

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«ТЮМЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Тобольский педагогический институт им. Д.И.Менделеева (филиал)
Тюменского государственного университета

УТВЕРЖДАЮ

Директор

Шилов С.П.



ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
ПО ДИСЦИПЛИНЕ

ОСНОВЫ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ

44.03.05 Педагогическое образование (с двумя профилями подготовки):
Профили математика; информатика
Форма обучения очная

1. Паспорт оценочных материалов по дисциплине

№ п/п	Темы дисциплины (модуля)/ разделы в ходе текущего контроля, вид промежуточной аттестации (зачет, экзамен, с указанием семестра)	Код и содержание контролируемой компетенции (или ее части)	Наименование оценочного средства (краткое описание с указанием количества вариантов, заданий и т.п.)
1	Введение в объектно-ориентированное программирование	ОК-3 ПК-4	Контрольные вопросы для обсуждения Задания лабораторных работ
2	Библиотеки объектов. Конструирование объектов		Контрольные вопросы для обсуждения Задания лабораторных работ
3	Визуальное программирование в среде программирования Lazarus.		Контрольные вопросы для обсуждения Задания лабораторных работ Контрольная работа
	Раздел 1-3		Вопросы для подготовки к экзамену (1-28).

2. Виды и характеристика оценочных средств

С целью текущего контроля знаний проводится проверка выполнения лабораторных заданий, выполнения практических заданий контрольных работ, опрос по вопросам для устного контроля знаний.

2.1. Контрольные вопросы

Контрольные вопросы используются для проведения анализа материала, самостоятельного углубления знаний, а также для самопроверки знаний студентов по отдельным вопросам и/или темам дисциплины. Критерии оценки ответа (табл.) доводятся до сведения обучающихся в начале занятий. Оценка объявляется в конце занятия.

Балл	Критерий оценивания
1-2	<ul style="list-style-type: none">- показывает знание основных понятий темы, грамотно пользуется терминологией;- проявляет умение анализировать и обобщать информацию, навыки связного описания явлений и процессов;- демонстрирует умение излагать учебный материал в определенной логической последовательности;- показывает умение иллюстрировать теоретические положения конкретными примерами;- демонстрирует сформированность и устойчивость знаний, умений и навыков;- могут быть допущены одна–две неточности при освещении второстепенных вопросов.
0	<ul style="list-style-type: none">- не раскрыто основное содержание учебного материала;- обнаружено незнание или непонимание большей или наиболее важной части учебного материала;- допущены ошибки в определении понятий, при использовании терминологии, в описании явлений и процессов, решении задач, которые не исправлены после нескольких наводящих вопросов;- не сформированы компетенции, отсутствуют соответствующие знания, умения и навыки.

2.2. Практические задания

Практические задания в рамках лабораторных занятий используются для оценки умений по темам дисциплины. Включают в себе выполнение практических расчетных заданий или выполнение проектов творческого характера с построением алгоритма решения и его реализации на языке программирования.

Балл	Критерий оценивания заданий
3	Свободно применяет полученные знания при выполнении практических заданий; Выполнил работу в полном объеме с соблюдением необходимой последовательности действий; В письменном отчете по работе правильно и аккуратно выполнены все записи; При ответах на контрольные вопросы правильно понимает их сущность, дает точное определение и истолкование основных понятий, использует специальную терминологию дисциплины, не затрудняется при ответах на видоизмененные вопросы, сопровождает ответ примерами.
1-2	Практическая работа выполнена не полностью, но объем выполненной части позволяет получить правильные результаты и выводы; В ходе выполнения работы студент продемонстрировал слабые практические навыки, были допущены ошибки; Студент умеет применять полученные знания при решении простых задач по готовому алгоритму; В письменном отчете по работе допущены ошибки; При ответах на контрольные вопросы правильно понимает их сущность, но в ответе имеются отдельные пробелы и при самостоятельном воспроизведении материала требует дополнительных и уточняющих вопросов преподавателя.
0	Практическая работа выполнена не полностью и объем выполненной работы не позволяет сделать правильных выводов, у студента имеются лишь отдельные представления об изученном материале, большая часть материала не усвоена; В письменном отчете по работе допущены грубые ошибки, либо он вообще отсутствует; На контрольные вопросы студент не может дать ответов, так как не овладел основными знаниями и умениями в соответствии с требованиями программы.

2.3. Контрольная работа

Контрольные работы проводятся по темам «Линейные программы. Операторы ветвления. Циклы», «Строковый тип данных», «Одномерные и двумерные массивы». Отчет о выполнении заданий оценивается по 5-ти балльной системе. Критерии оценки ответа (табл.) доводятся до сведения обучающихся в начале занятий.

