

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
"РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ
В.Ф. УТКИНА"



Прикладное программирование
рабочая программа дисциплины (модуля)

Закреплена за кафедрой	Автоматика и информационные технологии в управлении
Учебный план	12.05.01_20_00.plx
Квалификация	Специальность 12.05.01 Электронные и опико-электронные приборы и системы специального назначения инженер
Форма обучения	очная
Общая трудоемкость	8 ЗЕТ

Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	7 (4.1)		8 (4.2)		Итого	
	Неделя		16			
Вид занятий	уп	рп	уп	рп	уп	рп
Лекции	32	32	32	32	64	64
Лабораторные	16	16			16	16
Практические	16	16	32	32	48	48
Иная контактная работа	0,25	0,25	0,35	0,35	0,6	0,6
Консультирование перед экзаменом и практикой			2	2	2	2
Итого ауд.	64,25	64,25	66,35	66,35	130,6	130,6
Контактная работа	64,25	64,25	66,35	66,35	130,6	130,6
Сам. работа	71	71	42	42	113	113
Часы на контроль	8,75	8,75	35,65	35,65	44,4	44,4
Итого	144	144	144	144	288	288

г. Рязань

Программу составил(и):

к.т.н., доц., Корепанов Семен Евгеньевич

Корс

Рабочая программа дисциплины

Прикладное программирование

разработана в соответствии с ФГОС ВО:

ФГОС ВО - специалитет по специальности 12.05.01 Электронные и оптико-электронные приборы и системы специального назначения (приказ Минобрнауки России от 09.02.2018 г. № 93)

составлена на основании учебного плана:

Специальность 12.05.01 Электронные и оптико-электронные приборы и системы специального назначения
утвержденного учёным советом вуза от 30.01.2020 протокол № 6.

Рабочая программа одобрена на заседании кафедры

Автоматика и информационные технологии в управлении

Протокол от 04.06 2020 г. № 6

Срок действия программы: 2020-2021 уч.г.



Визирование РПД для исполнения в очередном учебном году

Рабочая программа пересмотрена, обсуждена и одобрена для исполнения в 2021-2022 учебном году на заседании кафедры
Автоматика и информационные технологии в управлении

Протокол от _____ 2021 г. № ____

Зав. кафедрой _____

Визирование РПД для исполнения в очередном учебном году

Рабочая программа пересмотрена, обсуждена и одобрена для исполнения в 2022-2023 учебном году на заседании кафедры
Автоматика и информационные технологии в управлении

Протокол от _____ 2022 г. № ____

Зав. кафедрой _____

Визирование РПД для исполнения в очередном учебном году

Рабочая программа пересмотрена, обсуждена и одобрена для исполнения в 2023-2024 учебном году на заседании кафедры
Автоматика и информационные технологии в управлении

Протокол от _____ 2023 г. № ____

Зав. кафедрой _____

Визирование РПД для исполнения в очередном учебном году

Рабочая программа пересмотрена, обсуждена и одобрена для исполнения в 2024-2025 учебном году на заседании кафедры

Автоматика и информационные технологии в управлении

Протокол от _____ 2024 г. № ____

Зав. кафедрой _____

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)	
1.1	Целью освоения дисциплины «Прикладное программирование» является формирование у будущих специалистов теоретических знаний, практических навыков и умений в части создания консольных и оконных приложений на языке C# с использованием технологии .NET и применением современных подходов к разработке программного обеспечения, а также паттернов проектирования.
1.2	Задачи дисциплины: знакомство студентов с технологиями .NET, обучение программированию на языке C#, обучение современным подходам к разработке программного обеспечения с использованием паттернов проектирования.

2. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ	
Цикл (раздел) ОП:	Б1.О
2.1	Требования к предварительной подготовке обучающегося:
2.1.1	Объектно-ориентированное программирование
2.1.2	Основы информационной безопасности
2.1.3	Программирование в системе MATLAB
2.1.4	Программирование и основы алгоритмизации
2.1.5	Информатика
2.1.6	Ознакомительная практика
2.1.7	Учебная практика
2.2	Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:
2.2.1	Базы данных
2.2.2	Выполнение и защита выпускной квалификационной работы
2.2.3	Научно-исследовательская работа
2.2.4	Преддипломная практика
2.2.5	Производственная практика

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)	
ОПК-3: Способен использовать современные информационные технологии и программное обеспечение при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	
ОПК-3.2. Использует программное обеспечение при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	
Знать Современные программные средства для решения задач профессиональной деятельности.	
Уметь Использовать программное обеспечение при решении задач профессиональной деятельности, соблюдая требования информационной безопасности.	
Владеть Современными программными средствами при решении задач профессиональной деятельности, соблюдая требования информационной безопасности.	

В результате освоения дисциплины (модуля) обучающийся должен

3.1	Знать:
3.1.1	Современные подходы к разработке ПО.
3.1.2	Знать основы процедурного, объектно-ориентированного и событийно-ориентированного подхода к программированию на языке C#.
3.1.3	Понятие потоков и коллекций на языке C#. Основы LINQ.
3.1.4	Знать основные паттерны (шаблоны) проектирования.
3.1.5	Основы тестирования ПО.
3.2	Уметь:
3.2.1	Применять свои знания при решении задач профессиональной деятельности, соблюдая требования информационной безопасности.
3.2.2	Работать с системой контроля версий Git.
3.2.3	Работать с оконными приложениями System.Windows.Forms
3.2.4	Писать unit-тесты для разрабатываемых приложений.
3.3	Владеть:

3.3.1 Навыками разработки ПО на языке C# в среде Visual Studio при решении задач профессиональной деятельности, соблюдая требования информационной безопасности.

