

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Цыбиков Бадмацэ Батзориг
Должность: Ректор
Дата подписания: 15.09.2024 20:21:34
Уникальный программный ключ:
056af948c3e48c6f3c571e429957a8ae7b757ae8

**Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Бурятская государственная сельскохозяйственная академия
имени В.Р. Филиппова»**

Экономический факультет

СОГЛАСОВАНО
Заведующий
выпускающей кафедрой
Информатика и
информационные
технологии в экономике

уч. ст., уч. зв.

ФИО

подпись

«__» _____ 20__ г.

УТВЕРЖДАЮ
Декан экономического
факультета

уч. ст., уч. зв.

ФИО

подпись

«__» _____ 20__ г.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

дисциплины (модуля)

Б1.В.08 Объектно-ориентированное программирование

**Направление подготовки
09.03.03 Прикладная информатика**

**Направленность (профиль) Прикладная информатика в экономике АПК
бакалавр**

Обеспечивающая
преподавание дисциплины
кафедра

Информатика и информационные технологии в
экономике

Разработчик (и)

подпись

уч.ст., уч. зв.

И.О.Фамилия

подпись

уч.ст., уч. зв.

И.О.Фамилия

Внутренние эксперты:
Председатель методической
комиссии экономического
факультета

подпись

уч.ст., уч. зв.

И.О.Фамилия

Заведующий методическим
кабинетом УМУ

подпись

И.О.Фамилия

Улан – Удэ, 2022

ВВЕДЕНИЕ

1. Оценочные материалы по дисциплине (модулю) являются обязательным обособленным приложением к Рабочей программе дисциплины (модуля) и представлены в виде оценочных средств.
2. Оценочные материалы является составной частью нормативно-методического обеспечения системы оценки качества освоения обучающимися указанной дисциплины (модуля).
3. При помощи оценочных материалов осуществляется контроль и управление процессом формирования обучающимися компетенций, из числа предусмотренных ФГОС ВО в качестве результатов освоения дисциплины (модуля).
4. Оценочные материалы по дисциплине (модулю) включают в себя:
 - оценочные средства, применяемые при промежуточной аттестации по итогам изучения дисциплины (модуля).
 - оценочные средства, применяемые в рамках индивидуализации выполнения, контроля фиксированных видов ВАРО;
 - оценочные средства, применяемые для текущего контроля;
5. Разработчиками оценочных материалов по дисциплине (модулю) являются преподаватели кафедры, обеспечивающей изучение обучающимися дисциплины (модуля) в Академии. Содержательной основой для разработки оценочных материалов является Рабочая программа дисциплины (модуля).

1. ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ ИЗУЧЕНИЯ
учебной дисциплины (модуля), персональный уровень достижения которых проверяется
с использованием представленных в п. 3 оценочных материалов

Компетенции, в формировании которых задействована дисциплина		Код и наименование индикатора достижений компетенции	Компоненты компетенций, формируемые в рамках данной дисциплины (как ожидаемый результат ее освоения)		
код	наименование		знать и понимать	уметь делать (действовать)	владеть навыками (иметь навыки)
1		2	3	4	5
Профессиональные компетенции					
ПКС-2	Способность разрабатывать и адаптировать прикладное программное обеспечение	ИД-1 _{ПКС-2} Знает языки программирования и приемы работы с базами данных ИД-2 _{ПКС-2} Умеет кодировать на языках программирования ИД-3 _{ПКС-2} Владеет навыками разработки прототипа ИС в соответствии с требованиями	Знает языки программирования и приемы работы с базами данных	Умеет кодировать на языках программирования	Владеет навыками разработки прототипа ИС в соответствии с требованиями

2. РЕЕСТР

**элементов оценочных материалов по дисциплине (модулю) / практике
(в том числе, вставить в соответствии с 3 и 5 разделами РП)**

Группа оценочных средств	Оценочное средство или его элемент
1	2
1. Средства для промежуточной аттестации по итогам изучения дисциплины	Вопросы для проведения итогового контроля (зачета)
	Критерии оценки к зачету
2. Средства для индивидуализации выполнения, контроля фиксированных видов (ВАРО)	Комплект заданий для самостоятельной работы обучающихся
	Критерии оценки самостоятельной работы обучающихся
	Шкала оценивания
	.
3. Средства для текущего контроля	Контрольные вопросы для проведения устных опросов
	Критерии оценки проведения устных опросов
	Шкала оценивания
	Комплект заданий для лабораторных работ
	Критерии оценки лабораторных работ
	Шкала оценивания
	Кейс-задания
	Критерии оценки кейс-заданий
	Шкала оценивания
	Комплект заданий для самостоятельной работы обучающихся
	Критерии оценки самостоятельной работы обучающихся
	Шкала оценивания
	Комплект тестовых заданий
	Критерии оценки тестовых заданий
Шкала оценивания	

3. Описание показателей, критериев и шкал оценивания компетенций в рамках дисциплины (модуля)

Код и название компетенции	Код индикатора достижений компетенции	Индикаторы компетенции	Показатель оценивания – знания, умения, навыки (владения)	Уровни сформированности компетенций				Формы и средства контроля формирования компетенций
				компетенция не сформирована	минимальный	средний	высокий	
				Оценки сформированности компетенций				
				2	3	4	5	
				Оценка «неудовлетворительно»	Оценка «удовлетворительно»	Оценка «хорошо»	Оценка «отлично»	
				Характеристика сформированности компетенции				
			Компетенция в полной мере не сформирована. Имеющихся знаний, умений и навыков недостаточно для решения практических (профессиональных) задач	Сформированность компетенции соответствует минимальным требованиям. Имеющихся знаний, умений, навыков в целом достаточно для решения практических (профессиональных) задач	Сформированность компетенции в целом соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в целом достаточно для решения стандартных (профессиональных) задач	Сформированность компетенции полностью соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в полной мере достаточно для решения сложных практических (профессиональных) задач		
Критерии оценивания								
ПКС-2 Способность разрабатывать и адаптировать прикладное программное обеспечение	ИД-1 _{ПКС-2}	Полнота знаний	Знает языки программирования и приемы работы с базами данных	Не знает языки программирования и приемы работы с базами данных	Знает частично языки программирования и приемы работы с базами данных	Знает хорошо языки программирования и приемы работы с базами данных	Знает в полном объеме языки программирования и приемы работы с базами данных	Перечень вопросов к зачету, комплект контрольных вопросов для проведения устных опросов, комплект заданий для лабораторных работ, комплект заданий для самостоятельной работы обучающихся, комплект тестовых заданий
	ИД-2 _{ПКС-2}	Наличие умений	Умеет кодировать на языках программирования	Не умеет кодировать на языках программирования	Умеет частично кодировать на языках программирования	Умеет хорошо кодировать на языках программирования	Умеет достаточно кодировать на языках программирования	
	ИД-3 _{ПКС-2}	Наличие навыков (владение опытом)	Владеет навыками разработки прототипа ИС в соответствии с требованиями	Не владеет навыками разработки прототипа ИС в соответствии с требованиями	Владеет частично навыками разработки прототипа ИС в соответствии с требованиями	Владеет хорошо навыками разработки прототипа ИС в соответствии с требованиями	Владеет в самостоятельно навыками разработки прототипа ИС в соответствии с требованиями	

4. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения основной профессиональной образовательной программы

