 2 строки, задайте свойству WordWrap значение true.

Обработчик события OnClick для кнопки Вычислить (Button1):

```
procedure TForm1.Button1Click(Sender: TObject);
var
  u: real;     // напряжение
  r: real;     // сопротивление
  i: real;     // ток
begin
  Label5.Caption:= ‘’;      // очищаем метку Label5
  try
    // инструкции, которые могут вызвать исключение (ошибку)
   u:= StrToFloat(Edit1.Text);
   r:= StrToFloat(Edit2.Text);
   i:= u/r;
  except                    // секция обработки исключений
   on EZeroDivide do        // деление на ноль
    begin
     ShowMessage(‘Сопротивление не может быть равно нулю!’);
     exit;
    end;
   on EConvertError do       // ошибка преобразования строки в число
    begin
     ShowMessage(‘Напряжение и сопротивление должны быть ’
          + ‘заданы числом’ + #13 +
         ‘При записи дробного числа используйте запятую’);
     exit;
    end;
```

```
  end;
  Label5.Caption:= FloatToStr(i) + ‘ A’;
end;
```

На рис. 2 показан вид экрана при правильном вводе данных, на рис. 3-5 – вид экрана при попытке ввести для сопротивления нулевое значение и при вводе неверных символов.

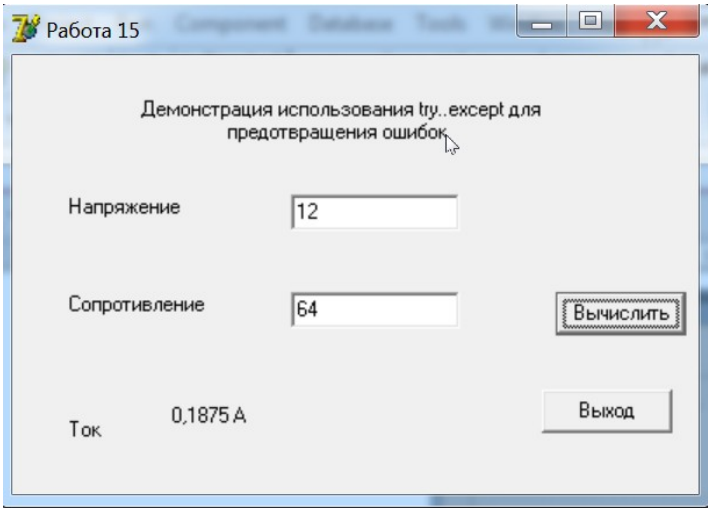

Рис. 2. Вид окна приложения при правильном вводе данных

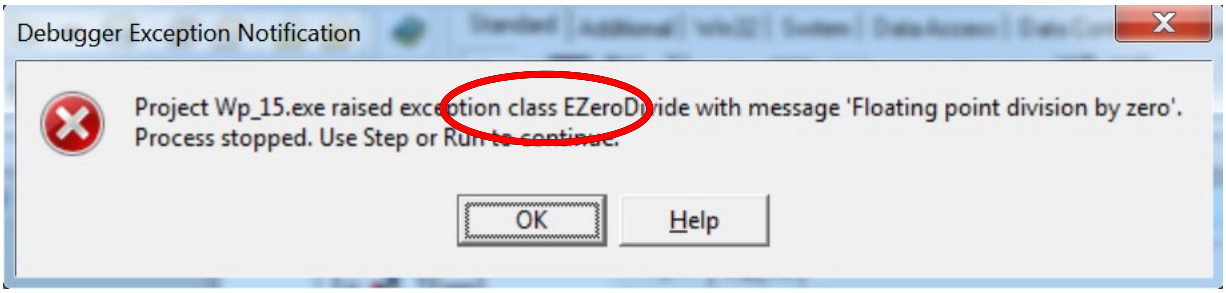

Рис. 3. Сообщение об ошибке при вводе нулевого значения сопротивления

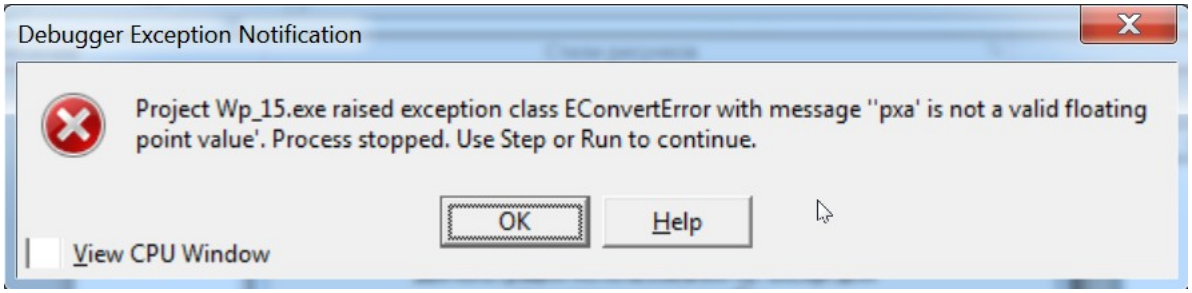

Рис. 4. Сообщение об ошибке при попытке ввести неверные символы

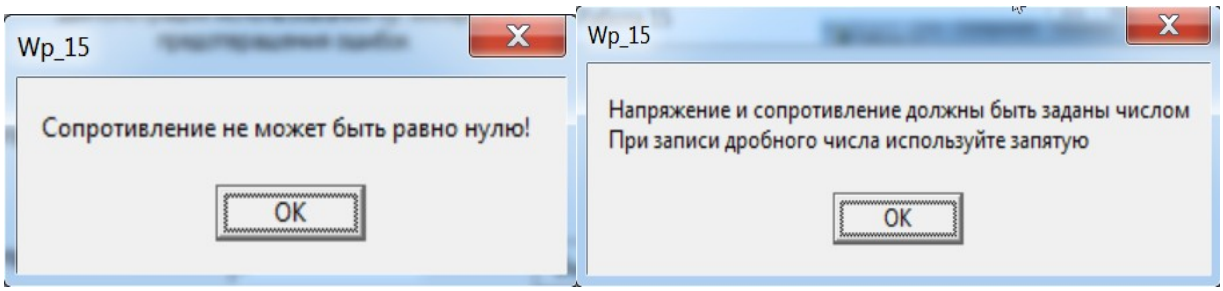

Рис. 5. Диалоговые окна приложения с сообщениями об ошибках (деление на ноль и ввод неверных символов)

Задание 1. Разработайте приложение-калькулятор, выполняющий 4 арифметических действия. Используйте конструкцию try..except для операции деления (исключить деление на ноль).

Задание 2. Усложните калькулятор: добавьте операции div и mod и используйте обработку неправильного ввода данных для этих операций (операции div и mod определены только для целых чисел).

