

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт сервиса, туризма и дизайна (филиал) СКФУ в г. Пятигорске

УТВЕРЖДАЮ

Зав. кафедрой Систем управления и
информационных технологий
ИСТИД (филиал) СКФУ в г. Пятигорске

И.М. Першин
«__» _____ 202_ г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
для проведения текущей и промежуточной аттестации

По дисциплине	ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ
Направление подготовки	09.03.02 Информационные системы и технологии
Направленность (профиль)	Информационные системы и технологии
Квалификация выпускника	бакалавр
Форма обучения	очная
Год начала обучения	2020

Объем занятий: Итого	108,0 ч.	4 з.е.
В т.ч. аудиторных	27,0 ч.	
Из них:		
Лекций	13,5 ч.	
Лабораторных работ	13,5 ч.	
Практических занятий	___ ч.	
Самостоятельной работы	54,0 ч.	
Экзамен в 5 семестре	27,0 ч.	

Дата разработки:

Предисловие

1. Назначение: для проверки знаний, умений и навыков текущего контроля и промежуточной аттестации.
2. Фонд оценочных средств текущего контроля и промежуточной аттестации на основе рабочей программы дисциплины «Объектно-ориентированное программирование» в соответствии с образовательной программой по направлению подготовки 09.03.02 Информационные системы и технологии, утвержденной на заседании учебно-методического совета СКФУ протокол №___ от _____ г.
3. Разработчики: Н.И. Битюцкая, доцент кафедры СУиИТ.
4. ФОС рассмотрен и утвержден на заседании кафедры систем управления и информационных технологий Протокол №___ от _____ г.
5. ФОС согласован с выпускающей кафедрой систем управления и информационных технологий Протокол №___ от _____ г.

6. Проведена экспертиза ФОС. Члены экспертной группы, проводившие внутреннюю экспертизу:

Председатель _____ Першин И.М.
_____ Чернышев А.Б.
_____ Антонов В.Ф.

Экспертное заключение: данные оценочные средства соответствует требованиям федерального государственного образовательного стандарта высшего образования, рекомендуются для использования в учебном процессе.

«___» _____

_____ (подпись)

7. Срок действия ФОС - один год.

**Паспорт фонда оценочных средств
для проведения текущего контроля и промежуточной аттестации**

По дисциплине

**ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ
ПРОГРАММИРОВАНИЕ**

Направление подготовки

**09.03.02 Информационные системы и
технологии**

Направленность (профиль)

Информационные системы и технологии

Квалификация выпускника

бакалавр

Форма обучения

очная

Год начала обучения

2020

Код оцениваемой компетенции (или её части)	Модуль, раздел, тема (в соответствии с Программой)	Тип контроля	Вид контроля	Компонент фонда оценочных средств	Количество заданий для каждого уровня, шт.	
					Базовый	Повышенный
ПК-8, ПК-17	Темы 1 - 9	Текущий	Письменный	Комплект заданий для лабораторных работ	26	11
	Темы 1 - 9	Текущий	Устный	Вопросы для собеседования	38	12
	Темы 1 - 9	Промежуточный (экзамен)	Устный	Вопросы к экзамену	30	9
				Вопросы для проверки уровня знаний	23	6
				Вопросы для проверки умений и навыков	7	3

Составитель _____ Битюцкая Н.И.

(подпись)

« ____ » _____ 20 ____ г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт сервиса, туризма и дизайна (филиал) СКФУ в г. Пятигорске

УТВЕРЖДАЮ

Зав. кафедрой Систем управления и
информационных технологий
ИСТИД (филиал) СКФУ в г. Пятигорске
И.М. Першин
«__» _____ 202_ г.