4.1. Типовые контрольные задания, необходимые для оценки знаний, умений, навыков

4.1.1. Средства для промежуточной аттестации по итогам изучения дисциплины

Нормативная база проведения промежуточной аттестации обучающихся по результатам изучения дисциплины: Б1.В.09 Объектно-ориентированное программирование	
1) действующее «Положение о текущем контроле успеваемости и промежуточной аттестации обучающихся ФГБОУ ВО Бурятская ГСХА»	
Основные характеристики промежуточной аттестации обучающихся по итогам изучения дисциплины	
1	2
Цель промежуточной аттестации -	установление уровня достижения каждым обучающимся целей и задач обучения по данной дисциплине, изложенным в п.2.2 настоящей программы
Форма промежуточной аттестации -	зачёт
Место процедуры получения зачёта в графике учебного процесса	1) участие обучающегося в процедуре получения зачёта осуществляется за счёт учебного времени (трудоемкости), отведённого на изучение дисциплины 2) процедура проводится в рамках ВАРО, на последней неделе семестра
Основные условия получения обучающимся зачёта:	1) обучающийся выполнил все виды учебной работы (включая самостоятельную) и отчитался об их выполнении в сроки, установленные графиком учебного процесса по дисциплине
Процедура получения зачёта -	Представлены в оценочных материалах по данной дисциплине
Методические материалы, определяющие процедуры оценивания знаний, умений, навыков:	

Перечень вопросов к зачету

1. Объектно-ориентированное программирование (ПКС-2)
2. Основные принципы ООП. (ПКС-2)
3. Основные этапы решения задач на компьютере. (ПКС-2)
4. Язык программирования OBJECT PASCAL. (ПКС-2)
5. Основы визуального программирования. (ПКС-2)
6. Класс. Понятие класса в ООП. (ПКС-2)
7. Объект. Понятие объекта в ООП. (ПКС-2)
8. Поле. Понятие поля в среде ООП. (ПКС-2)
9. Свойства объекта. Классификация. (ПКС-2)
10. Метод. Описание метода. (ПКС-2)
11. События. Описание событий. (ПКС-2)
12. Форма. Свойства формы. (ПКС-2)
13. Среда разработчика Delphi. (ПКС-2)
14. Структура проекта Delphi. (ПКС-2)
15. Структура модуля. (ПКС-2)
16. Состав проекта Delphi. Сохранение проекта. (ПКС-2)
17. Компиляция. Ошибки. Предупреждения и подсказки. (ПКС-2)
18. Компоненты Delphi. Вкладка Standart. (ПКС-2)
19. Компоненты Delphi. Вкладка Additional. (ПКС-2)
20. Идентификаторы. Константы. (ПКС-2)
21. Переменные. Выражения. (ПКС-2)
22. Операции в среде ООП. (ПКС-2)
23. Типы данных в среде ООП. (ПКС-2)
24. Стандартные процедуры и функции. (ПКС-2)
25. Функции преобразования. (ПКС-2)
26. Операторы. (ПКС-2)
27. Оператор условия. (ПКС-2)
28. Операторы цикла. (ПКС-2)
29. Процедуры и функции(ПКС-2)
30. Подпрограммы в среде ООП. (ПКС-2)

4.1.2. Средства

для индивидуализации выполнения, контроля фиксированных видов ВАРО

Фиксированные виды ВАРО не предусмотрены

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

5.2. Критерии оценки к зачету

зачет (86-100 баллов) ставится обучающемуся, обнаружившему систематические и глубокие знания учебно-программного материала, умения свободно выполнять задания, предусмотренные программой в типовой ситуации (с ограничением времени) и в нетиповой ситуации, знакомство с основной и дополнительной литературой, усвоение взаимосвязи основных понятий дисциплины в их значении приобретаемой специальности и проявившему творческие способности и самостоятельность в приобретении знаний.

зачет (71-85 баллов) ставится обучающемуся, обнаружившему полное знание учебно-программного материала, успешное выполнение заданий, предусмотренных программой в типовой ситуации (с ограничением времени), усвоение материалов основной литературы, рекомендованной в программе, способность к самостоятельному пополнению и обновлению знаний в ходе дальнейшей работы над литературой и в профессиональной деятельности.

зачет (56-70 баллов) ставится обучающемуся, обнаружившему знание основного учебно-программного материала в объеме, достаточном для дальнейшей учебы и предстоящей работы по специальности, знакомство с основной литературой, рекомендованной программой, умение выполнять задания, предусмотренные программой.

незачет (менее 56 баллов) ставится обучающемуся, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий, слабые побуждения к самостоятельной работе над рекомендованной основной литературой. Оценка «неудовлетворительно» ставится обучающимся, которые не могут продолжить обучение или приступить к профессиональной деятельности по окончании академии без дополнительных занятий по соответствующей дисциплине.

6. Оценочные материалы для организации текущего контроля успеваемости обучающихся

Форма, система оценивания, порядок проведения и организация *текущего контроля успеваемости* обучающихся устанавливаются Положением об организации текущего контроля успеваемости обучающихся.

Комплект контрольных вопросов для проведения устных опросов

1. Основная идея объектно-ориентированного подхода.
2. Модели в объектно-ориентированном подходе.
3. Микро- и макро- процессы проектирования.
4. Микропроцесс проектирования: определение и этапы.
5. Макропроцесс проектирования: определение и этапы.
6. Принципы объектно-ориентированного подхода.
7. Принцип абстрагирования. Пример.
8. Принцип иерархии.
9. Принцип иерархии - наследование. Пример.
10. Принцип иерархии - агрегации. Пример.
11. Принцип инкапсуляции. Пример.
12. Принцип модульности.
13. Принцип типизации.
14. Принцип типизации - полиморфизм. Пример.
15. Принцип параллелизма.
16. Принцип сохраняемости.
17. Понятия: объект, класс. Пример.
18. Категории объектов. Пример.
19. Понятие компонента. Пример.
20. Понятие свойства. Пример.
21. Понятие метода. Пример.
22. Понятие события. Пример.
23. Виды свойств. Пример обращения к сложному свойству.
24. Описание класса и объекта. Пример.
25. Связь объекта и обработчика его события (схема).
26. Сходство и различие объекта, класса и компонента.
27. Интегрированная среда Borland Delphi.
28. Визуальное проектирование в Borland Delphi.
29. Программирование в Borland Delphi.

30. Структура интегрированной среды и назначение основных элементов.
31. Интегрированная среда Borland Delphi: инспектор объекта.
32. Интегрированная среда Borland Delphi: дерево просмотра объектов.
33. Интегрированная среда Borland Delphi: окно редактирования кода и проектировщик формы.
34. Интегрированная среда Borland Delphi: главное меню.
35. Интегрированная среда Borland Delphi: палитра компонентов.
36. Палитра компонентов: определение и структура.

Критерии оценивания

- правильность ответа по содержанию задания (учитывается количество и характер ошибок при ответе);
- полнота и глубина ответа (учитывается количество усвоенных фактов, понятий и т.п.);
- сознательность ответа (учитывается понимание излагаемого материала);
- логика изложения материала (учитывается умение строить целостный, последовательный рассказ, грамотно пользоваться специальной терминологией);
- использование дополнительного материала;
- рациональность использования времени, отведенного на задание (не одобряется затянутость выполнения задания, устного ответа во времени, с учетом индивидуальных особенностей обучающихся).

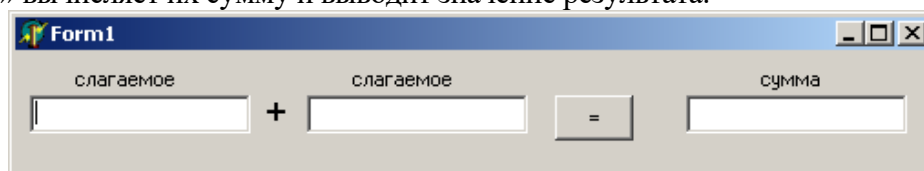
Шкала оценивания

Баллы для учета в рейтинге (оценка)	Степень удовлетворения критериям
86-100 баллов «отлично»	Обучающийся полно и аргументировано отвечает по содержанию вопроса (задания); обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только по учебнику, но и самостоятельно составленные; излагает материал последовательно и правильно.
71-85 баллов «хорошо»	Обучающийся достаточно полно и аргументировано отвечает по содержанию вопроса (задания); обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только по учебнику, но и самостоятельно составленные; излагает материал последовательно. Допускает 1-2 ошибки, исправленные с помощью наводящих вопросов.
56-70 баллов «удовлетворительно»	Обучающийся обнаруживает знание и понимание основных положений данного задания, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки.
менее 56 баллов «неудовлетворительно»	Обучающийся обнаруживает незнание ответа на соответствующее задание (вопрос), допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал. Отмечаются такие недостатки в подготовке обучающегося, которые являются серьезным препятствием к успешному овладению последующим материалом.