Балл	Критерий оценивания
"отлично"	Выполнил работу самостоятельно и без ошибок; допустил не более одного недочета; демонстрирует понимание способов и видов учебной деятельности по созданию алгоритма и программы; владеет терминологией и может прокомментировать этапы своей деятельности и полученный результат; может предложить другой способ деятельности или алгоритм выполнения задания.
"хорошо"	Выполнил работу самостоятельно и без ошибок; допустил не более двух (для простых задач) и трех (для сложных задач) недочетов; демонстрирует понимание способов и видов учебной деятельности по созданию алгоритма и программы; может прокомментировать этапы своей деятельности и полученный результат (например, дает комментарии о выполненных действиях при форматировании алгоритма ли листинга программы; затрудняется предложить другой способ деятельности или алгоритм выполнения задания.

"удовлетворительно"	Если студент правильно выполнил более 50% всех заданий и при этом: демонстрирует общее понимание способов и видов учебной деятельности по созданию алгоритма и программы; может прокомментировать некоторые этапы своей деятельности и полученный результат. Или при условии выполнения всей работы студент допустил: для простых задач – одну грубую ошибку или более четырех недочетов; для сложных задач – две грубые ошибки или более восьми недочетов. Сложным считается задание, которое естественным образом разбивается на несколько частей при его выполнении.
"неудовлетворительно"	Допустил число ошибок и недочетов, превышающее норму, при которой может быть выставлена оценка «удовлетворительно»; правильно выполнил не более 10% всех заданий. Или не приступил к выполнению работы.

2.4. Экзамен в форме собеседования по вопросам

Критерии выставления оценки

Оценка «отлично»:

- полно раскрыл содержание материала в объеме, предусмотренном программой;
- изложил материал грамотным языком в определенной логической последовательности, точно используя специализированную терминологию и символику;
- показал умение иллюстрировать теоретические положения конкретными примерами, применять их в новой ситуации при выполнении практического задания;
- продемонстрировал усвоение ранее изученных сопутствующих вопросов, сформированность и устойчивость используемых при ответе умений и навыков;
- отвечал самостоятельно без наводящих вопросов.

Оценка «хорошо»:

- в изложении допущены небольшие пробелы, не исказившие логического и информационного содержания ответа;
- нет определенной логической последовательности, неточно используется специализированная терминология и символика;
- допущены один-два недочета при освещении основного содержания ответа, исправленные по замечанию преподавателя;
- допущены ошибка или более двух недочетов при освещении второстепенных вопросов или в выкладках, легко исправленные по замечанию или вопросу преподавателя.

Оценка «удовлетворительно»:

- неполно или непоследовательно раскрыто содержание материала, но показано общее понимание вопроса, имелись затруднения или допущены ошибки в определении понятий, использовании терминологии, блок-схем и листингах, исправленные после нескольких наводящих вопросов преподавателя;
- студент не справился с применением теории в новой ситуации при выполнении практического задания, но выполнил задания обязательного уровня сложности по данной теме,
- при знании теоретического материала выявлена недостаточная сформированность основных умений и навыков.

Оценка «неудовлетворительно»:

- не раскрыто основное содержание учебного материала;
- обнаружено незнание или непонимание студентом большей или наиболее важной части учебного материала,
- допущены ошибки в определении понятий, при использовании терминологии, в блок-схемах и листингах программ, которые не исправлены после нескольких наводящих вопросов преподавателя.

3. Оценочные средства

3.1. Контрольные вопросы

1. Что такое программа?
2. Что такое язык программирования?
3. Каковы основные свойства программ?
4. Какие есть графические операторы?
5. Какие есть операторы ввода- вывода?
6. Какие есть математические функции?
7. Как записываются логические условия?
8. Что такое структурированные алгоритмы?
9. Что такое спецификации программ?
10. Что такое постановка задачи?
11. Что такое способ решения?
12. Что такое метод решения?
13. Каков порядок решения новых задач?
14. Каковы основные типы ошибок в программах?
15. Что такое индуктивные рассуждения?
16. Что такое сложные алгоритмы и программы?
17. Что такое упорядоченная последовательность?
18. Что такое упорядочение методом «пузырька»?
19. Как доказывается правильность сложных программ?
20. Что такое разработка программ «сверху-вниз»?
21. Методы, свойства, события
22. Создание первого проекта, операторы ввода и вывода данных.
23. Обработка ошибок
24. Организация проекта в Lazarus.
25. Основные файлы проекта.
26. Структура проекта
27. Организация взаимодействия форм
28. Работа с компонентами
29. Кнопка (TButton)
30. Надпись (TLabel)
31. Поле ввода (TEdit). Обработка исключительных ситуаций .
32. Флажок (TCheckBox)
33. Переключатель (T RadioButton)
34. Примеры программ с переключателями и флажками в структурах принятия решений
35. Список (TListBox)
36. Поле со списком (TComboBox)
37. Объединение элементов (TGroupBox)
38. Полоса прокрутки (TScrollBar)
39. Список с флажками (TCheckListBox)
40. Создание главного меню (TMainMenu)