Вопросы к экзамену

по дисциплине

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

Базовый уровень

Знать

1. Основные стили и парадигмы программирования. Парадигма ООП. Понятия объекта и класса.
2. Microsoft Visual Studio.NET: ее компоненты, версии, редакции и возможности.
3. Идентификаторы и литералы C#. Типы данных. Преобразование типов.
4. Арифметические и логические операторы C#.
5. Декларативные инструкции C#.
6. Инструкции выбора (if, switch).
7. Инструкции повторения (for, while, do-while, foreach).
8. Инструкции перехода (break, continue, goto, return).
9. Способы объявления и инициализации одномерных и двумерных массивов.
10. Свойства и методы класса Array.
11. Объявление класса. Модификаторы доступа.
12. Создание методов классов. Типы возвращаемых методом значений.
13. Конструкторы классов.
14. Задание свойств класса.
15. Назначение синтаксиса методов класса System.IO.File: OpenRead, OpenWrite, ReadAllText, ReadAllLines, WriteAllText, WriteAllLines, Copy, Move, Delete, Exists.
16. Методы класса String: Length, ToUpper, ToLower, Substring, IndexOf, Remove, Insert, Trim и др.
17. Основные кодировки символов и их особенности.
18. Создание экземпляра DateTime и DateTimeOffset.
19. Получение текущих даты и времени. Манипулирование датой и временем.
20. Форматирование даты и времени. Преобразование даты в строку.
21. .
22. Специальные форматные строки для даты и времени.

23. Преобразование строки в дату/время.

- Уметь 24. Настройка среды разработки Microsoft Visual Studio.NET.
Владеть 25. Создание, открытие и запуск программ в Microsoft Visual Studio.NET.
26. Создание исполняемых файлов.
27. Использование методики обработки строк для шифрования текстовых файлов.
28. Определение в программах различных типов ошибок.
29. Использование окна Watch для проверки значений переменных во время выполнения программы. Использование окна Command (Окно команд) для изменения значений переменных и исполнения команд в Visual Studio.
30. Исправление ошибок времени исполнения с помощью обработчика ошибок Try...Catch.

Повышенный уровень

- Знать 1. Понятия: инкапсуляция, наследование, полиморфизм.
2. Наследование классов.
3. Полиморфизм методов.
4. Переопределение и перегрузка методов.
5. Стандартные форматные строки для даты, чувствительные к культуре.
6. Стандартные форматные строки для даты, не чувствительные к культуре
Уметь 7. Использование инструментов отладки Visual Studio .NET для установки
Владеть точек останова и исправления ошибок.
8. Проверка конкретных условий возникновения ошибок с помощью оператора Catch When.
9. Использование свойств Err.Number и Err.Description для определения исключений.

1. Критерии оценивания компетенций

Оценка «отлично» выставляется студенту, если он продемонстрировал глубокие, исчерпывающие знания и творческие способности в понимании, изложении и использовании учебно-программного материала; логически последовательные, содержательные, полные, правильные и конкретные ответы на все поставленные вопросы и дополнительные вопросы преподавателя; свободное владение основной и дополнительной литературой, рекомендованной учебной программой.

Оценка «хорошо» выставляется студенту, если он продемонстрировал твердые и достаточно полные знания всего программного материала, правильное понимание сущности и взаимосвязи рассматриваемых процессов и явлений; последовательные, правильные, конкретные ответы на поставленные вопросы при свободном устранении замечаний по отдельным вопросам; достаточное владение литературой, рекомендованной учебной программой.

Оценка «удовлетворительно» выставляется студенту, если он продемонстрировал твердые знания и понимание основного программного материала; правильные, без грубых ошибок ответы на поставленные вопросы при устранении неточностей и несущественных ошибок в освещении отдельных положений при наводящих вопросах преподавателя; недостаточное владение литературой, рекомендованной учебной программой.

Оценка «неудовлетворительно» выставляется студенту, если он продемонстрировал неправильные ответы на основные вопросы, допущены грубые ошибки в ответах, непонимание сущности излагаемых вопросов; неуверенные и неточные ответы на дополнительные вопросы.

2. Описание шкалы оценивания

Промежуточная аттестация в форме экзамена предусматривает проведение обязательной экзаменационной процедуры и оценивается 40 баллами из 100. Минимальное количество баллов, необходимое для допуска к экзамену, составляет 33 балла. Положительный ответ студента на экзамене оценивается рейтинговыми баллами в диапазоне от **20** до **40** ($20 \leq S_{\text{экс}} \leq 40$), оценка **меньше 20** баллов считается неудовлетворительной.