Комплект заданий для лабораторных работ

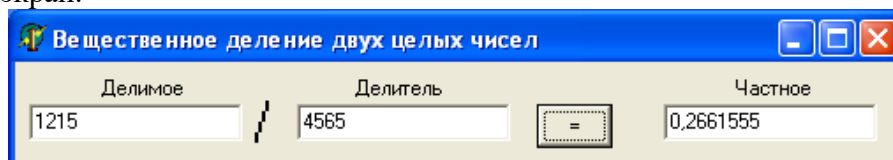
Задание 1.

Создать приложение, которое обеспечивает ввод двух целых чисел, по щелчку на кнопке с символом « \Rightarrow » вычисляет их сумму и выводит значение результата.



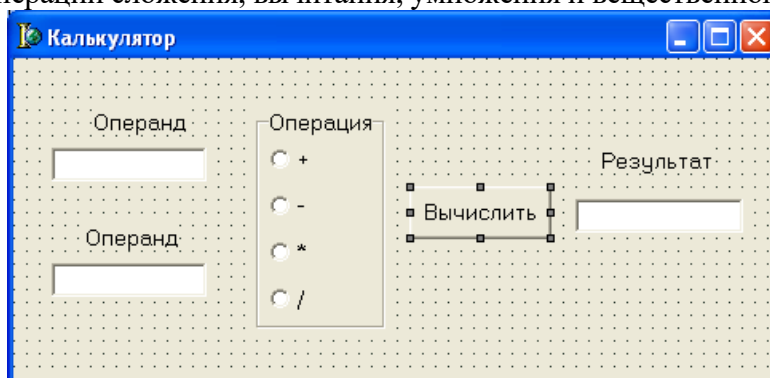
Задание 2.

Создайте приложение, реализующее ввод двух целых чисел, по щелчку на кнопке с символом « \Leftarrow » вычисляющее результат операции вещественного деления и выводящее значение результата на экран.



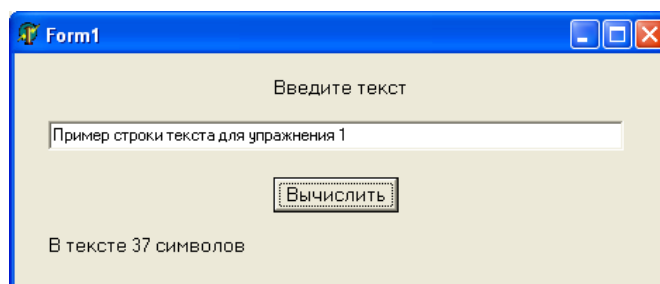
Задание 3.

Создайте приложение, обеспечивающее ввод двух целых чисел и выполнения над ними арифметических операций сложения, вычитания, умножения и вещественного деления.



Задание 4.

Создайте приложение, предлагающее пользователю ввести строку, определяющее количество символов в ней и выводящее результат, согласно окончанию слова «символов» с числом символов, например, «В тексте 1 символ», «В тексте 32 символа», «В тексте 47 символов».



Критерии оценивания:

- правильность выполнения задания на лабораторную работу в соответствии с вариантом;
- степень усвоения теоретического материала по теме лабораторной работы;
- способность продемонстрировать преподавателю навыки работы в инструментальной программной среде, а также применить их к решению типовых задач, отличных от варианта задания;
- качество подготовки отчета по лабораторной работе;
- правильность и полнота ответов на вопросы преподавателя при защите работы.

Шкала оценивания

Баллы для учета в рейтинге (оценка)	Степень удовлетворения критериям
86-100 баллов «отлично»	Выполнены все задания практической работы, обучающийся четко и без ошибок ответил на все контрольные вопросы
71-85 баллов «хорошо»	Выполнены все задания практической работы; обучающийся ответил на все контрольные вопросы с замечаниями
56-70 баллов «удовлетворительно»	Выполнены все задания практической работы с замечаниями; обучающийся ответил на все контрольные вопросы с замечаниями
менее 56 баллов «неудовлетворительно»	Обучающийся не выполнил или выполнил неправильно задания практической работы; обучающийся ответил на контрольные вопросы с ошибками или не ответил на контрольные вопросы

Кейс-задания

1) КЕЙС-ЗАДАНИЕ "СТРУКТУРНЫЙ АНАЛИЗ ПРОГРАММНОЙ СИСТЕМЫ"

Цель кейс-задания: ознакомиться с базовыми принципами структурного анализа программных систем и получить практические навыки его использования для конкретной предметной области.

Типовой вариант кейс-задания:

Задание: провести структурный анализ и построить DFD- диаграммы для заданной предметной области. Одна диаграмма должна представлять обобщенную работу системы, одна – детализацию одного из процессов (обе диаграммы должны включать 5-7 процессов). Для потоков данных, передаваемых между процессами, составить словарь данных. Построить STD-диаграмму системы.

1. Система коллективного редактирования документов. Система должна предоставлять возможность группе зарегистрированных пользователей выполнять совместное редактирование текстового документа. Исправления, внесенные каждым из пользователей (добавление, удаление, исправление фрагментов текста) отображаются отдельным цветом. Фиксация изменений в окончательном или промежуточном варианте документа разрешена одному из пользователей со статусом редактора.

2. Система резервного копирования файлов. Система должна по заданию пользователя осуществлять резервное копирование файлов из заданных папок в указанное хранилище по составленному расписанию. Система функционирует как многопользовательская, каждый пользователь может составить собственный список файлов и папок для копирования и расписание.

3. Система автоматизации обработки заявок клиентов на обслуживание средств ИТ. Клиенты имеют возможность оставлять заявки на обслуживание вычислительной техники, оператор принимает заявку, фиксирует ее в журнале, сообщают клиенту о результатах обработки заявки. Техники принимают свободные заявки, делают отметки о выполнении по окончании работ. Система должна обеспечивать: фильтрацию, сортировку заявок, формировать отчеты.

Контрольные вопросы.

1. Перечислите основные этапы жизненного цикла программного обеспечения.
2. Какие модели жизненного цикла ПО вам известны, каковы их основные отличительные особенности?
3. Какие методологии структурного анализа программных систем вам известны?
4. Что характеризует структурный подход к анализу программных систем? Какие основные диаграммы описывают модель ПО при структурном анализе в технологии 3VM?
5. С какой стороны описывает программную систему DFD-диаграмма? Какие основные элементы она использует?
6. Что такое ERD диаграмма? Для каких целей она включается в общую модель структурного анализа программной системы?
7. Назначение STD-диаграммы. Структура STD диаграммы.
8. Расскажите о результатах анализа программной системы из своего варианта.

2) КЕЙС-ЗАДАНИЕ "ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ АНАЛИЗ ПРОГРАММНОЙ СИСТЕМЫ"

Цель кейс-задания: ознакомиться с базовыми принципами объектно-ориентированного анализа программных систем и получить практические навыки его использования для конкретной предметной области.

Типовой вариант кейс-задания:

Задание: провести объектно-ориентированный анализ и построить для заданной предметной области диаграмму вариантов использования (Use Case), диаграмму классов, диаграмму последовательности и диаграмму состояний для одного из прецедентов.

1. Система автоматизированного тестирования. Преподаватель имеет возможности: вести базу вопросов, сформировать из них тесты, подтвердить регистрацию студента, составить список группы, просмотреть результаты тестирования. Студент может: зарегистрироваться в системе, пройти тест, просмотреть результаты тестирования.