B3. Тематика презентаций

1. Взаимодействие приложения с операционной системой
2. Спагетти-код и борьба с ним
3. Способы защиты программы от отладки и дизассемблирования
4. Дизассемблирование – основные понятия, инструменты, цели
5. Устройство процессора и режимы его работы

B4. Тематика рефератов

1. Использование функций операционной системы в Delphi / Lazarus
2. Сравнительный анализ структурного и объектно-ориентированного подхода в программировании
3. Основные алгоритмы дизассемблирования
4. Способы защиты информации
5. Базовые техники хакерства и борьба с ним
6. Декомпиляция и дизассемблирование

B5. Вопросы к коллоквиуму

1. Разработка приложений баз данных в Delphi / Lazarus
2. Отладка программ
3. Тестирование ПО
4. Средства защиты информации
5. Основные этапы разработки программ
6. Критерии надежности программ
7. Системное, прикладное и инструментальное программирование

Блок С. Задания практикоориентированного уровня для диагностирования сформированности компетенций («владеть»)

С1. Кейс-задача

Задача 1.

Разработать приложение в Delphi / Lazarus (допускается PascalABC.NET, C++, C#, Python) со следующим функционалом. Задается двумерный числовой массив. Приложение дает возможности:

1) вычисления сумм и средних значений по строкам и столбцам;
2) обнуления всего массива.

При этом массив может заполняться либо случайными числами, либо значениями, вводимыми с клавиатуры (предусмотреть обе возможности по выбору пользователя)

ИОПК-7.3. Разрабатывает и реализовывает на языке высокого уровня алгоритмы решения профессиональных задач

Блок А. Задания репродуктивного уровня («знать»)

А1. Вопросы для обсуждения

1. Основные понятия ООП
2. Структура типов данных в Delphi / Lazarus
3. Реализация линейных и разветвляющихся алгоритмов
4. Средства реализации циклов в Delphi / Lazarus
5. Структура проекта
6. Синтаксис языка ассемблера
7. Понятие обратного программирования

А2. Фонд тестовых заданий по дисциплине

1. Идентификатор может содержать следующие символы
   1) знаки препинания 2) буквы латинского алфавита
   3) буквы русского алфавита 4) знаки арифметических операций
   5) знак подчеркивания 6) пробел 7) цифры

2. Алфавит языка Delphi включает
   1) пробел 2) цифры 0..9 3) символы ?, %
   4) символы Δ, → 5) символы {, }, @, #
   6) буквы русского алфавита 7) буквы латинского алфавита

3. Поменять цвет кнопки `Button` можно
   1) с помощью свойства `Color` в Инспекторе объектов
   2) программно, с помощью оператора присваивания `Button.Color:=`
   3) невозможно

4. Установить соответствие

	Компонент		***Тип***
1	`Label`	A	визуальный
2	`Timer`	B	невизуальный
3	`StringGrid`		
4	`OpenDialog`		
5	`DrawGrid`		
6	`BitBtn`		

5. Выберите все верные утверждения
   1) Компоненты `Timer` и `MainMenu` не имеют свойства `Name`
   2) Главный файл проекта имеет расширение `.dpr`
   3) Свойства компонентов можно изменять только с помощью вкладки `Properties` Инспектора объектов
   4) Собрание объектов с аналогичными свойствами и методами называется класс

6. Ввод текстовой информации в Delphi можно реализовать с помощью
   1) поля `Memo` 2) ярлыка `Label` 3) кнопки `BitBtn`
   4) таблицы `StringGrid` 5) поля `Edit` 6) панели `Panel`
   7) поля `Image` 8) окна ввода 9) таблицы `DrawGrid`

7. Выберите все верные утверждения
   1) Delphi не поддерживает тип данных `single`
   2) Логический тип данных относится к порядковым
   3) Тип данных `Word` предназначен для хранения строк
   4) В переменную типа `Char` нельзя поместить значение ‘`abc`’

8. Установить соответствие

1	(* *)	A	однострочный комментарий
2	{$	B	получение адреса
3	^	C	многострочный комментарий
4	@	D	директива компилятора
5	//	E	указатель

9. Установить соответствие

1	`Eof`	A	Открывает файл
2	`Reset`	B	Закрывает файл
3	`Eoln`	C	Определяет конец файла
4	`Seek`	D	Определяет конец строки
5	`Close`	E	Находит в файле элемент с заданным номером

10. Оптимальный вариант сохранения приложения в начале разработки
   1) `File` → `Save` 2) `File` → `Save As`
   3) `File` → `Save As Project` 4) `File` → `Save All`

11. Подключение модальной формы к основной выполняется с помощью команды
   1) `File` → `Use Unit`
   2) `File` → `New` → `Form`

3) Component → Install Component
4) Project → Add New Project
5) View → Forms

12. Установите соответствие

1	StrToInt	A	преобразует действительное число в текст
2	StrToFloat	B	преобразует целое число в текст
3	FloatToStr	C	преобразует текст в действительное число
4	IntToStr	D	преобразует текст в целое число

13. Параметры, которые задаются при обращении к подпрограмме, называются _____

14. Для использования графических средств в Delphi предназначены классы
1) TPen 2) TBrush 3) TCursor 4) TControl
5) TPoint 6) TCanvas

15. Компонент Image предназначен для
1) отображения графического изображения на форме
2) установки цвета компонента
3) размещения рисунка на кнопке
4) рисования графических примитивов

Блок В. Задания реконструктивного уровня («уметь»)

В.1 Фонд тестовых заданий по дисциплине

1. Выберите все верные утверждения
1) Окно формы предназначено для создания и редактирования программного кода
2) Главное окно содержит дерево объектов и панели инструментов
3) Окно инспектора объектов имеет 2 вкладки – Свойства и События
4) Команда File → New → Application открывает окно формы и окно модуля формы нового приложения

2. Выберите все верные утверждения
1) Операции div и mod определены только для целых чисел
2) Операция сравнения для строк не определена
3) Раздел Interface кода модуля не содержит описания переменных и констант
4) Комментарий – это произвольный текст, оформленный особым образом, который игнорируется компилятором

3. Выберите все верные утверждения
   1) Компонент `Timer` находится на вкладке `System`
   2) `Delphi` – строго типизированный объектно-ориентированный язык программирования
   3) Отдельный экземпляр класса называется модуль
   4) Просмотреть главный файл проекта невозможно

4. Определить типы данных, которые используются в функции

```
function f1(a: ______; b: _____): ______;
  begin
   f1:= false;
   a:= b mod 3 = 0;
   if a then f1:= true
  end;
```

5. Изучите программный код, определите все действия, которые выполняются в процедуре, и расположите их в правильной последовательности.

```
procedure TForm1.bbRunClick(Sender:TObject);
     var i, n, s: integer;
     begin
       if Edit1.Text = '' then Exit;
       n:= StrToInt(Edit1.Text);
       Edit1.Text:= '';
       Edit1.SetFocus;
       s:= 0;
       for i:= 1 to n do s:= s + i;
       Memo1.Lines.Add (IntToStr(s));
     end;
```

   1) активизация поля редактирования
   2) вычисление суммы целых чисел от 1 до `s`
   3) очистка поля редактирования
   4) выход из процедуры в случае отсутствия текста в поле редактирования
   5) ввод значения `s` через поле редактирования
   6) добавление значения `s` в многострочный редактор
   7) вычисление суммы целых чисел от 1 до `n`
   8) ввод значения `n` через поле редактирования

6. Процедура

```
procedure TForm1.FormClick(Sender: TObject);
begin
  Form1.Canvas.MoveTo(10,10);
  Form1.Canvas.LineTo(200,10);
end;
```