Шкала соответствия рейтингового балла экзамена 5-балльной системе

Рейтинговый балл по дисциплине	Оценка по 5-балльной системе
35 – 40	Отлично
28 – 34	Хорошо
20 – 27	Удовлетворительно

3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Процедура проведения экзамена осуществляется в соответствии с Положением о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся по образовательным программам высшего образования в СКФУ.

В экзаменационный билет включаются два устных вопроса. Для подготовки по билету отводится 30 минут. При выставлении оценок учитывается уровень приобретенных компетенций студента.

Компонент «знать» оценивается теоретическими вопросами по содержанию дисциплины, компоненты «уметь» и «владеть» - практико-ориентированными вопросами.

При подготовке к ответу студенту предоставляется право пользования необходимым программным обеспечением.

Составитель _____ Н.И. Битюцкая

(подпись)

« ____ » _____ 20__ г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт сервиса, туризма и дизайна (филиал) СКФУ в г. Пятигорске

УТВЕРЖДАЮ

Зав. кафедрой Систем управления и
информационных технологий
ИСТИД (филиал) СКФУ в г. Пятигорске
И.М. Першин
«__» _____ 202_ г.

Вопросы для собеседования

по дисциплине

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

№ темы	Контрольные вопросы по теме
1-2	Базовый уровень: 1. Основные парадигмы программирования. 2. Основные инструменты (окна) VisualStudio и их назначение. Закрепление, перемещение и скрытие окон инструментов. 3. Какие файлы создаются при сохранении решения? 4. Как и для чего создается исполняемый файл приложения? Повышенный уровень: 5. Принципы ООП. Инкапсуляция. Наследование. Полиморфизм.
3-4	Базовый уровень: 1. Основные типы данных Visual C#. 2. Операторы объявления локальных и глобальных переменных. 3. Работа с математическими методами среды .NET Framework. 4. Инструкция выбора if. 5. Назначение и синтаксис оператора цикла for. 6. Назначение и синтаксис оператора цикла while . 7. Назначение и синтаксис оператора цикла do... while. Повышенный уровень: 8. Инструкция выбора switch. 9. Использование управляющих символов и форматных строк.
5-6	Базовый уровень: 1. Способы объявления и инициализации одномерных и двумерных массивов. 2. Какие инструкции используются для перебора элементов массива. 3. Свойства класса Array. 4. Методы класса Array. 5. Объявление класса. Модификаторы доступа. 6. Создание методов классов. Типы возвращаемых методом значений.

	<p>7. Конструкторы классов. 8. Задание свойств класса. Повышенный уровень: 9. Полиморфизм методов.Переопределение методов. 10. Перегруженные методы. Примеры</p>
7	<p>Базовый уровень: 1. Какие методы .NET Framework используются для загрузки в текстовое поле содержимого текстового файла? 2. Какие действия необходимы для создания нового текстового файла на диске? 3. Назначение и синтаксис методов OpenRead и OpenWrite класса System.IO.File. 4. Назначение и синтаксис методов ReadAllText и ReadAllLines класса System.IO.File. 5. Назначение и синтаксис методов WriteAllText и WriteAllLines класса System.IO.File. 6. Назначение и синтаксис методов Copy, Move, Delete, Exists класса System.IO.File. 7. Назначение метода .NET Framework Length. 8. Назначение и синтаксис метода .NETFrameworkSubstring. 9. Назначение и синтаксис метода .NET Framework IndexOf. 10. Назначение и синтаксис метода .NET Framework Remove. 11. Назначение и синтаксис метода .NET Framework Insert. Повышенный уровень: 12. Назовите основные кодировки символов и их особенности. 13. Как средствами C# получить юникод данного символа и символ по данному юникоду? 14. Какие действия необходимы для шифрования текстового файла? 15. Какие действия необходимы для расшифровки текстового файла?</p>
8-9	<p>Базовый уровень: 1. Элементы управления для работы с датами. 2. Методы C# для работы с датами и временем. 3. Создание экземпляра DateTime. 4. Создание экземпляра DateTimeOffset. 5. Определение текущих даты и времени. 6. Определение в программах различных типов ошибок. 7. Использование инструментов отладки Visual Studio .NET для установки точек останова и исправления ошибок. 8. Использование окна Watch для проверки значений переменных во время выполнения программы. Использование окна Command (Окно команд) для изменения значений переменных и исполнения команд в Visual Studio. Повышенный уровень: 1. Как вычислить, сколько дней, месяцев или лет прошло между двумя датами? 2. Как определить дату, отстоящую от заданной даты на определенное число дней, месяцев или лет? 3. Исправление ошибок времени исполнения с помощью обработчика ошибок Try...Catch.</p>