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Компетенции	Литература	Форма контроля
	Раздел 1. Программирование на языке C#					
1.1	Платформа .NET /Тема/	7	0			
1.2	/Лек/	7	2	ОПК-3.2-3 ОПК-3.2-У ОПК-3.2-В	Л1.1 Л1.4 Л1.6 Л1.7Л2.10 Э1 Э2 Э3 Э4 Э5	Зачет
1.3	/Ср/	7	4	ОПК-3.2-3 ОПК-3.2-У ОПК-3.2-В	Л1.1 Л1.4 Л1.6 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.4	Основы языка программирования C# /Тема/	7	0			
1.5	/Лек/	7	2	ОПК-3.2-3 ОПК-3.2-В ОПК-3.2-У	Л1.1 Л1.2 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.6	/Лаб/	7	4	ОПК-3.2-3 ОПК-3.2-У ОПК-3.2-В	Л3.1 Э1 Э2 Э3 Э4 Э5	Лабораторная работа
1.7	/Ср/	7	5	ОПК-3.2-У ОПК-3.2-3 ОПК-3.2-В	Л1.1 Л1.2 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.8	Операции и операторы /Тема/	7	0			
1.9	/Лек/	7	2	ОПК-3.2-3 ОПК-3.2-У ОПК-3.2-В	Л1.1 Л1.2 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.10	/Лаб/	7	4	ОПК-3.2-У ОПК-3.2-3 ОПК-3.2-В	Л3.1 Э1 Э2 Э3 Э4 Э5	Лабораторная работа
1.11	/Ср/	7	4	ОПК-3.2-3 ОПК-3.2-У ОПК-3.2-В	Л1.1 Л1.2 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.12	Процедурное и объектно-ориентированное программирование. Классы. Инкапсуляция. /Тема/	7	0			
1.13	/Лек/	7	2	ОПК-3.2-У	Л1.1 Л1.2 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.14	/Лаб/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л3.1 Э1 Э2 Э3 Э4 Э5	Лабораторная работа
1.15	/Ср/	7	6	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1 Л1.2 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.16	Массивы и индексы /Тема/	7	0			

1.17	/Лек/	7	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1 Л1.2 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.18	/Лаб/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л3.1 Э1 Э2 Э3 Э4 Э5	Лабораторная работа
1.19	/Ср/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1 Л1.2 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.20	Наследование. Полиморфизм. /Тема/	7	0			
1.21	/Лек/	7	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1 Л1.2 Л1.3 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.22	/Ср/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1 Л1.2 Л1.3 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.23	Абстрактные классы. Интерфейсы. /Тема/	7	0			
1.24	/Лек/	7	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.25	/Пр/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.26	/Ср/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.27	Символы. Строки. Регулярные выражения. /Тема/	7	0			
1.28	/Лек/	7	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.29	/Пр/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.30	/Ср/	7	5	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.31	Статические и вложенные классы. /Тема/	7	0			
1.32	/Лек/	7	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.33	/Ср/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.34	Класс Object. Перегрузка операторов. Обработка исключений. /Тема/	7	0			
1.35	/Лек/	7	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет

1.36	/Ср/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.37	Структуры. Перечисления. /Тема/	7	0			
1.38	/Лек/	7	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1 Л1.2 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.39	/Ср/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1 Л1.2 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.40	Делегаты. /Тема/	7	0			
1.41	/Лек/	7	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.42	/Ср/	7	5	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.43	События. /Тема/	7	0			
1.44	/Лек/	7	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Л1.4 Л1.7Л2.3 Э1 Э2 Э3 Э4 Э5	Зачет
1.45	/Пр/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Л1.4 Л1.7Л2.3 Э1 Э2 Э3 Э4 Э5	Зачет
1.46	/Ср/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Л1.4 Л1.7Л2.3 Э1 Э2 Э3 Э4 Э5	Зачет
1.47	Обобщения. Ограничения. /Тема/	7	0			
1.48	/Лек/	7	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.49	/Ср/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.50	Коллекции. /Тема/	7	0			
1.51	/Лек/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.52	/Пр/	7	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
1.53	/Ср/	7	10	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.3 Л1.4 Л1.7 Э1 Э2 Э3 Э4 Э5	Зачет
	Раздел 2. Промежуточная аттестация					
2.1	Подготовка к зачету, иная контактная работа /Тема/	7	0	<все>		
2.2	Сдача зачета /ИКР/	7	0,25			