   отрисовывает на поверхности формы

1) ломаную линию, состоящую из двух отрезков
2) два отрезка – из начала координат в точку (10, 10) и из начала координат в точку (200, 10)
3) горизонтальный отрезок
4) вертикальный отрезок

7. Выберите все верные утверждения
1) Начало координат в графическом окне Delphi расположено в нижнем левом углу
2) Ось OY объекта Canvas направлена сверху вниз
3) Свойство `PenPos` определяет текущую позицию пера
4) В операторе `Canvas.Pixels[x,y]:= clRed`; количество параметров в свойстве `Pixels` указано неверно

8. Выберите все верные утверждения
1) API-функция MessageBeep генерирует стандартный системный звук, тип которого указан параметром uType
2) При вызове функции MessageBeep воспроизведение звука осуществляется асинхронно с дальнейшим выполнением приложения
3) Свойство FrameCount определяет размер кадра в клипе
4) Процедура Stop прерывает воспроизведение видеоклипа и устанавливает свойство Active в значение True

Задания 9-10: по приведенному программному коду определить компонентный состав и функционал приложения

9.
```
unit z_01;

interface

uses
   Windows, Messages, SysUtils, Variants, Classes, Graphics,
Controls, Forms, Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    lbl1: TLabel;
    btn1: TButton;
    edt1: TEdit;
    edt2: TEdit;
    btn2: TButton;
    procedure FormActivate(Sender: TObject);
    procedure btn1Click(Sender: TObject);
    procedure btn2Click(Sender: TObject);
```

```
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation
{$R *.dfm}

procedure TForm1.FormActivate(Sender: TObject);
begin
  btn1.Caption := 'Старт';
  btn2.Enabled := False
end;

procedure TForm1.btn1Click(Sender: TObject);
begin
  Randomize;
  btn2.Enabled := True;
  btn2.Caption := 'Выход';
  edt1.Text := IntToStr(Random(1001));
  edt2.Text := IntToStr(Random(1001));
                                lbl1.Caption                := 
IntToStr(StrToInt(edt1.Text)*StrToInt(edt2.Text))
end;

procedure TForm1.btn2Click(Sender: TObject);
begin
  Close
end;

end.
```

10.
```
unit z_02;

interface

uses
   Windows, Messages, SysUtils, Variants, Classes, Graphics, 
Controls, Forms, Dialogs, StdCtrls, ExtCtrls, Buttons;
```

```
type
  TForm1 = class(TForm)
    btn1: TButton;
    pnl1: TPanel;
    edt1: TEdit;
    btn2: TBitBtn;
    procedure FormActivate(Sender: TObject);
    procedure btn1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation
{$R *.dfm}

procedure TForm1.FormActivate(Sender: TObject);
begin
  with pnl1 do
   begin
    Align := alTop;
    Cursor := crCross;
   end;
  btn2.Visible := False
 end;

procedure TForm1.btn1Click(Sender: TObject);
begin
  edt1.Text := 'Сегодня ' + DateToStr(Date);
  pnl1.Color := clLime;
  with btn2 do
   begin
     Visible := True;
     kind := bkClose
   end;
end;

end.
```

В2. Лабораторные работы

Лабораторная работа 1. Компонент ListBox. Нахождение делителей числа.
Цель работы: *изучение приемов и методов разработки приложений в Delphi на простых примерах с помощью компонентов ListBox, Panel, Bevel.*

1. Разработка интерфейса

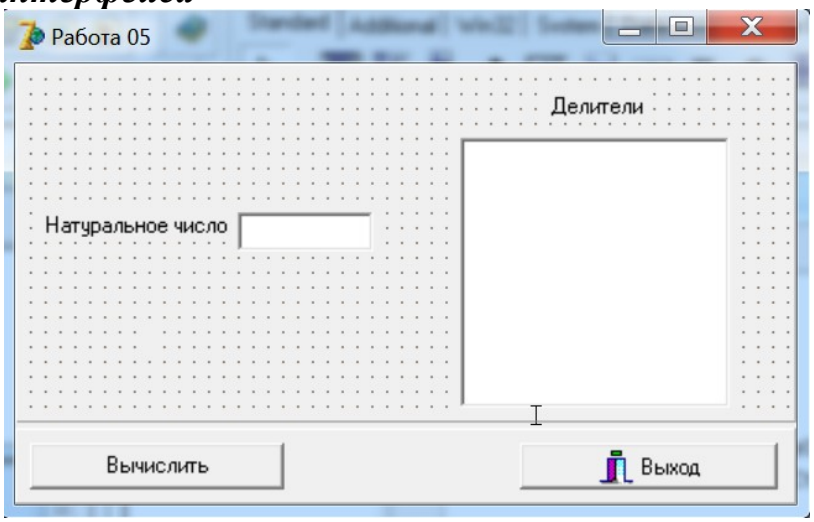

Рис. 6. Вид формы после размещения компонент и задания их свойств

Приложение состоит из формы (Form1), на которой внутри объемной рамки (компонент Bevel) расположены три метки (компонент Label), текстовое поле (компонент Edit) и многострочное текстовое поле (компонент ListBox). Две кнопки (компоненты Button и BitBtn) расположены внутри панели (компонент Panel).

Порядок работы приложения: В текстовое поле Edit вводится натуральное число, при нажатии на кнопку Вычислить (Button) определяются его делители и выводятся в многострочное текстовое поле ListBox. Если введенное число простое, соответствующее сообщение выводится в поле метки (Label). Кнопка Выход (BitBtn) предназначена для завершения работы приложения.

Разместите на форме компоненты примерно так, как показано на рис. 6, и задайте их свойства в соответствии с таблицей 1.

Таблица 1

Компонент	***Свойство***	***Значение***
Form1	Caption	Работа 05
Bevel1	Align	AlTop
	Style	bsRaised
Label1	Caption	Натуральное число
Label2	Caption	Делители
Label3	Caption	(очистить поле)
Edit1	Text	(очистить поле)
Button1	Caption	Вычислить
BitBtn1	Kind	bkClose

	Caption	Выход
Panel1	Caption	(очистить поле)
	Align	alBottom

2. Определение поведения приложения

1. Компонент `Edit1`, событие `OnKeyPress`:

```
procedure TForm1.Edit1KeyPress(Sender: TObject;
          var Key: Char);
begin
  if not (key in['0'..'9', '-', #8]) then Key:= #0;
end;
```

2. Компонент `Button1`, событие `OnClick`:

```
procedure TForm1.Button1Click(Sender: TObject);
var
  x, half, divider: word;
  i: byte;
begin
  Label3.Caption:= '';        // очистка метки при новом запуске
  ListBox1.Items.Clear;     // очистка многострочного текстового поля
  x:= StrToInt(Edit1.Text);
  half:= x div 2;
  divider:= 2; i:= 0;
  while divider <= half do
   begin
    if x mod divider = 0 then
     begin
      inc(i);
      ListBox1.Items.Add(IntToStr(i) + '-й делитель=' +
        IntToStr(divider));
     end;
    inc(divider);
   end;
  if i = 0 then Label3.Caption:= 'Число простое';
end;
```

Сохраните приложение и запустите его.

Задание. Измените код таким образом, что если число не является простым, в поле Label3 выводится сообщение «Число имеет … делителей», где вместо многоточия указано количество делителей числа.

Указание. Количество делителей равно значению переменной i.

