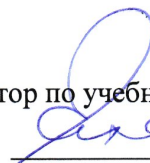


ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ
ИНКЛЮЗИВНОГО ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ГУМАНИТАРНО ЭКОНОМИЧЕСКИЙ
УНИВЕРСИТЕТ»

КАФЕДРА ЦИФРОВЫХ ТЕХНОЛОГИЙ

УТВЕРЖДАЮ

Проректор по учебно-методической работе



Е.С. Сахарчук

«27» 04 2022 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

Объектно-ориентированное программирование

наименование дисциплины

09.03.01 «Информатика и вычислительная техника»

шифр и наименование направления подготовки

Программное обеспечение вычислительной техники и информационных систем

направленность (профиль)

Москва 2022

Разработчик:

МГТЭУ, заведующий кафедрой цифровых технологий
место работы, занимаемая должность


подпись

Митрофанов Е.П.
Ф.И.О.

14.03
Дата

2020 г

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

рассмотрен и одобрен на заседании кафедры цифровых технологий

(протокол № 4 от «21» 03 2020 г.)

на заседании Учебно-методического совета МГТЭУ

(протокол № 1 от «27» 04 2022 г.)

Согласовано:

Представитель работодателя
или объединения работодателей



/ Демидов Л.Н./

к.т.н., доцент АО «Микропроцессорные системы»

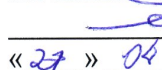
(должность, место работы)

«21» 03 2022 г.

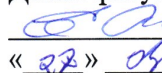
Начальник учебно-методического управления

 И.Г. Дмитриева
«27» 04 2022 г.

Начальник методического отдела

 Д.Е. Гапеенок
«27» 04 2022 г.

Декан факультета ПМИИ

 Е.В. Петрунина
«22» 04 2022 г.

Содержание

- 1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ**
- 2. ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ СРЕДСТВ**
- 3. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ**
- 4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ**
- 5. МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ**

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине ОБЪЕКТНО – ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

Оценочные средства составляются в соответствии с рабочей программой дисциплины и представляют собой совокупность контрольно-измерительных материалов (типовые задачи (задания), контрольные работы, тесты и др.), предназначенных для измерения уровня достижения обучающимися установленных результатов обучения.

Оценочные средства используются при проведении текущего контроля успеваемости и промежуточной аттестации.

1.2. Место дисциплины (модуля) в структуре образовательной программы направления подготовки

Учебная дисциплина «Объектно-ориентированное программирование» относится к части, формируемой участниками образовательных отношений блока «Дисциплин (модулей)» Б1. Изучение учебной «Объектно-ориентированное программирование» дисциплины базируется на знаниях, умениях и навыках, полученных студентами при изучении предшествующих курсов: «Основы информатики», «Алгоритмизация и программирование», «Математическая логика и теория алгоритмов», «Системное и прикладное программное обеспечение», «Компьютерный анализ». Изучение учебной дисциплины «Объектно-ориентированное программирование» необходимо для освоения таких дисциплин, как «Высокоуровневые методы информатики и программирования», «Проектный практикум» и производственной практики «Технологическая (проектнотехнологическая)».

Таблица 1 - Перечень компетенций, формируемых в процессе освоения дисциплины

Код компетенции	Наименование результата обучения
ПК – 2	Способен разрабатывать, внедрять и адаптировать прикладное программное обеспечение.
ПК – 3	Способен проектировать ИС в соответствии с профилем подготовки по видам обеспечения.
ПК – 8	Способен программировать приложения и создавать программные прототипы решения прикладных задач.

Конечными результатами освоения дисциплины являются сформированные когнитивные дескрипторы «знать», «уметь», «владеть», расписанные по отдельным компетенциям. Формирование дескрипторов происходит в течение всего семестра по этапам в рамках контактной работы, включающей различные виды занятий и самостоятельной работы, с применением различных форм и методов обучения (табл. 2).