2.3	Подготовка к зачету /Зачёт/	7	8,75		Л1.1 Л1.2 Л1.3 Л1.4 Л1.6 Л1.7Л2.3 Л2.10 Э1 Э2 Э3 Э4 Э5	Зачет
	Раздел 3. Динамическая типизация. Атрибуты. Поток.					
3.1	Анонимные и динамические типы. LINQ. /Тема/	8	0			
3.2	/Лек/	8	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1Л2.6 Э1 Э2 Э3 Э4 Э5	Экзамен
3.3	/Пр/	8	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1Л2.6 Э1 Э2 Э3 Э4 Э5	Экзамен
3.4	/Ср/	8	2,5	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.1Л2.6 Э1 Э2 Э3 Э4 Э5	Экзамен
3.5	Пространства имен. Директивы препроцессора. Атрибуты. /Тема/	8	0			
3.6	/Лек/	8	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Л1.7 Э1 Э2 Э3 Э4 Э5	Экзамен
3.7	/Ср/	8	2,5	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Л1.7 Э1 Э2 Э3 Э4 Э5	Экзамен
3.8	Понятие потока. /Тема/	8	0			
3.9	/Лек/	8	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.7Л2.8 Э1 Э2 Э3 Э4 Э5	Экзамен
3.10	/Пр/	8	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.7Л2.8 Э1 Э2 Э3 Э4 Э5	Экзамен
3.11	/Ср/	8	2,5	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.7Л2.8 Э1 Э2 Э3 Э4 Э5	Экзамен
	Раздел 4. Современные подходы к разработке ПО. Системы контроля версий.					
4.1	Технологии разработки ПО. /Тема/	8	0			
4.2	/Лек/	8	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.6Л2.1 Л2.2 Э1 Э2 Э3 Э4 Э5	Экзамен
4.3	/Ср/	8	2,5	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.6Л2.1 Л2.2 Э1 Э2 Э3 Э4 Э5	Экзамен
4.4	Системы контроля версий. /Тема/	8	0			
4.5	/Лек/	8	2	ОПК-3.2-3 ОПК-3.2-У ОПК-3.2-В	Л1.5 Э1 Э2 Э3 Э4 Э5 Э6	Экзамен
4.6	/Пр/	8	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5 Э1 Э2 Э3 Э4 Э5 Э6	Экзамен
4.7	/Ср/	8	2,5	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5 Э1 Э2 Э3 Э4 Э5 Э6	Экзамен
4.8	Основы тестирования ПО. /Тема/	8	0			

4.9	/Лек/	8	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5 Л1.6Л2.2 Л2.5 Э1 Э2 Э3 Э4 Э5	Экзамен
4.10	/Пр/	8	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5 Л1.6Л2.2 Л2.5 Э1 Э2 Э3 Э4 Э5	Экзамен
4.11	/Ср/	8	3	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5 Л1.6Л2.2 Л2.5 Э1 Э2 Э3 Э4 Э5	Экзамен
Раздел 5. Паттерны проектирования ПО						
5.1	Порождающие паттерны. /Тема/	8	0			
5.2	/Лек/	8	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5Л2.7 Л2.9 Э1 Э2 Э3 Э4 Э5	Экзамен
5.3	/Пр/	8	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5Л2.7 Л2.9 Э1 Э2 Э3 Э4 Э5	Экзамен
5.4	/Ср/	8	5	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5Л2.7 Л2.9 Э1 Э2 Э3 Э4 Э5	Экзамен
5.5	Структурные паттерны. /Тема/	8	0			
5.6	/Лек/	8	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5Л2.7 Л2.9 Э1 Э2 Э3 Э4 Э5	Экзамен
5.7	/Пр/	8	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5Л2.7 Л2.9 Э1 Э2 Э3 Э4 Э5	Экзамен
5.8	/Ср/	8	5	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5Л2.7 Л2.9 Э1 Э2 Э3 Э4 Э5	Экзамен
5.9	Поведенческие паттерны. /Тема/	8	0			
5.10	/Лек/	8	6	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5Л2.7 Л2.9 Э1 Э2 Э3 Э4 Э5	Экзамен
5.11	/Пр/	8	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5Л2.7 Л2.9 Э1 Э2 Э3 Э4 Э5	Экзамен
5.12	/Ср/	8	9	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.5Л2.7 Л2.9 Э1 Э2 Э3 Э4 Э5	Экзамен
Раздел 6. Создание оконных приложений Windows Forms						
6.1	Создание окон с помощью System.Windows.Forms. /Тема/	8	0			
6.2	/Лек/	8	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.4 Л1.7Л2.4 Э1 Э2 Э3 Э4 Э5	Экзамен

6.3	/Пр/	8	4	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.4 Л1.7Л2.4 Э1 Э2 Э3 Э4 Э5	Экзамен
6.4	/Ср/	8	2,5	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.4 Л1.7Л2.4 Э1 Э2 Э3 Э4 Э5	Экзамен
6.5	Визуализация графических данных средствами GDI+ /Тема/	8	0			
6.6	/Лек/	8	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Э1 Э2 Э3 Э4 Э5	Экзамен
6.7	/Ср/	8	2,5	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.3 Э1 Э2 Э3 Э4 Э5	Экзамен
6.8	Использование элементов управления System.Windows.Forms /Тема/	8	0			
6.9	/Лек/	8	2	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.7 Э1 Э2 Э3 Э4 Э5	Экзамен
6.10	/Ср/	8	2,5	ОПК-3.2-В ОПК-3.2-3 ОПК-3.2-У	Л1.2 Л1.7 Э1 Э2 Э3 Э4 Э5	Экзамен
Раздел 7. Промежуточная аттестация						
7.1	Подготовка к экзамену, иная контактная работа /Тема/	8	0	<все>		
7.2	Сдача экзамена /ИКР/	8	0,35			
7.3	Консультация перед экзаменом /Кнс/	8	2			
7.4	Подготовка к экзамену /Экзамен/	8	35,65		Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7Л2.1 Л2.2 Л2.3 Л2.4 Л2.5 Л2.6 Л2.7 Л2.8 Л2.9 Л2.10 Э1 Э2 Э3 Э4 Э5	Экзамен