Лабораторная работа 2. Отображение графических файлов.

***Цель работы**: изучение методов отображения и сохранения графических файлов.*

Порядок работы приложения: при нажатии на кнопку Загрузить появляется диалоговое окно открытия графического файла, который будет отображаться в поле Image. Кнопка Сохранить предназначена для сохранения графического файла, отображенного в поле Image.

1. Разработка интерфейса

Интерфейс представляет собой форму с кнопками для загрузки и сохранения файлов и поля для их отображения.

Разместите на форме `Form1` две кнопки `Button`, одну кнопка с битовым рисунком `BitBtn`, поле для отображения графики Image, метку Label и два диалоговых компонента `OpenPictureDialog` и `SavePictureDialog` (рис. 7).

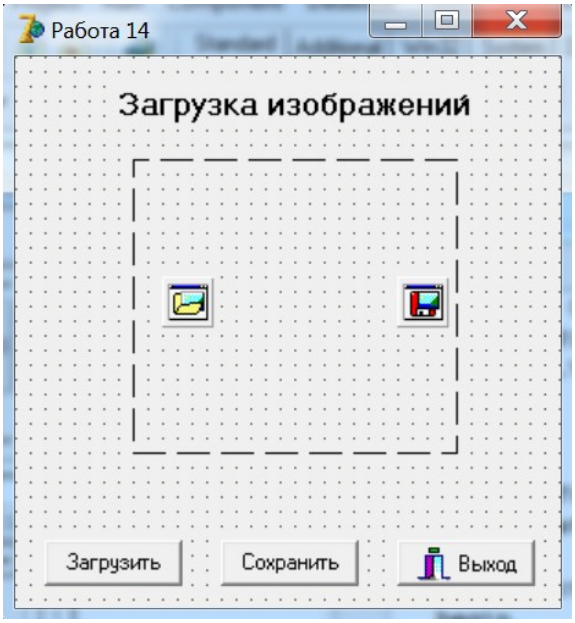

Рис. 7. Окно приложения на стадии проекта

Кнопку `BitBtn` оформите как кнопку, закрывающую приложение. Свойство `Caption` для компонентов задайте в соответствии с рис. 2.

Компоненты `OpenPictureDialog` и `SavePictureDialog` находятся на вкладке ***Dialogs***.

2. Определение поведения приложения

1. Кнопка Загрузить (`Button1`), событие `OnClick`:

```
procedure TForm1.Button1Click(Sender: TObject);
var fName: string;
begin
  if OpenPictureDialog1.Execute then
   begin
    fName:= OpenPictureDialog1.FileName;
```

```
    Image1.Picture.LoadFromFile(fName);
   end;
end;
```

2. Кнопка Сохранить (Button2), событие OnClick:

```
procedure TForm1.Button2Click(Sender: TObject);
var fName: string;
begin
  if SavePictureDialog1.Execute then
   begin
    fName:= SavePictureDialog1.FileName;
    Image1.Picture.SaveToFile(FName);
   end;
end;
```

Сохраните и запустите приложение.

Задание. Измените программный код так, чтобы при открытии приложения на форме была видна только кнопка Загрузить, а область рисунка занимала всю форму, кнопка Сохранить появлялась после нажатия на кнопку Загрузить, а кнопка Выход – после нажатия на кнопку Сохранить (рис. 8).

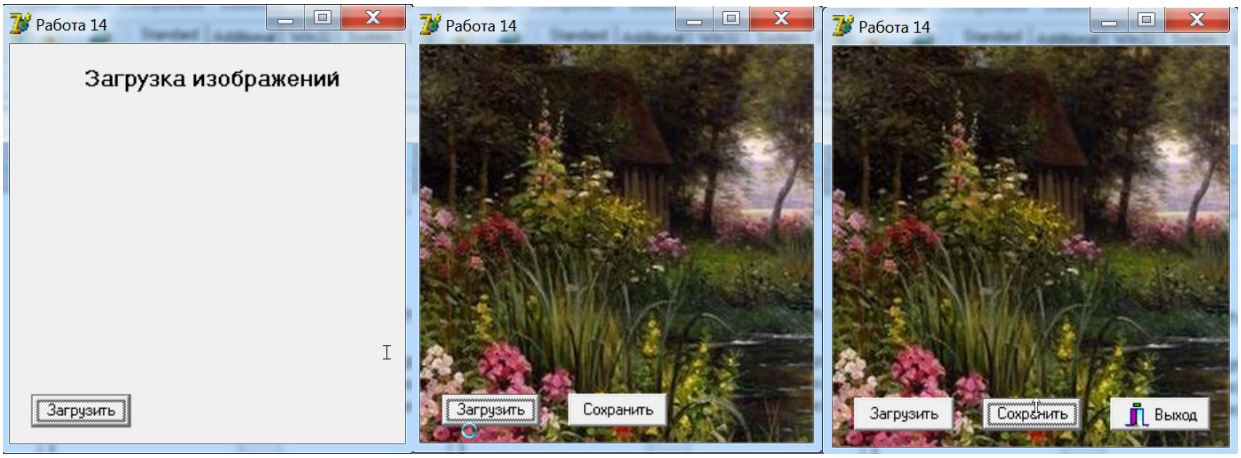

Рис. 8. Работа приложения после изменения кода

В3. Тематика презентаций

1. Подключение DLL в разных языках программирования (на примере 2-3 ЯП)
2. Управление окнами в многодокументном приложении
3. Свойства и методы объекта Screen
4. Разработка ScreenSaver
5. Средства анимации в Delphi / Lazarus
6. Объекты OLE в современных языках программирования
7. История языков ассемблера
8. Особенности работы с консольными приложениями
9. Многозадачность и многопоточность

В4. Тематика рефератов

2. Организация консольных приложений в Delphi и в C++
3. Основные принципы структурного подхода в программировании
4. Теорема Бёма-Якопини (теорема о структурном программировании)
9. Организация приложения Delphi
10. Взаимодействие приложений с операционной системой
11. Защита приложений от несанкционированного доступа
12. Процессы и потоки
13. Основные принципы структурного подхода в программировании
14. Проектирование структуры программы
15. Обзор технологий программирования
16. Архитектура компьютерных систем

В5. Вопросы к коллоквиуму

1. Парадигмы программирования
5. Структурное программирование
6. Декларативное программирование
7. Параллельное программирование
8. Объектно-ориентированное программирование
9. Функциональное программирование
10. Логическое программирование
11. Модульное программирование
12. Технологии OLE и COM
13. CASE-технология
14. Технология CORBA
15. Технология RAD

Блок С. Задания практикоориентированного уровня для диагностирования сформированности компетенций («владеть»)

С1. Кейс-задача

Задача 1.

Работа предназначена для выполнения без компьютера

Задание 1. По приведенному программному коду определить компонентный состав и функционал приложения.