2. Система электронных online-конференций. По инициативе одного из зарегистрированных пользователей создается конференция, в которой могут участвовать другие зарегистрированные организатором пользователи. Организатор может продемонстрировать медиаконтент участникам конференции, а затем в режиме chat'a ответить на их вопросы.

3. Система электронных аукционов. Зарегистрированные пользователи могут выставить лот на продажу, назначив начальную стоимость и дату торгов. Лот может снять с торгов сам продавец или администратор. Любой зарегистрированный пользователь может записаться на участие в торгах по лоту. Торги идут по схеме повышения стоимости в автоматическом режиме. Лот достается предложившему наибольшую цену.

4. Антивирус-ревизор. Система фиксирует состояние заданных пользователем папок и в случае изменения их состояния (количество файлов, размер, дата создания или модификации, атрибуты) отмечает измененные элементы файловой системы как подозрительные. Проверка состояния папок осуществляется либо по расписанию, либо команде пользователя.

5. Свободная электронная библиотеки. Зарегистрированный пользователь имеет возможность добавить новый файл-книгу в базу данных библиотеки, а также получить доступ к другим книгам. В случае платного доступа или в обмен на новую книгу пользователь может скачать файл с книгой, иначе – только читать с экрана. Библиотека предоставляет возможность поиска книг в библиотеке по различным критериям.

Контрольные вопросы.

1. Перечислите основные принципы объектно-ориентированной парадигмы разработки программного обеспечения. Дайте описание этим принципам.

2. Что такое UML? Для каких целей он предназначен?

3. Перечислите основные сущности, используемые на диаграммах UML.

4. Какие отношения можно выделить между сущностями UML?

5. Что общего и в чем разница отношений агрегации и композиции?

6. Для чего предназначены диаграммы UML? Какие типы диаграмм вам известны?

7. Диаграммы Use Case: назначение, принципы построения, примеры. Опишите Use Case диаграммы, построенные вами для своей задачи.

8. Для чего используются диаграммы классов? Опишите диаграмму классов, построенную вами для своей задачи.

9. Как строится и для чего предназначена диаграмма последовательности. Опишите диаграмму последовательности для прецедента из своей задачи.

10. Расскажите о назначении диаграмм состояний. Какие обозначения они используют? Какую сторону функционирования программной системы иллюстрируют?

3) кейс-задание "ЗНАКОМСТВО С ОБЪЕКТНО-ОРИЕНТИРОВАННЫМИ ЯЗЫКАМИ ПРОГРАММИРОВАНИЯ. РАЗРАБОТКА ПРОГРАММ С ИСПОЛЬЗОВАНИЕМ ОБЪЕКТОВ"

Цель кейс-задания: приобрести навыки программирования в объектно-ориентированном стиле: определять и создавать объекты, вызывать их методы.

Типовой вариант кейс-задания:

1. Определить класс «Многочлен», описывающий алгебраический многочлен от одной переменной, задаваемой степенью многочлена и массивом коэффициентов. Реализовать в классе методы, позволяющие вычислять значения многочлена для заданного аргумента, выполнять операции сложения, вычитания и умножения многочленов, взятия производной произвольного порядка с получением нового объекта-многочлена. В программе продемонстрировать использование объектов класса «Многочлен»

2. Определить класс «Комплексное число». Реализовать в виде класса методы для выполнения следующих операций над комплексными числами:

- сложение;
- вычитание;
- умножение;
- деление;
- модуль комплексного числа;
- возведение комплексного числа в степень n (n — натуральное).

В программе продемонстрировать использование объектов класса «Комплексное число»

3. Реализовать класс «Число в P -ичной системе счисления». Основание системы счисления должно быть компонентным данным класса. Реализовать в виде методов следующие операции над натуральными числами в P -ичной системе счисления ($2 \leq P \leq 9$):

- сложение;
- вычитание;
- умножение;
- деление;
- перевод из десятичной системы счисления в P -ичную;
- перевод из P -ичной системы счисления в десятичную;
- проверка правильности записи числа в P -ичной системе счисления;

В программе продемонстрировать использование объектов класса «Число в P -ичной системе счисления»

4. Реализовать в виде класса «Обыкновенная дробь», описывающий обыкновенные дроби вида P/Q (P — целое, Q — натуральное). Для класса реализовать методы реализующие операции с дробями:

- сложение;
- вычитание;
- умножение;
- деление;
- сокращение дроби;
- возведение дроби в степень n (n — натуральное);
- функции, реализующие операции отношения (равно, не равно, больше или равно, - меньше или равно, больше, меньше).

В программе продемонстрировать использование объектов класса «Обыкновенная дробь»

В программе продемонстрировать использование объектов класса «Квадратная матрица».

Контрольные вопросы

1. Дайте определения понятиям *класс* и *объект*.
2. Что такое *конструктор* и *деструктор* класса? Какую роль они играют в жизненном цикле объекта?
3. Каким образом в языке Си++ реализуется разделение компонент класса на интерфейс и внутреннюю реализацию?
4. Чем отличаются статические компоненты классов от прочих компонентов? Как определить статический компонент класса?
5. Какого назначения указателя *this*? Приведите примеры его использования.
6. Перечислите, какие компоненты могут быть включены в класс языка С#. В чем особенности использования каждого из них?
7. Как определяются свойства языка С#? Что такое *аксессоры* доступа? Приведите примеры определения свойств в классе.

4) КЕЙС-ЗАДАНИЕ "ПЕРЕГРУЗКА СТАНДАРТНЫХ ОПЕРАЦИЙ ЯЗЫКА ПРОГРАММИРОВАНИЯ ДЛЯ ПОЛЬЗОВАТЕЛЬСКИХ ОБЪЕКТОВ"

Цель кейс-задания: Получить навык модификации программного интерфейса объекта путем перегрузки стандартных операций для манипуляции с объектами пользовательских типов.

Типовой вариант кейс-задания:

1. Разработать класс «Римское число» для работы с числами, записанными в римской системе счисления. Реализовать возможность ввода и вывода чисел в римской системе счисления, а также методы, реализующие операции:

- перевод натурального числа из десятичной системы счисления в римскую;
- перевод числа из римской системы счисления в десятичную;
- сложение;
- вычитание;
- умножение;
- целочисленное деление и нахождение остатка от деления;
- операции отношения (описать в виде логических функций)
- проверка правильности записи римского числа.

Операции сложения, вычитания, деления и умножения римских чисел реализовать как перегрузки соответствующих операций.

В программе продемонстрировать использование объектов класса «Римское число».

2. Разработать класс «Стек», реализующий функционирование стека LIFO. В классе реализовать возможность добавление и удаление элемента, проверки, не пуст и не переполнен ли стек, получения количества элементов в стеке. Функции добавления и удаления информации в/из стека реализовать как перегрузку операций сложения и вычитания.

С помощью класса «Стек» проверить на сбалансированность по круглым и угловым скобкам текст в файле.

3. Определить класс «Множество», реализующий функционирование множества символов русского языка. В классе реализовать возможности: добавление и удаления символа в/из множества, сравнения с другим множеством, объединение множеств, пересечения множеств, вычитания множеств.

Функции объединения и вычитания множеств реализовать как перегрузки операций сложения и вычитания.

С использованием класса «Множество» найти в файле пару предложений, которые содержат один и тот же набор символов, а также пару предложений, в которых не совпадают ни один символ.

4. Определить класс «Квадратная матрица». В классе реализовать методы, позволяющие: вычислять определитель матрицы, транспонировать матрицу, получать обратную матрицу, проверять, не является ли матрица вырожденной, перемножать на другую матрицу.

Контрольные вопросы

1. Что такое интерфейс класса? Каким требованиям он должен удовлетворять?
2. Каковы достоинства перегрузки операций для объектов класса?
3. Как перегрузить операцию языка C++ внешней по отношению к классу функцией?
4. Почему перегрузка операций в языке C# осуществляется статической функцией?
5. Какие ограничения на перегрузки операций в используемом вами языке программирования вам известны? С чем они связаны?