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Оценочные материалы приведены в приложении к рабочей программе дисциплины (см. документ "Оценочные материалы дисциплины "Прикладное программирование")

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

6.1. Рекомендуемая литература

6.1.1. Основная литература

№	Авторы, составители	Заглавие	Издательство, год	Количество/название ЭБС
Л1.1	Васильев В. Н.	Основы программирования на языке C# : учебное пособие	Волгоград: Волгоградский институт бизнеса, 2010, 70 с.	978-5-9061-7234-1, http://www.iprbookshop.ru/11341.html

№	Авторы, составители	Заглавие	Издательство, год	Количество/название ЭБС
Л1.2	Агапов В. П.	Основы программирования на языке С# : учебное пособие	Москва: Московский государственный строительный университет, ЭБС АСВ, 2012, 128 с.	978-5-7264-0576-6, http://www.iprbookshop.ru/16366.html
Л1.3	Котов О. М.	Язык С#. Краткое описание и введение в технологии программирования : учебное пособие	Екатеринбург: Уральский федеральный университет, ЭБС АСВ, 2014, 208 с.	978-5-7996-1094-4, http://www.iprbookshop.ru/68524.html
Л1.4	Биллиг В. А.	Основы объектного программирования на С# (С# 3.0, Visual Studio 2008) : учебное пособие	Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017, 583 с.	978-5-4487-0145-0, http://www.iprbookshop.ru/72339.html
Л1.5	Зубкова Т. М.	Технология разработки программного обеспечения : учебное пособие	Оренбург: Оренбургский государственный университет, ЭБС АСВ, 2017, 469 с.	978-5-7410-1785-2, http://www.iprbookshop.ru/78846.html
Л1.6	Павлова Е. А.	Технологии разработки современных информационных систем на платформе Microsoft.NET : учебное пособие	Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020, 128 с.	978-5-4497-0360-6, http://www.iprbookshop.ru/89479.html
Л1.7	Байдачный С. С.	.NET Framework 2.0. Секреты создания Windows-приложений	Москва: СОЛОН-Пресс, 2016, 520 с.	5-98003-245-2, http://www.iprbookshop.ru/90354.html
6.1.2. Дополнительная литература				
№	Авторы, составители	Заглавие	Издательство, год	Количество/название ЭБС

№	Авторы, составители	Заглавие	Издательство, год	Количество/название ЭБС
Л2.1	Смирнов А. А.	Разработка прикладного программного обеспечения : учебное пособие	Москва: Евразийский открытый институт, Московский государственный университет экономики, статистики и информатики, 2003, 101 с.	2227-8397, http://www.iprbookshop.ru/10808.html
Л2.2	Троелсен Э.	Язык программирования C# 5.0 и платформа .NET 4.5	М.: Вильямс, 2015, 1312с.	978-5-8459-1957-1, 1
Л2.3	Смирнов А. А.	Прикладное программное обеспечение : учебное пособие	Москва: Евразийский открытый институт, 2011, 384 с.	978-5-374-00340-6, http://www.iprbookshop.ru/11079.html
Л2.4	Назаркин О. А.	Разработка графического пользовательского интерфейса в соответствии с паттерном Model-View-Viewmodel на платформе Windows Presentation Foundation. Основные средства WPF : учебное пособие по дисциплине «проектирование человеко-машинного интерфейса»	Липецк: Липецкий государственный технический университет, ЭБС АСВ, 2014, 61 с.	978-5-88247-679-2, http://www.iprbookshop.ru/55141.html
Л2.5	Воронцов Ю. А., Ерохин А. Г.	Разработка Windows-приложений в среде программирования Visual Studio.Net : учебно-методическое пособие по дисциплине информатика и программирование	Москва: Московский технический университет связи и информатики, 2016, 20 с.	2227-8397, http://www.iprbookshop.ru/61536.html
Л2.6	Котляров В. П.	Основы тестирования программного обеспечения	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016, 334 с.	5-94774-406-4, http://www.iprbookshop.ru/62820.html
Л2.7	Кара-Ушанов В. Ю.	SQL - язык реляционных баз данных : учебное пособие	Екатеринбург: Уральский федеральный университет, ЭБС АСВ, 2016, 156 с.	978-5-7996-1622-9, http://www.iprbookshop.ru/68419.html
Л2.8	Алексеев В. А.	Паттерны проектирования программных систем : методические указания к проведению лабораторных работ по курсу «архитектура программных систем»	Липецк: Липецкий государственный технический университет, ЭБС АСВ, 2016, 33 с.	2227-8397, http://www.iprbookshop.ru/74412.html

№	Авторы, составители	Заглавие	Издательство, год	Количество/название ЭБС
Л2.9	Туральчук К. А.	Параллельное программирование с помощью языка С#	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019, 189 с.	978-5-4486-0506-2, http://www.iprbookshop.ru/9714.html
Л2.10	Гамма Э., Хелм Р., Джонсон Р., Влссидес Дж.	Приемы объектно-ориентированного проектирования. Паттерны проектирования	СПб.: Питер, 2010, 366с.	978-5-469-01136-1, 1