Листинг программы

```
unit max1;
interface
uses
```

```
   Windows, Messages, SysUtils, Variants, Classes, Graphics,
Controls, Forms, Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    edt1: TEdit;
    edt2: TEdit;
    btn1: TButton;
    btn2: TButton;
    lbl1: TLabel;
    lbl2: TLabel;
    lbl3: TLabel;
    procedure FormActivate(Sender: TObject);
    procedure btn1Click(Sender: TObject);
    procedure btn2Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation
{$R *.dfm}

procedure TForm1.FormActivate(Sender: TObject);
begin
  lbl1.Caption:= 'Введите ...';
  lbl2.Caption:= 'Введите ...';
  btn1.Caption:= 'Старт';
  btn2.Visible:= False
end;

procedure TForm1.btn1Click(Sender: TObject);
begin
  btn2.Visible:= True;
  btn2.Caption:= 'Выход';
  if StrToFloat(edt1.Text) >= StrToFloat(edt2.Text)
    then lbl3.Caption:= edt1.Text
    else lbl3.Caption:= edt2.Text
end;
```

```
procedure TForm1.btn2Click(Sender: TObject);
begin
  Close
end;

end.
```

Задание 2. Описать вид окна при запуске приложения.
Задание 3. Если в приложении предусмотрен ввод числовых значений, указать их допустимые типы.
Задание 4. Описать процесс работы приложения и вид окна в конце его работы.
Задание 5. Дополнить код средствами проверки правильности ввода данных.

Задача 2.
Работа предназначена для выполнения без компьютера

Задание 1. По приведенному программному коду определить компонентный состав и функционал приложения.

Листинг программы

```
unit rg;
interface
uses
   Windows, Messages, SysUtils, Variants, Classes, Graphics,
Controls, Forms, Dialogs, StdCtrls, ExtCtrls;

type
  TForm1 = class(TForm)
    rg1: TRadioGroup;
    shp1: TShape;
    btn1: TButton;
    procedure rg1Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure btn1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation
```

```
{$R *.dfm}

procedure TForm1.FormActivate(Sender: TObject);
begin
  shp1.Visible:= False;
  btn1.Enabled:= False
end;

procedure TForm1.rg1Click(Sender: TObject);
var CurDate: TDate;
begin
  CurDate:= Date;
  btn1.Caption:= 'Выход';
  btn1.Enabled:= True;
  shp1.Visible:= True;
  case rg1.ItemIndex of
   0: Form1.Color:= clRed;
   1: Form1.Color:= clBlue;
   2: begin
       form1.Color:= clWhite;
       shp1.Shape:= stCircle;
       shp1.Brush.Style:= bsCross
      end;
   3: begin
       shp1.Brush.Style:= bsClear;
       shp1.Shape:= stRoundRect;
       shp1.Pen.Style:= psDot;
       shp1.Brush.Color:= clGreen
      end;
   4: ShowMessage('Сегодня ' + DateToStr(CurDate))
   end;
end;

procedure TForm1.btn1Click(Sender: TObject);
begin
  Close
end;

end.
```

Задание 2. Описать вид окна при запуске приложения.
Задание 3. Описать процесс работы приложения и вид окна в конце его работы.
Задание 4. Изменить вид окна приложения и код так, чтобы активация неактивных/невидимых объектов происходила в самостоятельной процедуре.

С2. Тематика курсовых проектов

1. Объектная модель MS PowerPoint

План

Глава 1 (теоретическая часть) – понятия классов и объектов; понятие и структура объектной модели; характеристика объектной модели MS PowerPoint.

Глава 2. (практическая часть) – пример использования объектной модели при разработке макроса в MS PowerPoint.

2. Разработка графического интерфейса пользователя в Delphi

План

Глава 1 (теоретическая часть) – понятие графического интерфейса; средства Delphi для разработки пользовательского интерфейса; алгоритм разработки графического интерфейса.

Глава 2. (практическая часть) – пример разработки графического интерфейса пользователя в Delphi.

3. Использование средств программирования на VBA для решения задач финансового анализа в MS Excel

План

Глава 1 (теоретическая часть) – средства MS Excel для решения задач финансового анализа; общая характеристика языка VBA; использование средств программирования для экономических и финансовых расчетов.

Глава 2 (практическая часть) – разработка приложения в MS Excel средствами VBA для решения практической задачи.

4. Разработка многодокументных приложений в Delphi

План

Глава 1 (теоретическая часть) – понятие однодокументных и многодокументных приложений, средства для работы с ними в Delphi.

Глава 2 (практическая часть) – разработка однодокументного и многодокументного приложения.

Блок D. Задания для использования в рамках промежуточной аттестации

D1.Перечень экзаменационных вопросов

1. Основные понятия объектно-ориентированного программирования
2. Структура проекта в Delphi
3. Базовые конструкции языка Delphi
4. Типы данных
5. Классы и объекты

6. Методы классов. Полиморфизм
7. Понятие события. Процедура обработки события
8. Структура файла модуля проекта
9. Обработка исключительных ситуаций. Классификация ошибок
19. Классы исключений
11. Понятие и виды подпрограмм
12. Создание и использование библиотек подпрограмм
13. Динамические структуры данных. Указатели
14. Динамические переменные, списки
15. Структура проекта DLL
16. Организация приложения в Delphi; однодокументное приложение
17. Многодокументные приложения
18. Одноэкземплярное приложение – понятие и особенности разработки
19. Объект Application
20. Объект Screen
21. Понятие потоков и процессов. Средства синхронизации потоков
22. Работа с графикой в среде Delphi. Классы TCanvas, TPen, TBrush, TFont
23. Компоненты Image, PaintBox, Shape
24. Простая анимация
25. Воспроизведение звука
26. Воспроизведение видеоклипов
27 Управление мультимедийными устройствами
28. Управляющие элементы ActiveX
29. Технология COM
30. Технология CASE
31. Понятие обратного программирования, области применения
32. Архитектура компьютерных систем
33. Структура памяти процессора
34. Ассемблер и язык ассемблера, основные понятия. Достоинства и недостатки
35. Регистры процессора, классификация
36. Синтаксис языка ассемблера, общая характеристика
37. Метки, переходы
38. Работа с числами в ассемблере
39. Знаковое и беззнаковое умножение
40. Целочисленное деление (div, idiv)
41. Циклы в ассемблере
42. Константы в ассемблере
43. Макросы в ассемблере
44. Режимы работы процессора
45. Уровни привилегий
46. Прерывания в защищенном режиме
47. Базовые понятия дизассемблирования
48. Основной алгоритм дизассемблирования
49. Классификация способов защиты

50. Преобразование кода программы – основные цели и способы организации

D2.Перечень экзаменационных задач

1. Разработать приложение, которое пересчитывает скорость из километров в час в метры в секунду
2. Разработать приложение для вычисления произведения и частного двух действительных чисел
3. Разработать приложение для вычисления произведения и частного двух случайных целых чисел
4. Разработать приложение, которое определяет, делится ли введенное число на 3 или на 5 и выводит соответствующее сообщение
5. Разработать приложение, которое проверяет, является введенный символ цифрой или буквой латинского алфавита и выводит соответствующее сообщение
6. Разработать приложение, в котором в зависимости от вводимого названия цвета появляется поле, окрашенное в этот цвет
7. Разработать приложение, в котором по экрану движется окружность
8. Разработать приложение, которое при нажатии на кнопку *Выполнить* выводит на экран прямоугольник, закрашенный случайным цветом
9. Напишите на ассемблере код функции, вычисляющей среднее арифметическое двух целых чисел
10. Напишите на ассемблере код функции, вычисляющей значение z по формуле $z=x^2-4x+1$
11. Напишите на ассемблере код функции, вычисляющей значение x по формуле $x=2a^2-\frac{b}{a}$
12. Напишите фрагмент кода на ассемблере, который меняет местами значения переменных a и b
13. Напишите фрагмент кода на ассемблере, который помещает большее из чисел a и b в регистр edx
14. Напишите фрагмент кода на ассемблере, который вычисляет произведение трех чисел и результат помещает в регистр ebx
15. Определите, какие действия выполняет функция

```
function f1(x, z: integer): integer;
asm
 mov ebx,z
 mov eax,x
 imul ebx
end;
```

16. Определите, какие действия выполняет функция

```
function f2(x, y: integer): integer;
asm
 mov edx,z
```

```
    mov eax,x
    idiv edx
  end;
```

Дисциплина «Технологии и методы программирования» предполагает выполнение курсового проекта/работы для систематизации, закрепления и углубления знаний, умений и навыков обучающихся в предметной области дисциплины.