5) КЕЙС-ЗАДАНИЕ "ПРОГРАММИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ ОБЪЕКТНО-ОРИЕНТИРОВАННЫХ ИЕРАРХИЙ, ПОСТРОЕННЫХ В СООТВЕТСТВИИ С ПРИНЦИПОМ НАСЛЕДОВАНИЯ"

Цель кейс-задание: Получить навык создания классов с использованием принципа наследования и реализации полиморфного поведения классов.

Типовой вариант кейс-задание:

1. Определить объект файл с поддержкой операций создания, копирования, перемещения, удаления файла, наполнения содержимым. Унаследовать от него класс «Зашифрованный файл» с поддержкой функций шифрации – дешифрации содержимого. В программе продемонстрировать функциональность разработанных классов.

2. Определить класс «Матрица» с возможностью динамического выделения и освобождения памяти, наполнения матрицы, сохранения и чтения из файла. Унаследовать класс «Треугольная матрица» с приведением матрицы к треугольному виду (если возможно) автоматически при каждом изменении значения элемента матрицы. содержимого. В программе продемонстрировать функциональность разработанных классов.

3. Определить класс «Шахматная фигура» и унаследовать от него классы, соответствующие шахматным фигурам «Ферзь», «Пешка», «Конь». Задать позицию и определить список фигур, которые может убить некоторая выбранная. содержимого.

4. Определить абстрактный класс Currency для работы с денежными суммами. Определить в нем методы перевода в рубли и вывода на экран. На его основе реализовать классы Dollar, Euro и Pound (фунт стерлингов) с возможностью пересчета в центы и пенсы соответственно и указанием текущего курса.. Создать класс Purse (кошелек), содержащий массив объектов этих классов в динамической памяти. Предусмотреть возможность случайного наполнения кошелька купюрами различного типа и подсчета общей суммы кошелька при изменении курса валют. В программе продемонстрировать функциональность разработанных классов.

5. Определить абстрактный класс Function (функция) с виртуальными методами вычисления значения функции $y = f(x)$ в заданной точке x и вывода результата на экран, поиска минимума и максимума функции на заданном интервале.. Унаследовать от класса Function классы Hyperbola и Parabola, Exponenta, в которых уточняется функция $f(x)$ и задаются коэффициенты соответствующих функций. Продемонстрировать функциональные возможности этих классов (получением значений, вычислением максимума или минимума).

Контрольные вопросы.

1. В чем заключается механизм наследования? Какие преимущества он предоставляет в ООП?
2. Как реализуется механизм наследования в программах на объектно-ориентированном языке программирования?
3. От чего зависит доступность унаследованного члена класса в производном классе в языке C++?
4. Что такое виртуальная функция? Приведите свой пример эффективного полиморфного поведения функций.
5. Почему для классов с объявленными виртуальными методами в языке C++ необходимо объявлять виртуальным и деструктор?
6. Что такое абстрактный класс C++? Какие ограничения существуют при его использовании? Для чего подобные классы определяются в программе?
7. Чем реализация механизма в языке C# отличается от реализации C++?
8. Что такое интерфейс C#? Как он определяется и используется в программе?
9. Почему нельзя создавать экземпляры интерфейсов в программах на языке C#?
10. Как концепцию интерфейса можно реализовать в языке C++, в котором не определен подобный тип данных?

6) КЕЙС-ЗАДАЧА "ЗНАКОМСТВО СО СТАНДАРТНЫМИ БИБЛИОТЕКАМИ КЛАССОВ СОВРЕМЕННЫХ СИСТЕМ ПРОГРАММИРОВАНИЯ"

Цель кейс-задание: познакомиться с классами из стандартных библиотек современных систем программирования и приобрести навыки их практического использования

Типовой вариант кейс-задание:

1. Программная модель магазина. Модель должна генерировать поток покупателей в виде списка объектов класса Покупатель. Каждый покупатель делает выбор товаров, формируя корзину покупок. Касса реализуется как объект Очередь, в конец которой добавляются покупатели после выбора товара и с определенной задержкой (имитирующей расчет стоимости) удаляются покупатели с головы. Программ должна выводить: количество покупателей в магазине, длину очереди, общую сумму приобретенных товаров.

2. Дополнительное задание – реализовать несколько касс обслуживания покупателей.

3. Программная модель мастерской по ремонту бытовой техники. Модель должна генерировать поток заявок на ремонт техники различного типа. Все заявки поступают в общую очередь, после чего сортируются по мастерам – попадают в соответствующие им очереди. Некоторые заявки (помеченные как срочные), должны обрабатываться в первую очередь. Обработка заявки имитируется временной задержкой случайной длительности. Программа должна постоянно выводить: размер каждой очереди, количество заявок, обработанных каждым мастером.

4. Программная модель планировщика задач с вытесняющей многозадачностью. Модель должна генерировать последовательной потоков исполнения в виде списка объекта типа Поток. Каждый поток имеет собственный приоритет, идентификатор, время работы. Потоки распределяются по отдельным очередям исполнения в соответствии со значением приоритетов. На процессоре исполняется в течении кванта времени поток из головы очереди самых высокоприоритетных потоков. Если поток завершил работу (сумма выделенных ему квантов времени превысила его время работы), он удаляется из очереди, если нет – перемещается в хвост своей очереди. Программа должна постоянно выводить: общее количество потоков, длины очередей потоков с различными приоритетами, количество завершивших работу потоков. Дополнительное задание – реализовать обработку потоков несколькими процессорами (ядрами)

5. Реализовать русско-английский и англо-русский словарь. Программа должна позволять пополнять словарь новыми словами на русском или английском языке с указанием перевода на другой язык и разъяснением перевода. Содержимое словаря должно сохраняться в файле. По запросу пользователя на перевод слова программа должна приводить все возможные переводы данного слова и выводить пояснения к каждому переводу. (Использовать Dictionary (MultyDictionary) C# или map (multimap) STL C++)

6. Программная модель ремонтной мастерской. Необходимо осуществлять сборку компьютеров до полной комплектации (мат. плата, процессор, память, жесткий диск). На склад мастерской поступают компьютеры с вышедшими из строя компонентами. Необходимо перекомпоновывать состав компьютеров (извлекая рабочие детали из одного и заменяя неработающие в другой), чтобы они становились рабочими (имелись в наличии все компоненты в рабочем состоянии). Полностью собранные компьютеры удаляются со склада. Программа должна выводить статистику по количеству компьютеров на ремонте в мастерской, а также количество отремонтированных компьютеров.

Контрольные вопросы

1. В чем заключается принцип обобщенного программирования? Назовите его достоинства и недостатки.
2. Шаблоны функций языка C++: определение, назначение, примеры использования.
3. Как определить шаблон класса в языке C++? Приведите самостоятельный пример шаблона класса. Какие правила и ограничения на определения шаблонов классов в языке C++ вы знаете?
4. Какие виды компонентов включены в состав библиотеки STL C++? Объясните назначения каждого вида компонент.
5. Перечислите известные вам контейнеры STL. В чем заключаются их различия? Каковы критерии выбора вида контейнера при решении конкретной задачи?
6. Назначение обобщенных алгоритмов и чем они отличаются от методов контейнеров? Приведите самостоятельный пример использования обобщенного алгоритма STL.
7. Как реализуется принцип обобщенного программирования в языке C#? Приведите пример определения обобщения C#.
8. Что такое ограничение типа при определении обобщения. Как задаются эти ограничения. Какие ограничения вам известны?
9. Классифицируйте коллекции C#. Какую роль в работе коллекций C# играют интерфейсы?
10. Обоснуйте выбор коллекции или контейнера в решении своего варианта задания.

Критерии оценивания

- Научно-теоретический уровень выполнения кейс-задания и выступления.
- Полнота решения кейса.