Таблица 2 - Формирование компетенций в процессе изучения дисциплины:

Код компетенции	Уровень освоения компетенций	Индикаторы достижения компетенций	Вид учебных занятий, формы и методы обучения, способствующие формированию и развитию компетенций	Контролируемые разделы и темы дисциплины	Оценочные средства, используемые для оценки уровня сформированности компетенции
ПК-2. Способен разрабатывать, внедрять и адаптировать прикладное программное обеспечение.		Знает			
	Недостаточный уровень	программные шаблоны; метрики и риски тестирования; базовые понятия качества			
	Базовый уровень	программного продукта и качества процесса	Лекционные занятия, самостоятельная работа	Раздел 1-12	Опрос, Контрольная работа, тестовые задания
	Средний уровень	разработки программного обеспечения; основные концепции и атрибуты качества программного			
	Высокий уровень	обеспечения (надежности, безопасности, удобства использования); функциональные характеристики применения программного обеспечения.			
		Умеет			
	Недостаточный уровень	реализовывать программные продукты на языках программирования высокого уровня; описывать архитектуру программного средства включая выделение: функциональных	практические занятия, самостоятельная работа	Раздел 1-12	Опрос, Контрольная работа, тестовые задания
	Базовый уровень	компонентов и модулей, структур данных, внешних и внутренних интерфейсов;			
Средний уровень	применять соответствующие программные или аппаратные архитектурные решения;				
Высокий уровень	использовать модели данных; анализировать и оценивать архитектуру на предмет атрибутов качества.				

		Владеет			
	Недостаточный уровень	навыками планирования процесса разработки программного продукта; навыками задания функциональных рамок подсистем; навыками определения наиболее значимых критериев качества программного продукта.	практические занятия, самостоятельная работа, практическая подготовка	Раздел 1-12	Опрос, Контрольная работа, тестовые задания
	Базовый уровень				
	Средний уровень				
	Высокий уровень				
ПК-3. Способен проектировать ИС в соответствии с профилем подготовки по видам обеспечения.		Знает			
	Недостаточный уровень	методы разработки, анализа и проектирования ПО; функциональное и техническое проектирование; паттерны проектирования; номенклатуру инструментальных средств, поддерживающих создание программного обеспечения; техники распределения задач на разработку между исполнителями.	Лекционные занятия, самостоятельная работа	Раздел 1-12	Опрос, Контрольная работа, тестовые задания
	Базовый уровень				
	Средний уровень				
	Высокий уровень				
		Умеет			
	Недостаточный уровень	работать с инструментальными средствами, поддерживающими создание программного обеспечения для информационных систем; использовать нотации для построения функциональной и процессной моделей исследуемой предметной области; проектировать компоненты программных средств.	Практические занятия, самостоятельная работа	Раздел 1-12	Опрос, Контрольная работа, тестовые задания
	Базовый уровень				
	Средний уровень				
	Высокий уровень				
		Владеет			
Недостаточный уровень	Владеет навыками определения содержания работ по созданию программного продукта; приемами работы с	Практические занятия, самостоятельная	Раздел 1-12	Опрос, Контрольная работа, тестовые задания	
Базовый уровень					

	Средний уровень	инструментальными средствами автоматизации проектирования и реализации программного продукта;	работа, практическая подготовка		
	Высокий уровень	навыками проектирования прикладных			
	Базовый уровень	программных продуктов, в том числе клиент-серверных приложений.			
	Средний уровень				
	Высокий уровень				
		Знает			
ПК-8. Способен программировать приложения и создавать программные прототипы решения прикладных задач.	Недостаточный уровень	основные языки программирования приложений; теоретические и методические основы технологии программирования, анализа и применения алгоритмических и программных решений;	Лекционные занятия, самостоятельная работа	Раздел 1-12	Опрос, Контрольная работа, тестовые задания
	Базовый уровень	методы и приёмы разработки программных прототипов решения прикладных задач.			
	Средний уровень				
	Высокий уровень				
		Умеет			
	Недостаточный уровень	реализовывать программные продукты для решения прикладных задач;	Практические занятия, самостоятельная работа	Раздел 1-12	Опрос, Контрольная работа, тестовые задания
	Базовый уровень	описывать архитектуру приложений включая выделение функциональных компонентов и модулей, структур данных, внешних и внутренних интерфейсов.			
	Средний уровень				
	Высокий уровень				
		Владеет			
	Недостаточный уровень	навыками разработки приложений и программных прототипов.	Практические занятия, самостоятельная	Раздел 1-12	Опрос, Контрольная работа, тестовые задания
	Базовый уровень				