Аттестация по курсовому проекту/работе производится в виде ее защиты в рамках промежуточного контроля обучающихся по данной дисциплине в 6 семестре в период теоретического обучения.

D3. Тематика курсовых проектов

1. Классы, объекты и коллекции в MS Office
2. Типы данных и управляющие структуры в Visual Basic
3. Объектные модели MS Word и MS Excel
4. Объектная модель MS PowerPoint
5. Структура объектных моделей приложений MS Office
6. Автоматизация работы с диаграммами в MS Excel средствами VBA
7. Объектная модель MS Outlook
8. Приложение «Калькулятор» в Delphi для работы с различными системами счисления
9. Инструментальные средства поддержки разработки кода в Delphi
10. Разработка графического интерфейса пользователя в Delphi
11. Разработка мультимедийных приложений
12. Обработка ошибок в Delphi и в Visual Basic
13. Использование средств программирования на VBA для решения задач финансового анализа в MS Excel
14. Использование средств программирования на VBA при работе со сводными таблицами в MS Excel.
15. Автоматизация работы с базами данных в среде MS Excel средствами офисного программирования
16. Средства для работы с файлами в Delphi
17. Разработка многодокументных приложений в Delphi
18. Работа с массивами в Visual Basic
19. Базовые конструкции языка C++
20. Работа с данными в C++

РАЗДЕЛ 3.Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Балльно-рейтинговая система является базовой системой оценивания сформированности компетенций обучающихся очной формы обучения.

Итоговая оценка сформированности компетенции(й) обучающихся в рамках балльно-рейтинговой системы осуществляется в ходе текущего контроля успеваемости, промежуточной аттестации и определяется как сумма баллов, полученных обучающимися в результате прохождения всех форм контроля.

Оценка сформированности компетенции(й) по дисциплине складывается из двух составляющих:

- ✓ первая составляющая – оценка преподавателем сформированности компетенции(й) в течение семестра в ходе текущего контроля успеваемости (максимум 100 баллов). Структура первой составляющей определяется технологической картой дисциплины, которая в начале семестра доводится до сведения обучающихся;
- ✓ вторая составляющая – оценка сформированности компетенции(й) обучающихся на экзамене (максимум – 30 баллов).

Для студентов очно-заочной формы обучения применятся 4-балльная шкала оценивания результатов текущего контроля успеваемости и промежуточной аттестации обучающихся.

Дисциплина «Технологии и методы программирования» предполагает выполнение курсового проекта для систематизации, закрепления и углубления знаний, умений и навыков обучающихся в предметной области дисциплины.

Аттестация по курсовому проекту производится в виде ее защиты в рамках промежуточного контроля обучающихся по данной дисциплине во 2 семестре в период теоретического обучения.

уровни освоения компетенций	продвинутый уровень	базовый уровень	пороговый уровень	допороговый уровень
100 – балльная шкала	85 и ≥	70 – 84	51 – 69	0 – 50
4 – балльная шкала	«отлично»	«хорошо»	«удовлетворительно»	«неудовлетворительно»

Шкала оценок при текущем контроле успеваемости по различным показателям

Показатели оценивания сформированности компетенций	***Баллы***	***Оценка***
Выполнение лабораторной работы	0-10	«неудовлетворительно»

		«удовлетворительно» «хорошо» «отлично»
Тестирование	0-30	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»
Выполнение письменной (контрольной) работы	0-30	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»
Ответ на коллоквиуме	0-30	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»
Решение кейс-задач	0-10	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»
Устный опрос	0-10	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»
Выполнение и защита реферата	0-5	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»
Подготовка и защита презентации	0-5	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»

Соответствие критериев оценивания уровню освоения компетенций по текущему контролю успеваемости

Баллы	***Оценка***	***Уровень освоения компетенций***	***Критерии оценивания***
0-50	«неудовлетвори	Допороговый	Обучающийся не приобрел знания,

	тельно»	уровень	умения и не владеет компетенциями в объеме, закрепленном рабочей программой дисциплины
51-69	«удовлетворительно»	Пороговый уровень	Не менее 50% заданий, подлежащих текущему контролю успеваемости, выполнены без существенных ошибок
70-84	«хорошо»	Базовый уровень	Обучающимся выполнено не менее 70% заданий, подлежащих текущему контролю успеваемости, или при выполнении всех заданий допущены незначительные ошибки; обучающийся показал владение навыками систематизации материала и применения его при решении практических заданий; задания выполнены без ошибок
85-100	«отлично»	Продвинутый уровень	Не менее 85% заданий, подлежащих текущему контролю успеваемости, выполнены самостоятельно и в требуемом объеме; обучающийся проявляет умение обобщать, систематизировать материал и применять его при решении практических заданий; задания выполнены с подробными пояснениями и аргументированными выводами

Шкала оценок по промежуточной аттестации

Наименование формы промежуточной аттестации	***Баллы***	***Оценка***
Экзамен	0-30	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»

Соответствие критериев оценивания уровню освоения компетенций по промежуточной аттестации обучающихся

Баллы	***Оценка***	***Уровень освоения компетенций***	***Критерии оценивания***
0-11	«неудовлетвори тельно»	Допороговый уровень	Обучающийся не приобрел знания, умения и не владеет компетенциями в объеме, закрепленном рабочей программой дисциплины; обучающийся не смог ответить на вопросы
12-19	«удовлетворите льно»	Пороговый уровень	Обучающийся дал неполные ответы на вопросы, с недостаточной аргументацией, практические задания выполнены не полностью, компетенции, осваиваемые в процессе изучения дисциплины сформированы не в полном объеме.
20-25	«хорошо»	Базовый уровень	Обучающийся в целом приобрел знания и умения в рамках осваиваемых в процессе обучения по дисциплине компетенций; обучающийся ответил на все вопросы, точно дал определения и понятия, но затрудняется подтвердить теоретические положения практическими примерами; обучающийся показал хорошие знания по предмету, владение навыками систематизации материала и полностью выполнил практические задания
26-30	«отлично»	Продвинутый уровень	Обучающийся приобрел знания, умения и навыки в полном объеме, закрепленном рабочей программой дисциплины; терминологический аппарат использован правильно; ответы полные, обстоятельные, аргументированные, подтверждены конкретными примерами; обучающийся проявляет умение

			обобщать, систематизировать материал и выполняет практические задания с подробными пояснениями и аргументированными выводами

РАЗДЕЛ 4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков, характеризующие этапы формирования компетенций

Описание процедуры выполнения лабораторной работы

Лабораторная работа выполняется в компьютерном классе в течение 1 пары (2 академических часа). Результатом выполнения лабораторной работы является работающее приложение в соответствии с заданием. Студент должен ответить на вопросы по выполнению задания, а также может выполнить дополнительные задания.