- Степень творчества и самостоятельности в подходе к анализу кейса и его решению. Доказательность и убедительность.
- Форма изложения материала (свободная; своими словами; грамотность устной или письменной речи) и качество презентации.
- Культура речи, жестов, мимики при устной презентации.
- Полнота и всесторонность выводов.
- Наличие собственных взглядов на проблему.

Шкала оценивания

Баллы для учета в рейтинге (оценка)	Степень удовлетворения критериям
86-100 баллов «отлично»	Кейс–задание выполнено полностью, в рамках регламента, установленного на публичную презентацию, студент(ы) приводит (подготовили) полную четкую аргументацию выбранного решения на основе качественно сделанного анализа. Демонстрируются хорошие теоретические знания, имеется собственная обоснованная точка зрения на проблему(ы) и причины ее (их) возникновения. В случае ряда выявленных проблем четко определяет их иерархию. При устной презентации уверенно и быстро отвечает на заданные вопросы, выступление сопровождается приемами визуализации. В случае письменного отчета-презентации по выполнению кейс-задания сделан структурированный и детализированный анализ кейса, представлены возможные варианты решения (3-5), четко и аргументировано обоснован окончательный выбор одного из альтернативных решений.
71-85 баллов «хорошо»	Кейс–задание выполнено полностью, но в рамках установленного на выступление регламента, студент(ы) не приводит (не подготовили) полную четкую аргументацию выбранного решения. Имеет место излишнее теоретизирование, или наоборот, теоретическое обоснование ограничено, имеется собственная точка зрения на проблемы, но не все причины ее возникновения установлены. При устной презентации на дополнительные вопросы выступающий отвечает с некоторым затруднением, подготовленная устная презентация выполненного кейс-задания не очень структурирована. При письменном отчете-презентации по выполнению кейс-задания сделан не полный анализ кейса, без учета ряда фактов, выявлены не все возможные проблемы, для решения могла быть выбрана второстепенная, а не главная проблема, количество представленных возможных вариантов решения – 2-3, затруднена четкая аргументация окончательного выбора одного из альтернативных решений.
56-70 баллов «удовлетворительно»	Кейс–задание выполнено более чем на 2/3, но в рамках установленного на выступление регламента, студент(ы) расплывчато раскрывает решение, не может четко аргументировать сделанный выбор, показывает явный недостаток теоретических знаний. Выводы слабые, свидетельствуют о недостаточном анализе фактов, в основе решения может иметь место интерпретация фактов или предположения, Собственная точка зрения на причины возникновения проблемы не обоснована или отсутствует. При устной презентации на вопросы отвечает с трудом или не отвечает совсем. Подготовленная презентация выполненного кейс-задания не структурирована. В случае письменной презентации по выполнению кейс-задания не сделан детальный анализ кейса, далеко не все факты учтены, для решения выбрана второстепенная, а не главная проблема, количество представленных возможных вариантов решения – 1-2, отсутствует четкая аргументация окончательного выбора решения.
менее 56 баллов «неудовлетворительно»	Кейс-задание не выполнено, или выполнено менее чем на треть. Отсутствует детализация при анализ кейса, изложение устное или письменное не структурировано. Если решение и обозначено в выступлении или отчете-презентации, то оно не является решением проблемы, которая заложена в кейсе..

Комплект заданий для самостоятельной работы обучающихся

1. В компоненте MaskEdit реализовать шаблон ввода вещественных чисел с не более чем тремя десятичными знаками.
2. Текст, введенный в компонент Edit, динамически (в процессе набора) отобразить в компоненте ListBox.
3. Текст, введенный в компонент Memo, при нажатии на Button отобразить в компоненте Edit.
4. Текст, введенный в компонент Edit, динамически (в процессе набора) отобразить в компоненте Label.
5. Текст, введенный в компонент Edit вывести в окне сообщений.
6. С помощью компонента MainMenu создать строчное меню с двумя опциями подменю: О программе (при выборе должно появиться окно с информацией о разработчике) и Выход (завершает работу приложения).
7. Написать программу обработчик, ограничивающую ввод символов в компонент Edit цифрами.
8. Добавить в проект форму Form2 и при нажатии на кнопку Button отобразить ее.
9. Написать программу обработчик, не допускающую ввод цифр в компонент Edit.
10. Создать приложение, определяющее выбранную кнопку в компоненте RadioGroup и реагирующее соответственным образом (пусть для определенности выводится сообщение, что выбрана та или иная кнопка).
11. Создать динамический запрос в БД средствами Delphi
12. Осуществить сортировку данных в таблицах БД.
13. Текст, введенный в компонент Memo, динамически (в процессе набора) отобразить в компоненте Edit.
14. Для компонента Button написать обработчик события OnMouseMove, с выводом сообщения «Курсор находится над объектом»
15. Текст, введенный в компонент Edit, динамически (в процессе набора) отобразить в компоненте Memo.
16. Для компонента Memo написать обработчик события OnMouseMove, с выводом сообщения «Курсор находится над объектом»

Критерии оценивания

- правильность ответа по содержанию задания (учитывается количество и характер ошибок при ответе);
- полнота и глубина ответа (учитывается количество усвоенных фактов, понятий и т.п.);
- сознательность ответа (учитывается понимание излагаемого материала);
- логика изложения материала (учитывается умение строить целостный, последовательный рассказ, грамотно пользоваться специальной терминологией);
- использование дополнительного материала;
- рациональность использования времени, отведенного на задание (не одобряется затянутость выполнения задания, устного ответа во времени, с учетом индивидуальных особенностей обучающихся).

Шкала оценивания

Баллы для учета в рейтинге (оценка)	Степень удовлетворения критериям
86-100 баллов «отлично»	Обучающийся полно и аргументировано отвечает по содержанию вопроса (задания); обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только по учебнику, но и самостоятельно составленные; излагает материал последовательно и правильно.
71-85 баллов «хорошо»	Обучающийся достаточно полно и аргументировано отвечает по содержанию вопроса (задания); обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только по учебнику, но и самостоятельно составленные; излагает материал последовательно. Допускает 1-2 ошибки, исправленные с помощью наводящих вопросов.
56-70 баллов «удовлетворительно»	Обучающийся обнаруживает знание и понимание основных положений данного задания, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки.

менее 56 баллов «неудовлетворительно»	Обучающийся обнаруживает незнание ответа на соответствующее задание (вопрос), допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал. Отмечаются такие недостатки в подготовке обучающегося, которые являются серьезным препятствием к успешному овладению последующим материалом.
--	--

**Комплект тестовых заданий
Вариант 1.**

1. В основе объектно-ориентированного подхода лежит понятие...
 - a) иерархии классов
 - b) формализации свойств
 - c) рекурсии
 - d) декомпозиции объектов
2. Методика анализа, проектирования и написания приложений с помощью классов, каждый из которых является целостным фрагментом кода и обладает свойствами и методами, называется _____ программированием.
 - a) структурным
 - b) объектно-ориентированным
 - c) формальным
 - d) модульным
3. Понятие «иерархия классов» присуще методологии _____ программирования
 - a) структурного
 - b) объектно-ориентированного
 - c) динамического
 - d) эвристического
4. Одной из важных функций, реализуемых системой программирования, является...
 - a) автоматическое тестирование программного продукта на всех вариантах входных данных
 - b) автоматическая сборка разработанных модулей в единый проект
 - c) анализ существующих программных продуктов по соответствующей тематике
 - d) автоматизация выбора языка программирования, исходя из постановки задачи
5. Основой метода структурного программирования являются...
 - a) принцип модульности разработки сложных программ
 - б) использование композиции трех базовых элементов - линейной, ветвления и циклической структур
 - в) использование композиции двух базовых элементов - ветвления и циклической структур
 - д) использование большого количества подпрограмм
6. Правила композиции, используемые при структурном подходе к составлению алгоритмов:
 - a) альтернативный выбор
 - b) цикл
 - c) подпрограмма
 - d) метки
7. Значение переменной d после выполнения фрагмента алгоритма (операции mod (x,y)- получение остатка целочисленного деления x на y, div (x,y)-целочисленное деление x на y)
k = 30
выбор