1. Критерии оценивания компетенций

Оценка «отлично» выставляется студенту, если он в ходе собеседования правильно ответил на все вопросы по теме собеседования, сопровождая ответы наглядными примерами.

Оценка «хорошо» выставляется студенту, если он в ходе собеседования правильно ответил на два или три вопроса по теме, при этом есть неуверенность с практическими примерами.

Оценка «удовлетворительно» выставляется студенту, если он в ходе собеседования неуверенно ответил на два вопроса по теме и не смог привести практические примеры.

Оценка «неудовлетворительно» выставляется студенту, если он ответил менее, чем на два вопроса по теме собеседования.

2. Описание шкалы оценивания

Максимально возможный балл за весь текущий контроль устанавливается равным **55**. Текущее контрольное мероприятие считается сданным, если студент получил за него не менее 60% от установленного для этого контроля максимального балла. Рейтинговый балл, выставляемый студенту за текущее контрольное мероприятие, сданное студентом в установленные графиком контрольных мероприятий сроки, определяется следующим образом:

Уровень выполнения контрольного задания	Рейтинговый балл (в % от максимального балла за контрольное задание)
Отличный	100
Хороший	80
Удовлетворительный	60
Неудовлетворительный	0

3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Собеседование со студентами проводится после выполнения ими каждой лабораторной работы и включает в себя ответы на контрольные вопросы по теме лабораторной работы.

Контрольные вопросы содержатся в задании к лабораторной работе и позволяют проверить уровень освоения компетенций ПК-8, ПК-17.

Каждому студенту предлагается ответить на два вопроса базового уровня и один вопрос повышенного уровня. При подготовке к ответу студенту предоставляется право пользования лекциями, методическими материалами к самостоятельной работе и выполнению лабораторных работ.

При оценивании ответов студента учитывается:

- знание теоретического материала по теме собеседования;
- умение объяснять код программ;
- способность приводить конкретные примеры, демонстрирующие суть вопроса.

Составитель _____ Битюцкая Н.И.
(подпись)

« ____ » _____ 20 ____ г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт сервиса, туризма и дизайна (филиал) СКФУ в г. Пятигорске

УТВЕРЖДАЮ

Зав. кафедрой Систем управления и
информационных технологий
ИСТИД (филиал) СКФУ в г. Пятигорске
_____ И.М. Першин
«__» _____ 202_ г.

Комплект заданий для лабораторных работ

по дисциплине

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

Тема лабораторной работы	Задания
Тема 1. Среда разработки VisualStudio.Net 2025. Инструменты VisualStudio.NET. Реализация алгоритмов линейной и разветвляющейся структуры.	Базовый уровень: 1. В среде VisualStudio создать консольное приложение на языке C#, которое будет вычислять значение функции в соответствии с номером варианта. 2. Предусмотреть ввод исходных данных с клавиатуры. 3. Предусмотреть проверку допустимости исходных данных. Повышенный уровень: 1. Выполнить дополнительное задание
Тема 2. Реализация алгоритмов циклической структуры.	Базовый уровень: 1. Разработать программу на C#, используя цикл с заданным числом повторений (оператор for) и с неизвестным числом повторений (операторы while и do ... while). Повышенный уровень: 1. Выполнить дополнительное задание.
Тема 3. Работа с одномерными и двумерными массивами.	Базовый уровень: 1. В среде VisualStudio создать консольное приложение на языке C# для решения задачи обработки одномерных массивов в соответствии с номером варианта. 2. Предусмотреть ввод элементов массива с клавиатуры. Повышенный уровень: 1. Выполнить дополнительное задание на обработку