Методика оценивания выполнения лабораторной работы

Баллы	Оценка	Показатели	Критерии
9-10	«отлично»	1.Полнота и правильность выполнения заданий лабораторной работы 2. Правильность ответов на вопросы по содержанию работы; 3. Самостоятельность выполнения работы 4. Выпонение дополнительных заданий и т.д.	Правильно выполнены все задания лабораторной работы, дополнительные задания, даны верные ответы на вопросы преподавателя
6-8	«хорошо»		Правильно выполнены все задания лабораторной работы и 50% дополнительных заданий, даны верные ответы на вопросы преподавателя
4-5	«удовлетворитель-но»		Правильно выполнены более 70% заданий лабораторной работы, даны верные ответы на 50% вопросов преподавателя
0-3	«неудовлетвори-тельно»		Выполнено менее 70% заданий лабораторной работы, ответы на вопросы не даны

Описание процесса тестирования

Тестирование проводится в компьютерном классе под контролем преподавателя (в исключительных случаях – с домашнего компьютера) в СДО «Прометей». На тестирование отводится 50 минут. Каждый вариант теста включает 30 тестовых заданий.

Баллы	Оценка	Показатели	Критерии
27-30	«отлично»	1. Полнота выполнения тестовых заданий 2. Своевременность выполнения 3. Правильность ответов на вопросы	Выполнено 85-100% заданий предложенного теста, правильно выполнены все задания вычислительного характера
19-26	«хорошо»		Выполнено 70-84% заданий предложенного теста, допущены ошибки в 1-2 заданиях вычислительного характера и неточности в заданиях типа «многие из многих».
13-18	«удовлетворительно»		Выполнено 50-69% заданий предложенного теста, ошибки допущены в заданиях вычислительного характера и в заданиях типа «многие из многих».
0-12	«неудовлетворительно»		Выполнено менее 50% заданий предложенного теста.

Описание процесса выполнения письменной (контрольной, самостоятельной) работы

Письменная работа выполняется в аудитории под контролем преподавателя. На выполнение письменной работы отводится 80 минут. Каждый студент получает оригинальный вариант работы, программный код (фрагмент кода). Задания должны быть выполнены письменно без помощи сред программирования.

Методика оценивания выполнения письменной работы

Баллы	Оценка	Показатели	Критерии
27-30	«отлично»	1. Полнота ответов на вопросы и выполнения заданий. 2. Своевременность выполнения работы 3. Правильность ответов на вопросы 4. Правильность выполнения заданий	Правильно выполнены все задания, допустимы мелкие недочеты, но студент в полной мере демострирует понимание предложенного кода
19-26	«хорошо»		Студент демострирует понимание предложенного кода, но допускает негрубые ошибки в ответах не более, чем на 2 вопроса
13-18	«удовлетворительно»		Понимание кода неполное, неверные ответы на 2-3 вопроса
0-12	«неудовлетворительно»		Ответы на вопросы показывают непонимание студентом предложенного кода

Описание процедуры проведения коллоквиума

Коллоквиум проводится во внеучебное время, состоит в ответах студентов на вопросы и выполнение практических заданий в устной или письменной форме. Перечень вопросов и заданий к коллоквиуму сообщается студентам заранее не позднее, чем за 3 дня до времени проведения коллоквиума. Студент должен

ответить не менее чем на 2 теоретических вопроса и выполнить не менее 3 практических заданий.

Методика оценивания ответа на коллоквиуме

Баллы	Оценка	Показатели	Критерии
25-30	«отлично»	1. Правильность данных ответов 2. Полнота и аргументированность данных ответов 3. Правильность выполнения практических заданий 4. Ответы на дополнительные вопросы	Даны полные и аргументированные ответы на поставленные вопросы. Обнаружено понимание материала, может обосновать свои суждения, применить знания на практике, привести примеры. Все практические задания выполнены правильно.
18-24	«хорошо»		Студент обнаруживает понимание материала, может обосновать свои суждения. Выполнено не менее 2 практических заданий
12-17	«удовлетворительно»		Студент обнаруживает знание и понимание основных положений задания, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил, выполнено только 2 практических задания.
0-11	«неудовлетворительно»		Студент обнаруживает незнание ответов на теоретические вопросы и основных определений, в состоянии выполнить не более одного практического задания.

Описание процедуры выполнения кейс-задач

Кейс-задача выполняется студентом самостоятельно или в составе группы из 2-3 человек. В зависимости от сложности задачи, на ее решени может отводится от 2 академических часов (1 пара) до 2 дней. После завершения процесса решения студент (группа) должен в течение 5-7 минут изложить постановку задачи, алгоритм решения и результат.

Методика оценивания решения кейс-задач

Баллы	Оценка	Показатели	Критерии
9-10	«отлично»	1. Полнота решения кейс-задачи 2. Анализ других	Основные требования к решению кейс-задач выполнены. Продемонстрировано умение анализировать ситуацию и возможные

		решений 3. Своевременность выполнения 4. Правильность ответов на вопросы	решения, выбирать оптимальное решение; умение работать с информацией, в том числе – находить дополнительную информацию, необходимую для уточнения ситуации. Продемонстрированы навыки четкого и точного изложения собственной точки зрения в устной и письменной форме.
6-8	«хорошо»		Основные требования к решению кейс-задач выполнены, но при этом допущены недочеты. В частности, недостаточно раскрыты навыки критического оценивания альтерна-тивных решений, недостаточно обоснован выбор собственного решения
4-5	«удовлетворитель-но»		Имеются существенные отступления от требований к решению кейс-задач. В частности, отсутствуют навыки анализа альтернативных решений и выбора оптимального решения, выбранный алгоритм приводит к некорректному решению, план действий недостаточно детализирован или отсутствует, хотя решение получено.
0-3	«неудовлетвори-тельно»		Кейс-задача не решена, обнаруживается существенное непонимание проблемы.

Описание процедуры проведения устного опроса

Устный опрос проводится во время практических занятий по заранее подготовленным вопросам, которые выбираются преподавателем к соответствующей теме. Количество вопросов определяется преподавателем и зависит от темы, количества студентов в группе и т.д.

Методика оценивания ответов на устные вопросы

Баллы	Оценка	Показатели	Критерии
9-10	«отлично»	1. Правильность данных ответов 2. Полнота и аргументированность данных ответов 3. Количество вопросов, на которые ответил студент	Даны полные и аргументированные ответы на поставленные вопросы. Обнаружено понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только по учебнику, но и самостоятельно составленные. Изложение материала последовательно и правильно.

6-8	«хорошо»		Студент дает ответ, удовлетворяющий тем же требованиям, что и для оценки «отлично», но приводит недостаточно убедительные примеры, не очень удачно связывает теорию с практикой.
4-5	«удовлетвори-тельно»		Студент обнаруживает знание и понимание основных положений задания, но: 1) излагает материал неполно и допускает неточности в определении понятий или формулировке правил; 2) не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; 3) излагает материал непоследовательно и допускает ошибки.
0-3	«неудовлетво-рительно»		Студент обнаруживает незнание ответа на вопрос, допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал; отмечаются такие недостатки в подготовке студента, которые являются серьезным препятствием к успешному овладению последующим материалом.