6.1.3. Методические разработки

№	Авторы, составители	Заглавие	Издательство, год	Количество/название ЭБС
Л3.1	Корепанов С.Е., Стротов В.В.	Основы программирования на языке С#: метод. указ. к лаб. работам. Часть 1 : Методические указания	Рязань: РИЦ РГРТУ, 2020,	https://elib.rsreu.ru/ebs/download/2576

6.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	Официальный интернет портал РГРТУ [электронный ресурс] http://www.rsreu.ru			
Э2	Образовательный портал РГРТУ [электронный ресурс]. - Режим доступа: по паролю.- https://edu.rsreu.ru			
Э3	Электронная библиотека РГРТУ [электронный ресурс]. - Режим доступа: доступ из корпоративной сети РГРТУ - по паролю. - http://elib.rsreu.ru/			
Э4	Электронно-библиотечная система IRPbooks [электронный ресурс]. - Режим доступа: доступ из корпоративной сети РГРТУ - свободный, доступ из сети интернет- по паролю. - https://www.iprbookshop.ru/			
Э5	Электронно-библиотечная система «Лань» [электронный ресурс]. - Режим доступа: доступ из корпоративной сети РГРТУ - свободный, доступ из сети интернет- по паролю. - https://e.lanbook.com			
Э6	Информационный ресурс о системе контроля версий Git [электронный ресурс]. - Режим доступа: свободный. - https://git-scm.com/book/ru/v2			

6.3 Перечень программного обеспечения и информационных справочных систем

6.3.1 Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства

Наименование	Описание
Операционная система Windows	Коммерческая лицензия
Kaspersky Endpoint Security	Коммерческая лицензия
LibreOffice	Свободное ПО
Visual studio community	Свободное ПО
Adobe Acrobat Reader DC	Свободное ПО
Mozilla Thunderbird	Свободное ПО

6.3.2 Перечень информационных справочных систем

6.3.2.1	Информационно-правовой портал ГАРАНТ.РУ http://www.garant.ru
6.3.2.2	Система КонсультантПлюс http://www.consultant.ru
6.3.2.3	Справочная правовая система «КонсультантПлюс» (договор об информационной поддержке №1342/455-100 от 28.10.2011 г.)

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

1	445 учебно-административный корпус. Учебная аудитория для проведения учебных занятий лекционного и семинарского типа, практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации Специальная мебель (54 посадочных места), компьютер с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду РГРТУ, мультимедиа проектор, экран, доска, колонки звуковые.
---	--

2	430 учебно-административный корпус. Учебная аудитория для проведения учебных занятий семинарского типа, лабораторных работ, практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации 24 учебных компьютеров с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду РГРТУ, сервер данных
3	449 учебно-административный корпус. Учебная аудитория для проведения учебных занятий семинарского типа, лабораторных работ, практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации 15 компьютеров с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду РГРТУ, проектор, экран, доска, магнитный усилитель, фазовращатель, асинхронные приводы, осциллограф, электронный микроскоп, учебный роботизированный стенд, учебный комплект роботизированного оборудования Mindstorms, видеокамера
4	447 учебно-административный корпус. Помещение для самостоятельной работы обучающихся 10 компьютеров с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду РГРТУ, учебный роботизированный стенд, видеокамеры, сервер данных

8. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Методические материалы приведены в приложении к рабочей программе дисциплины (см. документ "Методическое обеспечение дисциплины "Прикладное программирование")

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. В.Ф. УТКИНА**

Кафедра «Автоматики и информационных технологий в управлении»

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДИСЦИПЛИНЫ

ПРИКЛАДНОЕ ПРОГРАММИРОВАНИЕ

Специальность 12.05.01

«Электронные и оптико-электронные приборы
и системы специального назначения»

ОПОП

«Оптико-электронные информационно-измерительные приборы и системы»

Квалификация выпускника – инженер

Формы обучения – очная

Рязань 2020 г.

Оценочные материалы – это совокупность учебно-методических материалов (контрольных заданий, описаний форм и процедур), предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной профессиональной образовательной программы.

Цель – оценить соответствие знаний, умений и уровня приобретенных компетенций обучающихся целям и требованиям основной профессиональной образовательной программы в ходе проведения текущего контроля и промежуточной аттестации.

Основная задача – обеспечить оценку уровня сформированности общекультурных, общепрофессиональных и профессиональных компетенций, приобретаемых обучающимся в соответствии с этими требованиями.

Контроль знаний проводится в форме текущего контроля и промежуточной аттестации.

Текущий контроль успеваемости проводится с целью определения степени усвоения учебного материала, своевременного выявления и устранения недостатков в подготовке обучающихся и принятия необходимых мер по совершенствованию методики преподавания учебной дисциплины (модуля), организации работы обучающихся в ходе учебных занятий и оказания им индивидуальной помощи.

К контролю текущей успеваемости относятся проверка знаний, умений и навыков, приобретенных обучающимися в ходе выполнения индивидуальных заданий на практических занятиях и лабораторных работах. При оценивании результатов освоения практических занятий и лабораторных работ применяется шкала оценки «зачтено – не зачтено». Количество лабораторных и практических работ и их тематика определена рабочей программой дисциплины, утвержденной заведующим кафедрой.