Средний уровень	работа, практическая подготовка		
Высокий уровень			

2. ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ СРЕДСТВ¹

Таблица 3

№	Наименование оценочного средства	Характеристика оценочного средства	Представление оценочного средства в ФОС
	Опрос	Средство контроля усвоения учебного материала темы, раздела или разделов дисциплины, организованное как учебное занятие в виде собеседования преподавателя с обучающимися.	Вопросы по темам/разделам дисциплины
	Тест	Средство, позволяющее оценить уровень знаний обучающегося путём выбора им одного из нескольких вариантов ответа на поставленный вопрос. Возможно использование тестовых вопросов, предусматривающих ввод обучающимися короткого и однозначного ответа на поставленный вопрос.	Тестовые задания
	Контрольная работа	Оценочное средство, ориентированное на выполнение комплексной работы, освещающей несколько аспектов предмета дисциплины (факультатива)	Задание для выполнения контрольной работы

**Приведенный перечень оценочных средств при необходимости может быть дополнен.*

¹ Указываются оценочные средства, применяемые в ходе реализации рабочей программы данной дисциплины.

3. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

Оценивание результатов обучения по дисциплине Информатика осуществляется в соответствии с Положением о текущем контроле успеваемости и промежуточной аттестации обучающихся.

Предусмотрены следующие виды контроля: текущий контроль (осуществление контроля всех видов аудиторной и внеаудиторной деятельности обучающегося с целью получения первичной информации о ходе усвоения отдельных элементов содержания дисциплины) и промежуточная аттестация (оценивается уровень и качество подготовки по дисциплине в целом).

Показатели и критерии оценивания компетенций, формируемых в процессе освоения данной дисциплины, описаны в табл. 4.

Таблица 4.

Код компетенции	Уровень освоения компетенций	Индикаторы достижения компетенций	Критерии оценивания результатов обучения
ПК-2. Способен разрабатывать, внедрять и адаптировать прикладное программное обеспечение		Знает	
	Недостаточный уровень	программные шаблоны; метрики и риски тестирования; базовые понятия качества программного продукта и качества процесса разработки программного обеспечения; основные концепции и атрибуты качества программного обеспечения (надежности, безопасности, удобства использования);	Не знает программные шаблоны; метрики и риски тестирования; базовые понятия качества программного продукта и качества процесса разработки программного обеспечения; основные концепции и атрибуты качества программного обеспечения (надежности, безопасности, удобства использования); функциональные характеристики применения программного обеспечения.
	Базовый уровень	функциональные характеристики применения программного обеспечения.	Знает программные шаблоны; метрики и риски тестирования; базовые понятия качества программного продукта и качества процесса разработки программного обеспечения; основные концепции и атрибуты качества программного обеспечения (надежности, безопасности, удобства использования); функциональные характеристики применения программного обеспечения.

	Средний уровень		Хорошо знает программные шаблоны; метрики и риски тестирования; базовые понятия качества программного продукта и качества процесса разработки программного обеспечения;
			основные концепции и атрибуты качества программного обеспечения (надежности, безопасности, удобства использования); функциональные характеристики применения программного обеспечения.
	Высокий уровень		Отлично знает программные шаблоны; метрики и риски тестирования; базовые понятия качества программного продукта и качества процесса разработки программного обеспечения; основные концепции и атрибуты качества программного обеспечения (надежности, безопасности, удобства использования); функциональные характеристики применения программного обеспечения.
		Умеет	
	Недостаточный уровень	реализовывать программные продукты на языках программирования высокого уровня; описывать архитектуру программного средства включая выделение: функциональных компонентов и модулей, структур данных, внешних и внутренних интерфейсов; применять соответствующие программные или аппаратные архитектурные	Не умеет реализовывать программные продукты на языках программирования высокого уровня; описывать архитектуру программного средства включая выделение: функциональных компонентов и модулей, структур данных, внешних и внутренних интерфейсов; применять соответствующие программные или аппаратные архитектурные решения; использовать модели данных; анализировать и оценивать архитектуру на предмет атрибутов качества.