	<p>двумерных массивов.</p> <p>2. Предусмотреть заполнение массива случайными числами.</p>
<p>Тема 4. Создание методов (подпрограмм).</p>	<p>Базовый уровень:</p> <ol style="list-style-type: none"> 1. Написать программу для сортировки элементов двух одномерных массивов натуральных чисел в порядке возрастания. 2. Ввод, отображение и сортировку массивов оформить в виде методов. 3. Создать метод для нахождения максимального элемента в массиве и применить его для каждого из введенных массивов. <p>Повышенный уровень:</p> <ol style="list-style-type: none"> 1. Внесите изменения в программный код так, чтобы осуществлялась сортировка по алфавиту двух массивов строкового типа.
<p>Тема 5. Создание классов.</p>	<p>Базовый уровень:</p> <ol style="list-style-type: none"> 1. Разработать консольное приложение на C# в среде Visual Studio для решения задачи в соответствии номером варианта. Создать класс. Поля и методы класса перечислены в задании. 2. Доступ к значениям полей должен осуществляться с использованием общедоступных операций set и get. 3. Включить в класс методы, обеспечивающие: <ul style="list-style-type: none"> – заполнение полей данных; – отображение (вывод) полей данных; – конструктор, который инициализирует поля класса. 4. Главная программа должна демонстрировать выполнение всех методов класса. <p>Повышенный уровень:</p> <ol style="list-style-type: none"> 1. Выполнить дополнительное задание.
<p>Тема 6. Наследование классов.</p>	<p>Базовый уровень:</p> <ol style="list-style-type: none"> 1. Разработать два варианта проектов классов в соответствии с вариантом задания: <ul style="list-style-type: none"> – проекты независимых друг от друга классов; – проекты зависимых классов с помощью наследования. 2. В каждом классе предусмотреть: <ul style="list-style-type: none"> – поля и методы, указанные в задании; – доступ к значениям полей с использованием свойств (set и get); – как минимум, два конструктора для инициализации полей класса; – метод для вывода значений полей класса. 3. Разработать два консольных приложения на C# в среде Visual Studio, реализующих каждый из вариантов проектов классов. 4. Главная программа (program) должна демонстрировать выполнение всех методов класса.

	<p>Повышенный уровень:</p> <ol style="list-style-type: none"> 1. Сравнить объёмы программы без наследования классов и программы с наследованием классов. Сделать это для исходных файлов и для исполняемых файлов, поместив результаты в таблицу.
<p>Тема 7. Полиморфизм методов.</p>	<p>Базовый уровень:</p> <ol style="list-style-type: none"> 1. Для классов, созданных в лабораторной работе № 6, предусмотреть переопределение методов в производном классе. 2. Реализовать по два конструктора в базовом и производном классах. <p>Повышенный уровень:</p> <ol style="list-style-type: none"> 1. Выполнить дополнительное задание.
<p>Тема 8. Работа с текстовыми файлами и обработка строк</p>	<p>Базовый уровень:</p> <ol style="list-style-type: none"> 1. Написать программу, которая позволяет зашифровывать и расшифровывать с помощью «шифра Цезаря» текстовый файл или сообщение, написанное на русском языке. Этот шифр реализует следующее преобразование текста: каждая русская буква исходного текста заменяется третьей после нее буквой в алфавите, который считается написанным по кругу. 2. Знаки препинания и другие символы, отличные от строчных или прописных букв русского алфавита, не шифруются. <p>Повышенный уровень:</p> <ol style="list-style-type: none"> 1. Изменить программный код так, чтобы шифровались и арабские цифры.
<p>Тема 9. Методы C# для работы с датами.</p>	<p>Базовый уровень:</p> <ol style="list-style-type: none"> 1. Написать программу «Расчет заработной платы», которая для заданного количества сотрудников вычисляет сумму к выдаче. 2. Программа должна вычислять для каждого сотрудника: начисленную сумму; сумму удержаний; сумму к выдаче. 3. Программа должна формировать итоговую таблицу расчета заработной платы по всем сотрудникам, содержащую следующие поля: ФИО; Оклад; Стаж; Возраст; Доплата; Начислено; Удержано; К выдаче. 4. Считать, что заработная плата рассчитывается за текущий месяц. 5. Доплата за стаж работы на данном предприятии вычисляется следующим образом: если стаж составляет от 0 до 5 лет, то доплата равна 10% от оклада; если стаж составляет от 5 до 10 лет, то доплата равна 30% от оклада; если стаж равен или больше 10 лет, то доплата равна 50% от оклада. 6. Удерживается со всех сотрудников 13% от начисленной суммы (НДФЛ). <p>Повышенный уровень:</p> <ol style="list-style-type: none"> 1. Внести изменения в программный код так, чтобы сотрудник, у которого день рождения в месяце начисления зарплаты, получал премию в размере 20% от оклада.