Описание процедуры выполнения и защиты реферата

Реферат выполняется студентом самостоятельно в соответствии с темой, предложенной преподавателем. Подготовка реферата заключается в поиске необходимой информации, ее анализе и составлении соответствующего текста. К защите реферата желательно подготовить презентацию.

Процедура защиты реферата состоит в изложении в течение 5-7 минут содержания реферата в присутствии преподавателя и дргих студентов группы и ответах на вопросы.

Методика оценивания выполнения рефератов

Баллы	Оценка	Показатели	Критерии
5	«отлич-но»	1. Соответствие теме собранного материала 2. Полнота раскрытия	Собранный материал соответствует теме, тема полностью раскрыта, есть электронная презентация, даны ответы на все вопросы

3-4	«хорошо»	темы 3. Наличие электронной презентации 4. Правильность и полнота ответов на вопросы	Собранный материал соответствует теме, тема раскрыта, даны ответы на большую часть вопросов
1-2	«удовлетворительно»		Тема реферата раскрыта в достаточной степени, даны ответы на некоторые вопросы
0	«неудовлетворительно»		Тема реферата не раскрыта или раскрыта не полностью, студент не может ответить на вопросы

Описание процедуры выполнения и представления презентации

Презентация выполняется студентом самостоятельно в соответствии с темой, предложенной преподавателем. Подготовка презентации заключается в поиске необходимой информации, ее анализе, подготовке соответствующего иллюстративного материала и сопровождающего текста.

Процедура представления презентации состоит в демонстрации презентации и ответах на вопросы по ее содержанию.

Методика оценивания выполнения презентаций

Баллы	Оценка	Показатели	Критерии
5	«отлично»	1. Соответствие теме собранного материала 2. Полнота раскрытия темы 3. Самостоятель-ность выполнения презентации 4. Правильность и полнота ответов на вопросы	Собранный материал соответствует теме, тема полностью раскрыта, даны ответы на все вопросы
3-4	«хорошо»		Собранный материал соответствует теме, тема раскрыта, даны ответы на большую часть вопросов
1-2	«удовлетворительно»		Тема презентации раскрыта в достаточной степени, даны ответы на некоторые вопросы
0	«неудовлетворительно»		Тема презентации не раскрыта или раскрыта не полностью, студент не может ответить на вопросы, использована готовая презентация

Описание процедуры проведения экзамена

Экзамен проводится в компьютерном классе в устной или письменной форме. Билет содержит 2 теоретических вопроса и 1 практическое задание. Время на подготовку – 45 минут.

Методика оценивания ответа на экзамене

Баллы	Оценка	Показатели	Критерии
26-30	«отлично»	1. Полнота изложения теоретического материала 2. Правильность и	Дан полный, в логической последовательности развернутый ответ на поставленный вопрос, где он продемонстрировал знания

		аргументированность изложения 3. Полнота и правильность решения практического задания 4. Самостоятельность ответа 5. Культура речи	предмета в полном объеме учебной программы, достаточно глубоко осмысливает дисциплину, самостоятельно и правильно отвечает на дополнительные вопросы, приводит собственные примеры. Практические задания выполнены без ошибок.
20-25	«хорошо»		Дан развернутый ответ на поставленный вопрос, где студент демонстрирует знания, приобретенные на лекционных и семинарских занятиях, а также полученные посредством изучения обязательных учебных материалов по курсу, дает аргументированные ответы, приводит примеры, в ответе присутствует свободное владение монологической речью, логичность и последовательность ответа. Однако допускается неточность в ответе. Практические задания выполнены с небольшими неточностями.
12-19	«удовлетворитель-но»		Дан ответ, свидетельствующий о знании основных положений изучаемой дисциплины, отличающийся недостаточной глубиной и полнотой раскрытия темы, недостаточным умением давать аргументированные ответы и приводить примеры, недостаточно свободным владением монологической речью. Допускается несколько ошибок в содержании ответа и решении практических заданий.
0-11	«неудовлетвори-тельно»		Дан ответ, который содержит ряд серьезных неточностей, обнаруживающий незнание основных определений и положений изучаемой дисциплины. Практическое задание выполнено с серьезными ошибками. Наводящие и

			дополнительные вопросы ситуацию не улучшают.

Описание процедуры подготовки и защиты курсовых проектов

Курсовой проект выполняется студентом самостоятельно по теме, предоставляемой преподавателем либо по назначению кафедры.

Процедура защиты курсового проекта состоит в изложении в течение 5-7 минут содержания проекта в присутствии преподавателя и дргих студентов группы и ответах на вопросы.

Методика оценивания курсовых проектов

Оценка	Показатели	Критерии
«отлично»	1. Полнота раскрытия темы курсового проекта 2. Своевременность выполнения курсового проекта 3. Оформление курсового проекта 4. Результаты, полученные автором курсового проекта	Во введении приводится обоснование выбора темы, полностью раскрыта ее актуальность, четко определены и грамотно поставлены цель и задачи курсового проекта. Если проект носит реферативный характер, основная часть демонстрирует большое количество изученных автором источников и их анализ. Присутствуют выводы и грамотные обобщения. Если проект предусматривает разработку приложения, основная часть содержит алгоритм разработки, а приложение – полный программный код. Курсовой проект написан в стиле академического письма (использован научный стиль изложения материала.) Автор адекватно применял терминологию, правильно оформил ссылки. Оформление работы соответствует требованиям, библиография, приложения оформлены на отличном уровне. Объем работы соответствует требованиям.
«хорошо»		Во введении цель и задачи сформулированы недостаточно четко. Тема раскрыта, но есть погрешности в офрмлении, разработанное приложение работает, но не весь требуемый функционал реализован. В случае реферативного проекта – выводы не вполне соответствуют изложенному материалу.
«удовлетворительно»		Введение содержит лишь попытку обоснования выбора темы и актуальности, отсутствуют четкие формулировки. Нечетко определены цель и задачи. В основнойчасти не сделаны самостоятельные выводы, либо приложение не доведено до рабочего состояния. Стиль изложения не соответствует научному, есть грубые ошибки в оформлении.
«неудовлетворительно»		Введение не содержит обоснования актуальности темы. Тема не раскрыта, приложение не реализовано, оформление не соответствует требованиям

Лист актуализации оценочных материалов по дисциплине «Технологии и методы программирования»

Оценочные материалы пересмотрены,
обсуждены и одобрены на заседании кафедры

Протокол от «_____»_____ 20___ г. №_____

Зав. кафедрой _____

Оценочные материалы пересмотрены,
обсуждены и одобрены на заседании кафедры

Протокол от «_____»_____ 20___ г. №_____

Зав. кафедрой _____

Оценочные материалы пересмотрены,
обсуждены и одобрены на заседании кафедры

Протокол от «_____»_____ 20___ г. №_____

Зав. кафедрой _____

Оценочные материалы пересмотрены,
обсуждены и одобрены на заседании кафедры

Протокол от «_____»_____ 20___ г. №_____

Зав. кафедрой _____