Базовый уровень	решения; использовать модели данных; анализировать и оценивать архитектуру на предмет атрибутов качества.	Умеет реализовывать программные продукты на языках программирования высокого уровня; описывать архитектуру программного средства включая выделение: функциональных компонентов и модулей, структур данных, внешних и внутренних интерфейсов; применять соответствующие программные или аппаратные архитектурные решения; использовать модели данных; анализировать и оценивать архитектуру на предмет атрибутов качества.
Средний		Хорошо умеет реализовывать

уровень		программные продукты на языках программирования высокого уровня; описывать архитектуру программного средства включая выделение: функциональных компонентов и модулей, структур данных, внешних и внутренних интерфейсов; применять соответствующие программные или аппаратные архитектурные решения; использовать модели данных; анализировать и оценивать архитектуру на предмет атрибутов качества.
Высокий уровень		Отлично умеет реализовывать программные продукты на языках программирования высокого уровня; описывать архитектуру программного средства включая выделение: функциональных компонентов и модулей, структур данных, внешних и внутренних интерфейсов; применять соответствующие программные или аппаратные архитектурные решения; использовать модели данных; анализировать и оценивать архитектуру на предмет атрибутов качества.
	Владеет	

	Недостаточный уровень	навыками планирования процесса разработки программного продукта; навыками задания функциональных рамок подсистем; навыками определения наиболее значимых критериев качества программного продукта.	Не владеет навыками планирования процесса разработки программного продукта; навыками задания функциональных рамок подсистем; навыками определения наиболее значимых критериев качества программного продукта.
	Базовый уровень	качества программного продукта.	Владеет навыками планирования процесса разработки программного продукта; навыками задания функциональных рамок подсистем; навыками определения наиболее значимых критериев качества программного продукта.
	Средний уровень		Хорошо владеет навыками планирования процесса разработки программного продукта; навыками задания функциональных рамок подсистем; навыками определения наиболее значимых критериев

			качества программного продукта.
	Высокий уровень		Отлично владеет навыками планирования процесса разработки программного продукта; навыками задания функциональных рамок подсистем; навыками определения наиболее значимых критериев качества программного продукта.
ПК-3. Способен проектировать ИС в соответствии с профилем подготовки по видам обеспечения.		Знает	
	Недостаточный уровень	методы разработки, анализа и проектирования ПО; функциональное и техническое проектирование; паттерны проектирования; номенклатуру инструментальных средств, поддерживающих создание программного обеспечения;	Не знает методы разработки, анализа и проектирования ПО; функциональное и техническое проектирование; паттерны проектирования; номенклатуру инструментальных средств, поддерживающих создание программного обеспечения; техники распределения задач на разработку между исполнителями.

Базовый уровень	техники распределения задач на разработку между исполнителями.	Знает методы разработки, анализа и проектирования ПО; функциональное и техническое проектирование; паттерны проектирования; номенклатуру инструментальных средств, поддерживающих создание программного обеспечения; техники распределения задач на разработку между исполнителями.
Средний уровень		Хорошо знает методы разработки, анализа и проектирования ПО; функциональное и техническое проектирование; паттерны проектирования; номенклатуру инструментальных средств, поддерживающих создание программного обеспечения; техники распределения задач на разработку между исполнителями.
Высокий уровень		Отлично знает методы разработки, анализа и проектирования ПО; функциональное и техническое проектирование; паттерны проектирования; номенклатуру инструментальных средств, поддерживающих создание программного обеспечения; техники распределения задач на разработку между исполнителями.

Недостаточный уровень	Умеет работать с инструментальными средствами, поддерживающими создание программного обеспечения для информационных систем; использовать нотации для построения функциональной и процессной моделей	Не умеет работать с инструментальными средствами, поддерживающими создание программного обеспечения для информационных систем; использовать нотации для построения функциональной и процессной моделей исследуемой предметной области; проектировать компоненты программных средств.
-----------------------	---	--