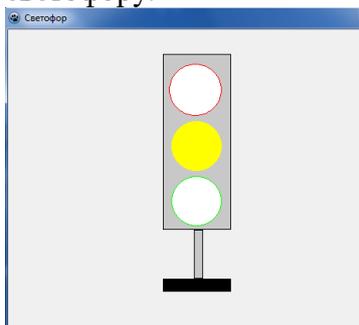
3.2. Практические задания

1. Загрузите программу. Проведите эксперименты:

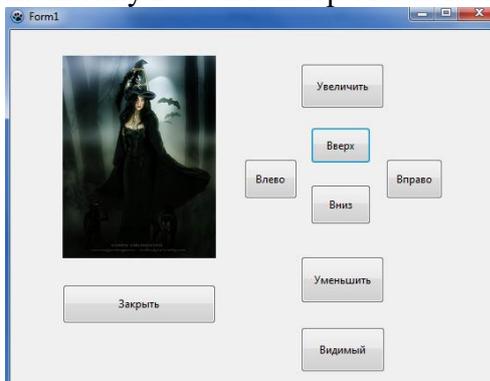
Действие	Результат
Щелкните в поле Caption паспорта формы Form1 и наберите- ФОРМА	

Измените свойство формы Width-200	
Измените свойство формы Color- clLime	
Измените свойство формы Height- 100	

2. Заданы два катета прямоугольного треугольника. Найти гипотенузу и углы треугольника.
3. Известна диагональ квадрата d . Вычислить площадь S и периметр P квадрата.
4. Треугольник задан величинами своих сторон – a , b , c . Найти углы треугольника – α , β , γ .
5. Тело имеет форму параллелепипеда с высотой h . Прямоугольник в основании имеет диагональ d . Известно, что диагонали основания пересекаются под углом α . Найти объем тела V и площадь поверхности S .
6. Задан первый член геометрической прогрессии и ее знаменатель. Вычислить сумму n членов геометрической прогрессии и значение n -го члена.
7. Тело падает с высоты h . Какова его скорость в момент соприкосновения с землей и когда это произойдет.
8. Создать программу, выполняющую следующие действия. После запуска программы по щелчку мышью на кнопке «Приветствие» появляется сообщение «Первые успехи!». Для выхода из программы необходимо щелкнуть мышью на кнопке «Выход».
9. Создать программу, которая будет генерировать случайные числа от нуля до десяти и сверять их с введенными с клавиатуры.
10. Используя генератор случайных чисел, напишите программу «Орел и решку», в которой при нажатии на кнопку в окне редактора Edit будет выдаваться «орел» или «решку».
11. Создать программу, в которой окружности поочередно будут менять цвета аналогично светофору.



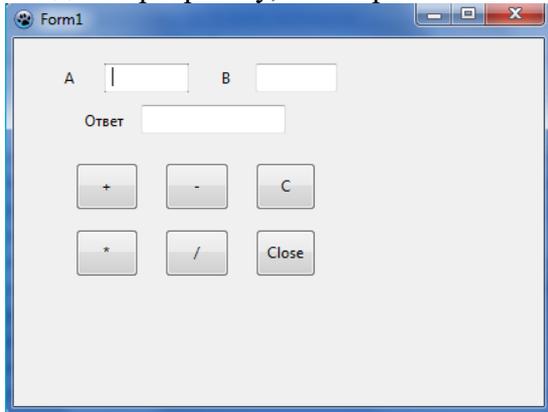
12. Создать программу, в которой окружности поочередно будут менять цвета аналогично светофору. Добавьте кнопку, по нажатию на которую можно будет переключать цвета в светофоре самостоятельно (для переключения используйте объект RadioGroup. По нажатию на кнопку Button таймер отключается и включается RadioGroup)