Результат выполнения каждого индивидуального задания должен соответствовать всем критериям оценки в соответствии с компетенциями, установленными для заданного раздела дисциплины.

Промежуточный контроль по дисциплине осуществляется проведением зачёта (7 семестр) и экзамена (8 семестр). При оценивании результатов освоения дисциплины на зачете применяется шкала оценки «зачтено – не зачтено», а на экзамене – традиционная балльная система (отлично, хорошо, удовлетворительно, неудовлетворительно).

Форма проведения зачета и экзамена – устный ответ по утвержденным экзаменационным билетам, сформулированным с учетом содержания учебной дисциплины. После выполнения письменной работы обучающегося производится ее оценка преподавателем и, при необходимости, проводится теоретическая беседа с обучаемым для уточнения экзаменационной оценки.

Паспорт оценочных материалов по дисциплине

Модуль 1

№ п/п	Контролируемые разделы (темы) дисциплины (результаты по разделам)	Код контролируемой компетенции (или её части)	Вид, метод, форма оценочного мероприятия
1	2	3	4
1	<i>Раздел 1</i> Программирование на языке C#	ОПК-3.2-3 ОПК-3.2-У ОПК-3.2-В	Зачет, лабораторная работа

Модуль 2

№ п/п	Контролируемые разделы (темы) дисциплины (результаты по разделам)	Код контролируемой компетенции (или её части)	Вид, метод, форма оценочного мероприятия
1	2	3	4
1	<i>Раздел 3</i> Динамическая типизация. Атрибуты. Потоки.	ОПК-3.2-3 ОПК-3.2-У ОПК-3.2-В	Экзамен
2	<i>Раздел 4</i> Современные подходы к разработке ПО. Системы контроля версий.	ОПК-3.2-3 ОПК-3.2-У ОПК-3.2-В	Экзамен
3	<i>Раздел 5</i> Паттерны проектирования ПО	ОПК-3.2-3 ОПК-3.2-У ОПК-3.2-В	Экзамен
4	<i>Раздел 6</i> Создание оконных приложений Windows Forms	ОПК-3.2-3 ОПК-3.2-У ОПК-3.2-В	Экзамен

Критерии оценивания компетенций (результатов)

- 1). Уровень усвоения материала, предусмотренного программой.
 - 2). Умение анализировать материал, устанавливать причинно-следственные связи.
 - 3). Ответы на вопросы: полнота, аргументированность, убежденность, умение
 - 4). Качество ответа (его общая композиция, логичность, убежденность, общая эрудиция)
 - 5). Использование дополнительной литературы при подготовке ответов.
- Уровень освоения сформированности знаний, умений и навыков по дисциплине оценивается в форме бальной оценки:

«Отлично» заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание учебно-программного материала, умение

свободно выполнять задания, предусмотренные программой, изучивший основную, и знакомый с дополнительной литературой, рекомендованной программой. Как правило, оценка «отлично» выставляется студентам, усвоившим взаимосвязь основных понятий дисциплины в их значении для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала.

«Хорошо» заслуживает студент, обнаруживший полное знание учебно-программного материала, успешно выполняющий предусмотренные в программе задания, усвоивший основную литературу, рекомендованную в программе. Как правило, оценка «хорошо» выставляется студентам, показавшим систематический характер знаний по дисциплине и способным к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности.

«Удовлетворительно» заслуживает студент, обнаруживший знания основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по специальности, справляющийся с выполнением заданий, предусмотренных программой, знакомый с основной литературой, рекомендованной программой. Как правило, оценка «удовлетворительно» выставляется студентам, допустившим погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя.

«Неудовлетворительно» выставляется студенту, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение или приступить к профессиональной деятельности по окончании вуза без дополнительных занятий по соответствующей дисциплине.

Оценка «зачтено» выставляется студенту, который прочно усвоил предусмотренный программный материал; правильно, аргументировано ответил на все вопросы, с приведением примеров; показал глубокие систематизированные знания, владеет приемами рассуждения и сопоставляет материал из разных источников: теорию связывает с практикой, другими темами данного курса, других изучаемых предметов.

Обязательным условием выставленной оценки является правильная речь в быстром или умеренном темпе. Дополнительным условием получения оценки «зачтено» могут стать хорошие успехи при выполнении самостоятельной работы, а также уверенная защита лабораторных работ.

Оценка «не зачтено» выставляется студенту, который не справился с 50% вопросов и заданий билета, в ответах на другие вопросы допустил существенные ошибки. Не может ответить на дополнительные вопросы, предложенные преподавателем. Целостного представления об изучаемой дисциплине у студента нет. Оценивается качество устной речи и изложение письменного материала, как и при выставлении положительной оценки.