Базовый уровень	исследуемой предметной области; проектировать компоненты программных средств.	Умеет работать с инструментальными средствами, поддерживающими создание программного обеспечения для информационных систем; использовать нотации для построения функциональной и процессной моделей исследуемой предметной области; проектировать компоненты программных средств.
Средний уровень		Хорошо умеет работать с инструментальными средствами, поддерживающими создание программного обеспечения для информационных систем; использовать нотации для построения функциональной и процессной моделей исследуемой предметной области; проектировать компоненты программных средств.
Высокий уровень		Отлично умеет работать с инструментальными средствами, поддерживающими создание программного обеспечения для информационных систем; использовать нотации для построения функциональной и процессной моделей исследуемой предметной области; проектировать компоненты программных средств.
	Владеет	
Недостаточный уровень	навыками определения содержания работ по созданию программного продукта; приемами работы с инструментальными средствами автоматизации	Не владеет навыками определения содержания работ по созданию программного продукта; приемами работы с инструментальными средствами автоматизации проектирования и

	проектирования и реализации программ: продукта; навыками проектирования прикладных программных продукт	реализации программного продукта; навыками проектирования прикладных программных продуктов, в том числе клиент-серверных приложений.
--	--	--

	Базовый уровень	в том числе клиент-серверных приложений	Владеет навыками определения содержания работ по созданию программного продукта; приемами работы с инструментальными средствами автоматизации проектирования и реализации программного продукта; навыками проектирования прикладных программных продуктов, в том числе клиент-серверных приложений.
	Средний уровень		Хорошо владеет навыками определения содержания работ по созданию программного продукта; приемами работы с инструментальными средствами автоматизации проектирования и реализации программного продукта; навыками проектирования прикладных программных продуктов, в том числе клиент-серверных приложений.
	Высокий уровень		Владеет на высоком уровне навыками определения содержания работ по созданию программного продукта; приемами работы с инструментальными средствами автоматизации проектирования и реализации программного продукта; навыками проектирования прикладных программных продуктов, в том числе клиент-серверных приложений.
		Знает	
ПК-8. Способен программировать приложения и создавать программные прототипы решения	Недостаточный уровень	основные языки программирования приложений; теоретические и методические основы технологии программирования, анализа и применения алгоритмических и	Не знает основные языки программирования приложений; теоретические и методические основы технологии программирования, анализа и применения алгоритмических и программных решений; методы и приёмы разработки программных

прикладных задач		программных решений; методы и приёмы разработки программных прототипов решения прикладных задач.	прототипов решения прикладных задач.
	Базовый уровень		Знает основные языки программирования приложений; теоретические и методические основы технологии программирования, анализа и применения алгоритмических и программных решений; методы и приёмы разработки программных прототипов решения прикладных задач.
	Средний уровень		Хорошо знает основные языки программирования приложений; теоретические и методические основы технологии программирования, анализа и применения алгоритмических и программных решений; методы и приёмы разработки программных прототипов решения прикладных задач.
	Высокий уровень		Отлично знает основные языки программирования приложений; теоретические и методические основы технологии программирования, анализа и применения алгоритмических и программных решений; методы и приёмы разработки программных прототипов решения прикладных задач.
		Умеет	
Недостаточный уровень	реализовывать программные продукты для решения прикладных задач; описывать архитектуру приложений включая выделение функциональных компонентов и модулей, структур данных, внешних и внутренних интерфейсов.	Не умеет реализовывать программные продукты для решения прикладных задач; описывать архитектуру приложений включая выделение функциональных компонентов и модулей, структур данных, внешних и внутренних интерфейсов.	

	Базовый уровень		Умеет реализовывать программные продукты для решения прикладных задач; описывать архитектуру приложений включая выделение функциональных компонентов и модулей, структур данных, внешних и внутренних интерфейсов.
	Средний уровень		Хорошо умеет реализовывать программные продукты для решения прикладных задач; описывать архитектуру приложений включая выделение функциональных компонентов и модулей, структур данных, внешних и внутренних интерфейсов.
	Высокий уровень		Отлично умеет реализовывать программные продукты для решения прикладных задач; описывать архитектуру приложений включая выделение функциональных компонентов и модулей, структур данных, внешних и внутренних интерфейсов.
		Владеет	
	Недостаточный уровень	навыками разработки приложений и программных прототипов	Не владеет навыками разработки приложений и программных прототипов
	Базовый уровень		Владеет навыками разработки приложений и программных прототипов
	Средний уровень		Хорошо владеет навыками разработки приложений и программных прототипов
Высокий уровень	Владеет на высоком уровне навыками разработки приложений и программных прототипов		