13. Создать программу, в которой по нажатию на кнопки картинка начнет перемещаться, исчезать, появляться и менять свои размеры. Сделайте так, чтобы при нажатии на кнопку «Видимый» менялся текст согласно наличию или отсутствию картинки на форме (используйте в написании кода свойство Caption). Добавьте кнопку «Движение по

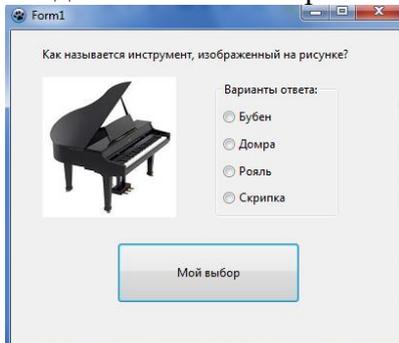
диагонали» и пропишите код для нее (используйте свойства перемещения по вертикали и по горизонтали вместе).

14. Создать программу, в которой по нажатию на кнопки будут производиться вычисления.



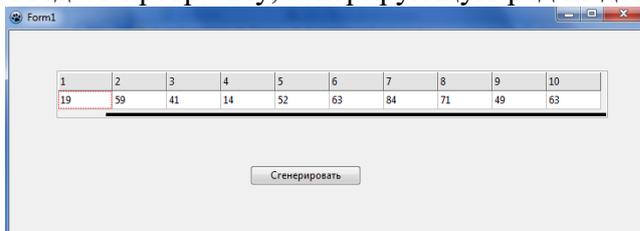
15. Для предыдущей задачи добавьте в калькулятор кнопки для вычисления квадратного корня, возведения в квадрат, нахождения Sin и Cos.

16. Создать тестовый вопрос с выбором одного варианта ответа.



17. Сделайте аналогичную программу, но с множественным выбором ответов (используйте объект CheckGroup).

18. Создать программу, генерирующую ряд из десяти чисел.

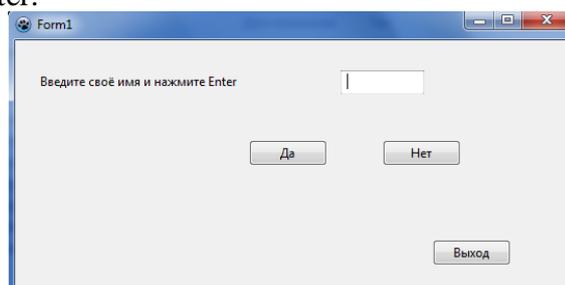


19. Напишите программу для создания двумерного массива.

20. Создать программу, в которой прыгает шарик, отталкиваясь от краев формы.

21. Напишите программу для описания броуновского движения (Объекты должны отталкиваться не только от краев формы, но и друг от друга).

22. Создать программу, в которой будет происходить диалог компьютера с пользователем по нажатию на кнопки и Enter.



3.3. Задания контрольной работы

1. Построить систему классов для описания плоских геометрических фигур: круга, квадрата, прямоугольника. Предусмотреть методы для создания объектов, перемещения на плоскости, изменения размеров и поворота на заданный угол.
2. Построить описание класса, содержащего информацию о почтовом адресе организации. Предусмотреть возможность раздельного изменения составных частей адреса, создания и уничтожения объектов этого класса.
3. Составить описание класса для представления комплексных чисел с возможностью задания вещественной и мнимой частей как числами типов `double`, так и целыми числами. Обеспечить выполнение операций сложения, вычитания и умножения комплексных чисел.
4. Составить описание класса для работы с цепными списками строк (строки произвольной длины) с операциями включения в список, удаления из списка элемента с заданным значением данного, удаления всего списка или конца списка, начиная с заданного элемента.
5. Составить описание класса для объектов-векторов, задаваемых координатами концов в трехмерном пространстве. Обеспечить операции сложения и вычитания векторов с получением нового вектора (суммы или разности), вычисления скалярного произведения двух векторов, длины вектора, косинуса угла между векторами.
6. Составить описание класса прямоугольников со сторонами, параллельными осям координат. Предусмотреть возможность перемещения прямоугольников на плоскости, изменения размеров, построения наименьшего прямоугольника, содержащего два заданных прямоугольника, и прямоугольника, являющегося общей частью (пересечением) двух прямоугольников.
7. Составить описание класса для определения одномерных массивов целых чисел (векторов). Предусмотреть возможность обращения к отдельному элементу массива с контролем выхода за пределы индексов, возможность задания произвольных границ индексов при создании объекта и выполнения операций поэлементного сложения и вычитания массивов с одинаковыми границами индексов, умножения и деления всех элементов массива на скаляр, печати (вывода на экран) элементов массива по индексам и всего массива.
8. Составить описание класса для определения одномерных массивов строк фиксированной длины. Предусмотреть возможность обращения к отдельным строкам массива по индексам, контроль выхода за пределы индексов, выполнения операций поэлементного сцепления двух массивов с образованием нового массива, слияния двух массивов с исключением повторяющихся элементов, печать (вывод на экран) элементов массива и всего массива.
9. Составить описание класса многочленов от одной переменной, задаваемых степенью многочлена и массивом коэффициентов. Предусмотреть методы для вычисления значения многочлена для заданного аргумента, операции сложения, вычитания и умножения многочленов с получением нового объекта-многочлена, печать (вывод на экран) описания многочлена.
10. Составить описание класса одномерных массивов строк, каждая строка которых задается длиной и указателем на выделенную для нее память. Предусмотреть возможность обращения к отдельным строкам массива по индексам, контроль выхода за пределы индексов, выполнения операций поэлементного сцепления двух массивов с образованием нового массива, слияния двух массивов с исключением повторяющихся элементов, печать (вывод на экран) элементов массива и всего массива.
11. Составить описание объектного типа `TMatr`, обеспечивающего размещение матрицы произвольного размера с возможностью изменения числа строк и столбцов, вывода на экран подматрицы любого размера и всей матрицы.