Типовые контрольные задания или иные материалы

Вопросы к зачету по дисциплине

1. Обзор платформы Microsoft .NET. Общеязыковая исполняющая среда Common Language Runtime (CLR). Библиотека базовых классов Framework Class Library (FCL). Стандартная система типов Common Type System (CTS). Общеязыковая спецификация Common Language Specification (CLS). Языки программирования .NET. Компиляция и исполнение .NET приложений.
2. Обзор Visual Studio. Основные окна Visual Studio, используемые при программировании. Интегрированная система документации. Типы приложений. Общая структура программы. Оператор using. Точка входа программы Main(). Пример простой программы. Документирование кода.
3. Иерархия классов для типов данных. Понятие структурных и ссылочных типов. Стек. Куча. Встроенные типы данных. Псевдонимы. Оператор sizeof.
4. Переменная. Правила именования. Область видимости и время существования переменных. Константы. Класс Console. Форматирование строк для вывода на консоль. Класс Math.
5. Преобразование типов (Явное и неявное преобразование, повышающее и понижающее преобразование). Checked и Unchecked. Класс Convert, методы ToString и Parse. Неявно типизированные локальные переменные.
6. Операции. Приоритеты операций. Арифметические операции. Операции отношения. Логические операции. Поразрядные операции. Операции присваивания.
7. Операторы. Условные конструкции (условный оператор if, оператор многозначного выбора switch, тернарный оператор). Операторы перехода (goto, break, continue). Циклические конструкции (while, do-while, for, foreach))
8. Понятие процедурного программирования. Методы. Модификаторы параметров аргументов методов (ref, out, params). Перегрузка методов. Параметры аргументов при вызове методов (позиционные, именованные, опциональные). Рекурсивные методы.
9. Понятие объектно-ориентированного программирования (ООП). Понятие класса. Поля класса. Методы класса. Объекты и экземпляры. Создание экземпляра класса по сильной и слабой ссылке. Передача экземпляра класса в качестве аргумента метода.
10. Парадигмы ООП. Инкапсуляция. Спецификаторы доступа. Свойства класса. Автоматически реализуемые свойства.
11. Инициализация экземпляров класса. Понятие конструктора класса. Использование ключевого слова this для обращения к полям класса. Вызов конструкторов по цепочке. Частичные классы. Частичные методы. Поля только для чтения.
12. Понятие массива. Типы массивов. Одномерный массив. Многомерные массивы. Прямоугольные и зубчатые массивы. Класс Array.
13. Понятие индексатора. Примеры с целочисленными и строковыми типами принимаемых аргументов индексатора.

14. Понятие наследования. Создание многоуровневой иерархии классов. Модификатор доступа `protected`. Вызов конструктора базового класса из производного класса.
15. Понятие полиморфизма. Классический полиморфизм. Приведение к базовому и производному типу. Замещение метода базового класса. Виртуальные методы. Операторы `as` и `is`. Модификатор `sealed`.
16. Понятие абстракции. Понятие абстрактного класса. Абстрактные члены.
17. Понятие интерфейса. Реализация интерфейсов. Множественное наследование интерфейсов. Техника объединения реализации одноименных абстрактных методов. Наследование от класса и интерфейса, у которых совпадают сигнатуры членов.
18. Класс `Char`. Массив символов `char[]`. Класс `String`. Способы создания строк. Строка – неизменяемый тип. Понятие интернирования строк. Класс `StringBuilder`.
19. Регулярные выражения. Класс `Regex`. Метасимволы. Квантификаторы. Примеры шаблонов.
20. Статические классы и их члены. Понятие методов расширения.
21. Понятие структуры. Работа со структурными типами. Отличие структуры от класса. Вложенные структуры и классы. Упаковка (`boxing`) и распаковка (`unboxing`). Структура `DateTime`.
22. Понятие Перечисления. Типы перечислений. Примеры использования перечислений.
23. Понятие делегатов. Создание делегата. Создание экземпляра делегата. Вызов методов, сообщенных с делегатом. Неизменяемость делегатов. Комбинирование делегатов. Анонимные методы. Лямбда-операторы. Лямбда-выражения. Техника предположения делегата.
24. Понятие события. Создание события и его свойства. Обработчики событий. Подписка и отписка от событий. Паттерн проектирования `Model-View-Presenter`.
25. Понятие обобщений. Идентификатор `<T>`. Обобщение и упаковка-распаковка. Ограничения обобщений. Ограничение `new()`. Ограничение `struct`. Ограничение `class`. Ограничение `<base class name>`. Неприкрытое ограничение типа.
26. Понятие коллекции. Интерфейс `IEnumerable`. Интерфейс `IEnumerator`. Ключевое слово `yield`. Циклический оператор `foreach`. Интерфейс `IEnumerable<T>`. Другие интерфейсы для работы с обобщенными и необобщенными коллекциями.
27. Коллекции `ArrayList`, `List<T>`, `Dictionary<TKey, TValue>` и другие. Свойства и методы коллекций.
28. Роль класса `Object`. Методы класса `Object`. Интерфейс `ICloneable`.
29. Перегрузка операторов. Правила перегрузок. Операторы явного и неявного преобразования типа `explicit` и `implicit`.
30. Понятие обработки исключений. Ключевые слова `try`, `catch`, `finally`, `throw`. Обработка нескольких конструкций `catch`. Примеры использования.