4. Методические материалы, определяющие процедуры оценивания результатов обучения

По видам заданий приводится описание того, каким образом необходимо выполнить данное задание, способы и механизмы его выполнения, выбор номера варианта и др. Примеры методических материалов, определяющих процедуру оценивания результатов обучения, характеризующих этапы формирования компетенций:

- Кейсовые технологии как средство формирования компетенций

- Методические указания по разработке оценочных средств
- Разработка и применение деловых игр
- Формирование портфолио, обучающегося как современная оценочная технология
- Иные методические материалы, определяющие процедуры оценивания результатов обучения в ходе реализации рабочей программы дисциплины

5.Материалы для проведения текущего контроля и промежуточной аттестации

Тематика рефератов, проектов, творческих заданий, эссе и т.п.

Не предусмотрено

Вопросы к зачету

1. Назовите основные принципы ООП.
2. Назовите составные части модели известной как трёхуровневое приложение.
3. Назначение .Net Framework?
4. Что такое managed code?
5. Что такое assembly?
6. Поддерживает ли C# множественное наследование?
7. Назовите класс .NET, от которого наследуются все классы.
8. Назовите различия между структурами и классами.
9. Какой синтаксис используется для указания класса родителя в C#?
10. Что такое абстрактный класс?
11. Что такое интерфейс класса?
12. Что означает ключевое слово «virtual» для метода или свойства?
13. Что такое делегат?
14. Как отсортировать элементы массива в убывающем порядке?
15. Чем перекрытый метод отличается от перегруженного метода?
16. Что такое JIT-компиляция?
17. Что такое «сборщик мусора»?
18. Назовите методы явного преобразования типов.
19. Что такое обобщенные классы?
20. Что такое одномерный массив, многомерный массив, зубчатый массив?
21. Назовите основные типы коллекций.
22. Как перехватить исключение в C#?
23. Опишите преимущества использования LINQ.

Вопросы к экзамену

1. Эволюция методологий программирования. Парадигмы программирования.
2. Основные принципы объектного подхода.
3. Общезыковая исполняющая среда (CLR). Промежуточный язык (IL). Сборки. Классы .NET.
4. Классы и структуры. Статические классы. Класс Object (C#).

5. Концепция наследования. Типы наследования. Наследование реализации.
 6. Массивы в C#. Класс Array. Использование массивов в качестве параметров.
 7. Операции. Безопасность типов. Проверка объектов на предмет равенства.
 8. Интерфейсы и типы коллекций. Списки. Словари. Множества.
- Битовые массивы
9. Проектирование с использованием объектов.
 10. Проектирование с использованием библиотек и шаблонов.
 11. LINQ. Стандартные операции запросов.
 12. Классы исключений. Перехват исключений. Классы исключений.
 13. Жизненный цикл объекта. Конструкторы и деструкторы
 14. Объектно-ориентированное визуальное программирование интерфейсов пользователя в .NET (Windows Forms: окна, элементы управления и компоненты).
 15. Инфраструктура Windows Presentation Foundation (WPF) как альтернатива Windows Forms.
 16. Расширяемый язык разметки для приложений (XAML).
 17. Создание документов с помощью WPF.
 18. Объектная модель ADO.NET. Подключение к БД.
 19. Объектная модель ADO.NET. Выполнение команд.
 20. Привязка данных к элементам управления Windows Forms.
 21. Платформа .NET Framework для Web-приложений. ASP.NET Web Forms: модель страницы ASPX, мастер-страницы, навигация.
 22. Привязка данных к элементам управления ASP.NET Web Forms.
 23. Платформа .NET Framework для Web-приложений. ASP.NET Web Forms и ASP.NET MVC.
 24. Асинхронное программирование.
 25. Интерфейсы пользователя.
 26. Перегрузка и переопределение методов класса.
 27. Методы для установления и получения значений частных свойств класса.
 28. Обобщенные классы. 29. Абстрактные классы.
 30. Объектно-ориентированная технология доступа к данным ADO.NET

Entity Framework

Контролируемые компетенции: ПК – 2, ПК – 3, ПК – 8

Оценка компетенций осуществляется в соответствии с таблицей 4