при	div (k, 12)= 4:	d:=k;
при	div (k, 12) <5:	d:=2;
при	mod (k, 12) >9:	d:=3;
иначе:		d:=1;

Все
равно:
 - a) 2
 - b) 50
 - c) 1
 - d) 3
8. В отношении модель – объект НЕ НАХОДЯТСЯ понятия:
 - a) карта – местность
 - b) макет – дом
 - c) аннотация – реферат
 - d) глобус – Земля

9. Основной целью структурного программирования является:
- решение задач, для которых нет явного алгоритма
 - исключение использования подпрограмм
 - организация программного обеспечения с максимальной связностью модулей
10. В чём состоит отличие процедуры от функции?
- Функция возвращает результат работы в виде значения, которое может использоваться в качестве операнда в любом выражении.
 - Процедура возвращает результат работы в виде значения, которое может использоваться в качестве операнда в любом выражении.
 - Отличий нет.
11. Назначение свойства AutoSize?
- Размер объекта определяется автоматически его надписью или изображением на его поверхности (True).
 - Перенос слов при переполнении длины на другую строку.
 - Выравнивание объекта относительно контейнера, в котором находится объект.
12. Каково назначение свойства Align?
- Высота объекта в пикселях.
 - Ширина объекта в пикселях.
 - +Выравнивание объекта относительно контейнера, в котором находится объект.
13. Каково назначение свойства Name?
- Для изменения содержания объекта.
 - Для вывода надписи на поверхности объекта.
 - Для ссылки на объект в программе.
14. Каково назначение свойства Caption?
- Содержит текст, который будет отображаться вместе с объектом.
 - Определяет, является ли объект видимым или невидимым.
 - Для ссылки на объект в программе.
15. Каково назначение свойства Transparent?
- Задаёт прозрачный объект (True).
 - Устанавливает перенос слов при переполнении длины на другую строку.
 - Определяет, является ли объект видимым или невидимым.
16. Каково назначение свойства Visible?
- При значении свойства True размер объекта определяется его надписью или изображением на его поверхности.
 - Определяет, является ли объект видимым.
 - Для вывода надписи на поверхности объекта.
17. Каково назначение объекта ComboBox?
- Представляет собой прямоугольную область, заполненную строками списка для выбора пользователем нужных строк.
 - Позволяет выбирать одну строку из списка или непосредственно вводить ее в поле ввода.
 - Позволяет работать с текстовой информацией, представленной в виде двумерной таблицы.
18. Каково назначение объекта Button?
- Кнопка быстрого доступа: не может получать фокус, фиксируется положение нажатия, и не может открывать модальные окна.
 - Это кнопка, которая содержит надпись и рисунок на кнопке.
 - Это стандартная кнопка, которая содержит надпись.
19. Каково назначение объекта CheckBox?
- Используется для задания логических значений типов: «Да»(флажок установлен), «Нет»(флажок сброшен), «Не знаю»(флажок запрещен).
 - Графически выделять группу логически связанных объектов и объединять в одну группу зависимые переключатели.
 - Используется для вывода большого объема текста при ограниченном поле вывода.
20. Каково назначение объекта RadioButton?
- Зависимый переключатель. Объединяется с другими переключателями в группу. Включение одного из них автоматически выключает остальные в группе.
 - Используется для задания логических значений типов: «Да»(флажок установлен), «Нет»(флажок сброшен), «Не знаю»(флажок запрещен).
 - Кнопка, которая кроме надписи содержит рисунок.
21. Объединение данных и методов их обработки; разделение доступа к данным между классами называется ... (Инкапсуляция)
22. Классы объектов могут наследовать свойства друг друга согласно принципу (наследование)
23. Методы объектов могут переопределяться по принципу ... (полиморфизм)
24. Файл проекта имеет расширение ... (.dpr)

25. Файл формы имеет расширение ... (.dfm)
26. Готовые элементы интерфейса для программы (кнопки, переключатели, поля ввода, диалоги и т.п.) размещаются на ... (палитра компонентов).
27. Для управления свойствами свойствами (вкладка Properties) и событиями (Events) объекта, выбранного в окне формы существует окно ... (Инспектора объектов)
28. Текстовая метка используется для создания ... (надписей) на форме
29. Текстовое поле редактирования используется для (ввода) строки данных.
30. Для включения какой-либо настройки в программе используется ... (переключатель CheckBox)

Вариант 2.

1. В системе программирования отладчик используется для...
 - a) поиска и установления ошибок
 - b) создания библиотек подпрограмм
 - c) редактирование текстов программ
 - d) перевода программы в машинные коды
2. Модульная структура программы отражает одну из особенностей _____ программирования
 - a) эвристического
 - b) динамического
 - c) логического
 - d) структурного
3. Понятием «переменная» в традиционных языках программирования называется ...
 Именованная область памяти, в которой хранится некоторое значение
 Любое законченное минимальное смысловое выражение на языке программирования;
 Служебное слово на языке программирования
 Описание действий, которые должна выполнить программа.
4. Можно ли изменить значение параметра-константы в подпрограмме?
 - a) Нет.
 - b) Да.
5. Отличие переменной от константы
 - a) Отличий нет.
 - b) Переменная не изменяется при выполнении программы
 - c) Константу нельзя изменить при выполнении программы.
6. Должна ли переменная быть объявлена до её использования?
 - a) Да.
 - b) Не всегда.
 - c) Нет.
7. Могут ли быть в программе одноимённые глобальные и локальные переменные?
 - a) Нет.
 - b) Да.
8. Допускается ли массив записей?
 - a) Да.
 - b) -Нет.
9. Какие типы данных хранятся в переменной типа Variant?
 - a) Данные любого типа.
 - b) Данные любого типа, кроме записей множеств, статических массивов, файлов, классов, ссылок на классы, указателей и Int64.
 - c) Только перечисляемые, интервальные и структурированные типы.
10. Тип результата деления:
 - a) Integer.
 - b) Variant.
 - c) Extended
11. Чему равно для чётного числа N выражение $N \bmod 2$?
 - a) 0.
 - b) -1.
12. Какие существуют режимы вычисления логического выражения?
 - a) Последовательный и выборочный.
 - b) Полный и выборочный.
 - c) Сокращённый и полный (выражение вычисляется до конца).
13. Имеют ли логические операции больший приоритет, чем операции сравнения?
 - a) Да.
 - b) Нет.
14. В какой последовательности выполняются операции при равенстве приоритета?
 - a) In, And, Not, +.