	2. Стаж и возраст считаются на 1-ое число месяца начисления.
--	--

1. Критерии оценивания компетенций

Оценка «отлично» выставляется студенту, если он правильно выполнил все задания базового и повышенного уровней.

Оценка «хорошо» выставляется студенту, если он правильно выполнил все задания только базового уровня.

Оценка «удовлетворительно» выставляется студенту, если он выполнил все задания базового уровня, но допустил незначительные ошибки, которые исправил после указания на них.

Оценка «неудовлетворительно» выставляется студенту, если он выполнил не все задания базового уровня или допустил грубые ошибки при выполнении заданий.

2. Описание шкалы оценивания

Максимально возможный балл за весь текущий контроль устанавливается равным **55**. Текущее контрольное мероприятие считается сданным, если студент получил за него не менее 60% от установленного для этого контроля максимального балла. Рейтинговый балл, выставляемый студенту за текущее контрольное мероприятие, сданное студентом в установленные графиком контрольных мероприятий сроки, определяется следующим образом:

Уровень выполнения контрольного задания	Рейтинговый балл (в % от максимального балла за контрольное задание)
Отличный	100
Хороший	80
Удовлетворительный	60
Неудовлетворительный	0

3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

По каждой лабораторной работе студент готовит письменный отчет, который должен содержать: постановку задачи, описание этапов выполнения задания, скриншоты разработанных форм, листинг программного кода основных модулей и выводы.

Предлагаемые студенту задания для лабораторных работ позволяют проверить уровень освоения компетенций ПК-8, ПК-17.

На выполнение каждой лабораторной работы и подготовку отчета отводится время в соответствии с рабочей программой дисциплины.

При выполнении лабораторных работ и подготовке отчета студенту разрешается пользоваться лекциями, методическими материалами к выполнению лабораторных работ и справочными сведениями по языку и среде разработки.

При проверке отчета по лабораторной работе учитываются:

- правильность выполнения заданий базового и повышенного уровней;
- последовательность описания этапов выполнения работы;
- рациональность использованных алгоритмов решения задачи;
- справедливость сделанных выводов.

Составитель _____ Битюцкая Н.И.
(подпись)

«___»_____20 г.

Оценочный лист

№ п/п	Ф.И.О. студента	Параметры состояния образованности								Итоговый балл
		Предметно-информационная составляющая образованности			Деятельностно-коммуникативная составляющая образованности			Ценностно-ориентационная составляющая образованности		
		Контрольно-методический срез	Общеучебные умения и навыки			Уровень развития устной речи	Умение работать с информацией	Грамотность	Умение использовать полученные знания в повседневной жизни	
Умение анализировать	Умение доказывать		Умение делать выводы							
1.										
2.										
3.										
4.										
5.										
6.										
7.										
8.										
9.										
10.										
11.										
12.										
13.										
14.										
15.										
16.										
17.										
18.										
19.										
20.										
21.										
22.										