3.4. Вопросы к экзамену

1. Основные принципы объектно-ориентированного программирования.
2. Понятие объекта. Построение объекта.

3. Понятие класса. Организация и структура класса. Конструктор и деструктор.
4. Понятие метода. Динамические, виртуальные и абстрактные методы.
5. Среда программирования Lazarus. Элементы интегрированной среды разработчика: назначение, основные окна, выполнение основных действий.
6. Файловая система. Технология разработки программ. Этапы создания Lazarus-приложений. Структура программы в среде Lazarus.
7. Основные типы данных: встроенные и определяемые пользователем.
8. Динамические структуры данных.
9. Функции. Математические функции. Функции преобразования.
10. Подпрограммы в среде Lazarus: понятие, назначение, виды и структура.
11. Модуль. Структура модуля. Понятие программного кода. Стандартные модули в среде Lazarus.
12. Консольное приложение. Структура и инструкции.
13. Организация работы условных структур в среде Lazarus.
14. Обработка циклических операций в среде Lazarus.
15. Обработка исключительных ситуаций в среде Lazarus.
16. Свойства компонентов и типы свойств. Примеры программного изменения свойств в среде Lazarus.
17. События, методы обработки событий. Методы обработки событий программистом.
18. Форма. Основные характеристики компонента форма. Управление видимостью форм.
19. Понятие вторичной формы. Модальные и немодальные диалоговые окна.
20. Кнопки. Основные свойства, события и методы.
21. Флажки и переключатели. Основные свойства, события и методы.
22. Списки. Основные свойства, события и методы.
23. Использование таблиц в среде Lazarus, способы заполнения.
24. Компоненты среды Lazarus, реализующие диалоги.
25. Использование главного и контекстного меню проекта. Конструктор меню.
26. Технология Drag and Drop «тащи и бросай» и ее реализация средствами среды Lazarus.
27. Средства и компоненты среды Lazarus, предназначенные для работы с базами данных.
28. Работа с графическими элементами в среде Lazarus.

3.5. Балльно-рейтинговая аттестация

Экзамен может быть выставлен автоматически в рамках балльной системы, разработанной преподавателем и доведенной до сведения обучающихся на первом занятии

Распределение баллов по темам и видам работ

№	№ темы	Формы оцениваемой работы	Количество часов	Макс. количество баллов
1	Лабораторная работа 1-15	Отчет о выполнении заданий лабораторных работ. Ответы на контрольные вопросы	30	45
	Лекции 1-8	Конспект лекций. Ответы на контрольные вопросы	16	16
	Самостоятельная работа	Письменный отчет	69	34
	Контрольная работа	Отчет о выполнении заданий контрольной работы	2	5
		Итого		72

Промежуточная аттестация может быть выставлена с учетом совокупности баллов, полученных обучающимся в рамках текущего контроля, включающего выполнение и защиту лабораторных и контрольных работ.

№	Баллы	Оценки
1.	0-60	Неудовлетворительно
2.	61-75	Удовлетворительно
3.	76-90	Хорошо
4.	91-100	Отлично