- b) Not, And, +, In.
 - c) And, Not, In.
15. У какой операции выше приоритет – сравнения или логической?
- a) У логической операции.
 - b) У операции сравнения.
16. В чём состоит отличие процедуры от функции?
- a) Функция возвращает результат работы в виде значения, которое может использоваться в качестве операнда в любом выражении.
 - b) Процедура возвращает результат работы в виде значения, которое может использоваться в качестве операнда в любом выражении.
 - c) Отличий нет.
17. В чём состоит отличие параметра-переменной от выходного параметра?
- a) Не требует исходного значения от аргумента; при этом память, занимаемая аргументом, не очищается.
 - b) Требуется исходное значение от аргумента; при этом память, занимаемая аргументом, очищается. Подпрограмма формирует значение этого аргумента.
 - c) +Не требует исходного значения от аргумента; при этом память, занимаемая аргументом, очищается. Подпрограмма формирует значение этого аргумента.
18. Каково отличие параметров-переменных от параметров-значений?
- a) Параметры-переменные не могут быть изменены процедурой(функцией), а параметры-значения могут изменить своё значение в подпрограмме.
 - b) +Параметры-значения не могут быть изменены процедурой(функцией), а параметры-переменные могут изменить своё значение в подпрограмме.
 - c) Отличий нет.
19. Допускается ли вложенность составных операторов?
- a) Да.
 - b) Нет.
 - c) Не всегда.
20. Разрешается ли передавать управление извне вовнутрь цикла или другой подпрограммы?
- a) Нет.
 - b) Да.
21. Почему рекомендуется минимизировать число операторов GoTo в программе?
- a) Для увеличения скорости работы программы.
 - b) Для увеличения надёжности программы.
 - c) Для повышения наглядности исходного текста программы.
22. Можно ли досрочно выйти из любого цикла?
- a) Да, командой Exit.
 - b) Да, командой Break.
 - c) Нет.
23. Можно ли выполнить цикл For с шагом 2?
- a) Нет.
 - b) Да.
24. Может ли счётчик счётного цикла изменяться на число, отличное от единицы?
- a) Нет.
 - b) Да.
25. Для чего используется репозиторий (repository)?
- a) Репозиторий предназначен для изменения свойств объектов.
 - b) Репозиторий служит для хранения типовых объектов, называемых шаблонами (заготовками), используемых при разработке других приложений.
 - c) Репозиторий служит для настройки состава панелей компонентов, просмотра компонентов и пакетных файлов, в которых они хранятся.
26. Каково назначение свойства Name?
- a) Для изменения содержания объекта.
 - a) Для вывода надписи на поверхности объекта.
 - b) Для ссылки на объект в программе.
27. Каково назначение свойства Caption?
- a) Содержит текст, который будет отображаться вместе с объектом.
 - b) Определяет, является ли объект видимым или невидимым.
 - c) Для ссылки на объект в программе.
30. Каково назначение встроенных констант?
- a) Повышение наглядности исходного текста программы для пользователя.
 - b) Увеличение скорости работы программы.
 - c) Повышение надёжности программы.

Вариант 3.

1. В состав средств программирования на языке высокого уровня обязательно входит
 - a) Жесткий диск
 - b) +Транслятор
 - c) Табличный редактор
 - d) Инструкция программиста
2. Процесс написания программы никогда НЕ включает...
 - a) Изменение физических параметров компьютера
 - b) Процесс отладки
 - c) Редактирование текста программы
 - d) Запись операторов в соответствующей языку программирования форме
3. Система программирования предоставляет программисту возможность :
 - a) Анализа существующих программных продуктов по соответствующей тематике;
 - b) Планирование рабочего дня;
 - c) Выбора зыка программирования;
 - d) Автоматической сборки разработанных модулей в единый проект
 - e) Автоматического построения математической модели, исходя их постановки задачи
4. Языком логического программирования является
 - a) ЛИСП
 - b) Пролог
 - c) АДА
 - d) Бейсик
5. В результате работы алгоритма
 - a) $Y: =X+5$
 - b) $Y: =X$
 - c) $Y: =X+Y$

Вывод Y

Переменная Y приняла значение 14. Укажите число, которое являлось значением переменной X до начала работы алгоритма.

- a) 5
 - b) 2
 - c) 7
 - d) 10
6. Каково назначение свойства AutoSize?
 - a) Размер объекта определяется автоматически его надписью или изображением на его поверхности (True).
 - b) Перенос слов при переполнении длины на другую строку.
 - c) Выравнивание объекта относительно контейнера, в котором находится объект.
7. Каково назначение свойства Align?
 - a) Высота объекта в пикселях.
 - b) Ширина объекта в пикселях.
 - c) +Выравнивание объекта относительно контейнера, в котором находится объект.
8. Каково назначение свойства Enabled?
 - a) +Разрешает либо блокирует работу с объектом.
 - b) Задаёт прозрачный объект (True).
 - c) Определяет, является ли объект видимым.
9. Каково назначение свойства Text?
 - a) Устанавливает перенос слов при переполнении длины на другую строку.
 - b) +Определяет текущее значение объекта.
 - c) Для вывода надписи на поверхности объекта.
10. Каково назначение события OnClick?
 - a) Двойной щелчок левой кнопкой мыши по объекту.
 - b) Нажатие любой кнопки мыши.
 - c) +Щелчок левой кнопкой мыши по объекту.
11. Каково назначение события OnChange?
 - a) +Событие происходит при изменении состояния объекта.
 - b) Получение фокуса элементом управления.
 - c) Событие происходит при изменении размеров объекта.
12. Каково назначение метода Refresh?
 - a) +Формирование и визуализация формы или объекта.
 - b) Изменении размеров объекта.
 - c) Нажатие любой кнопки мыши.

13. Каково назначение объекта Мемо?
 - a) Предназначен для ввода и/или отображения одной текстовой строки.
 - b) +Используется для ввода и/или отображения многострочного текста.
 - c) Для задания текста, выводимого на поверхности объекта.
14. Каково назначение объекта ListBox?
 - a) +Представляет собой прямоугольную область, заполненную строками списка для выбора пользователем нужных строк.
 - b) Позволяет работать с текстовой информацией, представленной в виде двумерной таблицы.
 - c) Позволяет выбирать одну строку из списка или непосредственно вводить ее в поле ввода.
15. Каково назначение объекта ComboBox?
 - a) Представляет собой прямоугольную область, заполненную строками списка для выбора пользователем нужных строк.
 - b) +Позволяет выбирать одну строку из списка или непосредственно вводить ее в поле ввода.
 - c) Позволяет работать с текстовой информацией, представленной в виде двумерной таблицы.
16. Каково назначение объекта Button?
 - a) Кнопка быстрого доступа: не может получать фокус, фиксируется положение нажатия, и не может открывать модальные окна.
 - b) Это кнопка, которая содержит надпись и рисунок на кнопке.
 - c) +Это стандартная кнопка, которая содержит надпись.
17. Каково назначение объекта CheckBox?
 - a) +Используется для задания логических значений типов: «Да»(флажок установлен), «Нет» (флажок сброшен), «Не знаю» (флажок запрещен).
 - b) Графически выделять группу логически связанных объектов и объединять в одну группу зависимые переключатели.
 - c) Используется для вывода большого объема текста при ограниченном поле вывода.
18. Каково назначение объекта RadioButton?
 - a) Зависимый переключатель.Объединяется с другими переключателями в группу.Включение одного из них автоматически выключает остальные в группе.
 - b) Используется для задания логических значений типов: «Да»(флажок установлен), «Нет» (флажок сброшен), «Не знаю» (флажок запрещен).
 - c) Кнопка,которая кроме надписи содержит рисунок.
19. Объединение данных и методов их обработки; разделение доступа к данным между классами называется ... (Инкапсуляция).
20. StrToInt (Строка):integer; - преобразование строки в ... (целое) число
- 21.StrToFloat (Строка):Extended; - преобразование строки в ... (вещественное) число
22. Классы объектов могут наследовать свойства друг друга согласно принципу(наследование)
23. Методы объектов могут переопределяться по принципу ... (полиморфизм)
24. Файл проекта имеет расширение ...(.dpr)
25. Файл формы имеет расширение ... (.dfm)
26. Готовые элементы интерфейса для программы (кнопки, переключатели, поля ввода, диалоги и т.п.) размещаются на ... (палитра компонентов).
27. Для управления свойствами свойствами (вкладка Properties) и событиями (Events) объекта, выбранного в окне формы существует окно ... (Инспектора объектов)
28. Текстовая метка используется для создания ... (надписей) на форме
29. Текстовое поле редактирования используется для (ввода) строки данных.
30. Для включения какой-либо настройки в программе используется ... (переключатель CheckBox)

Критерии оценивания

- отношение правильно выполненных заданий к общему их количеству.

Шкала оценивания

Баллы для учета в рейтинге (оценка)	Степень удовлетворения критериям
86-100 баллов «отлично»	Выполнено от 9 до 10 заданий
71-85 баллов «хорошо»	Выполнено от 7 до 8 заданий
56-70 баллов «удовлетворительно»	Выполнено от 5 до 6 заданий
0-55 баллов «неудовлетворительно»	Выполнено менее 5 заданий