Вопросы к экзамену по дисциплине

1. Анонимные типы. Динамическая типизация. Тип `dynamic`.
2. Понятие LINQ. Запросы LINQ. Ключевые слова: `from`, `where`, `select`, `group`, `orderby`, `join`, `let`.
3. Пространства имен. Директива `using`. Подключение новых сборок к проекту. Модификаторы доступа при работе со сборками. Директивы компилятора: `#if`, `#endif`, `#region`, `#else`, `#elif`, `#define`, `#undef`.
4. Понятие атрибута. Предопределенные и пользовательские атрибуты. Класс `Attribute`. `AttributeUsage`, `Conditional`, `Obsolete`.
5. Понятие многозадачности на основе процессов и на основе потоков. Пространство имен `System.Threading`. Классы `Thread`, `Monitor`, делегаты `ThreadStart`, `ParameterizedThreadStart`.
6. Понятие приоритетного и фонового потока. Понятие синхронизации потоков. Синхронный и асинхронный потоки. Ключевое слово `lock`.
7. Понятие технологии разработки ПО. Основные этапы разработки ПО.
8. Модели жизненного цикла ПО. Водопадные модели. V-образная модель. Итеративная модель. Спиральная модель.
9. Современные подходы к разработке ПО.
10. Понятие системы контроля версий. Система `Git`. Работа с удаленным репозиторием. Запись изменений, чтение информации, операции отмены.
11. Понятие системы контроля версий. Система `Git`. Управление ветками: слияние, перебазирование, удаление. Решение конфликтов.
12. Понятие паттерна проектирования. Порождающие паттерны. Паттерн `Abstract Factory`.
13. Понятие паттерна проектирования. Порождающие паттерны. Паттерн `Abstract Builder`.
14. Понятие паттерна проектирования. Порождающие паттерны. Паттерн `Factory Method`.
15. Понятие паттерна проектирования. Порождающие паттерны. Паттерн `Prototype`.
16. Понятие паттерна проектирования. Порождающие паттерны. Паттерн `Abstract Singleton`.
17. Понятие паттерна проектирования. Структурные паттерны. Паттерн `Adapter`.
18. Понятие паттерна проектирования. Структурные паттерны. Паттерн `Brige`.
19. Понятие паттерна проектирования. Структурные паттерны. Паттерн `Composite`.
20. Понятие паттерна проектирования. Структурные паттерны. Паттерн `Decorator`.
21. Понятие паттерна проектирования. Структурные паттерны. Паттерн `Facade`.
22. Понятие паттерна проектирования. Структурные паттерны. Паттерн `Flyweight`.
23. Понятие паттерна проектирования. Поведенческие паттерны. Паттерн `Chain of Responsibility`.
24. Понятие паттерна проектирования. Поведенческие паттерны. Паттерн `Command`.

25. Понятие паттерна проектирования. Поведенческие паттерны. Паттерн Interpreter.
26. Понятие паттерна проектирования. Поведенческие паттерны. Паттерн Iterator.
27. Понятие паттерна проектирования. Поведенческие паттерны. Паттерн Mediator.
28. Понятие паттерна проектирования. Поведенческие паттерны. Паттерн Memento. Паттерн State.
29. Понятие паттерна проектирования. Поведенческие паттерны. Паттерн Observer.
30. Понятие паттерна проектирования. Поведенческие паттерны. Паттерн Strategy.
31. Понятие паттерна проектирования. Поведенческие паттерны. Паттерн Template Method.
32. Понятие паттерна проектирования. Поведенческие паттерны. Паттерн Visitor.
33. Обзор пространства имен System.Windows.Forms. Проект Windows Forms. Класс Application. Класс Control. Класс Form.
34. Проект Windows Forms. Работа с меню (MenuStrip), со строкой состояния (StatusStrip) и таймером (Timer). Создание дочерних окон.
35. Обзор пространства имен GDI+. Обзор пространства имен System.Drawing. Тип Point(F). Тип Rectangle(F). Класс Region.
36. Рисование в окне. Пространство имен System.Windows.Drawing. Класс Graphics. Основные классы для рисования – Pen, Brush.
37. Рисование в окне. Пространство имен System.Windows.Drawing. Класс Graphics. Основные классы для рисования – Color, Font.
38. Элементы управления Windows Forms. Базовые элементы управления. Элементы Label, TextBox, MaskedTextBox, Button.
39. Элементы управления Windows Forms. Базовые элементы управления. Элементы CheckBox, RadioButton, GroupBox, CheckedListBox, ListBox, ComboBox.
40. Элементы управления Windows Forms. Создание изображений. Контроль анимации.
41. Классификация видов тестирования ПО.
42. Понятие Unit-тестирования.

Типовые задания для самостоятельной работы

Модуль 1 (семестр 7)

1. История платформы .NET.
2. Встроенные типы данных на языке C#.
3. Виды условных операторов в C#.
4. Обеспечение инкапсуляции при написании программ на языке C#.
5. Проявление полиморфизма в C#.
6. Сравнение интерфейсов и абстрактных классов в C#.
7. Статические члены в C#.
8. Классы для работы со строками в C#.
9. Сравнение структур и классов в C#.
10. Классификация коллекций в C#.

Модуль 2 (семестр 8)

1. Понятие динамической типизации.
2. Работа с потоками в C#.
3. Модели жизненного цикла ПО.
4. Обзор одного из паттернов проектирования.
5. Обзор метода управления проектами – SCRUM.
6. Создание модулей в Паскале.
7. Принцип работы с оконными приложениями Windows Forms.
8. Обзор видов тестирования ПО.
9. Обзор пространства имен System.Drawing.
10. Правила формирования запроса LINQ.

Лабораторный практикум

Модуль 1

№ п/п	№ раздела дисциплины	Наименование лабораторной работы	Трудоемкость, час
1	1	Основы языка программирования c#	4
2	1	Операции и операторы	4
3	1	Классы	4
4	1	Массивы и индексы. Наследование и полиморфизм	4

Практические занятия

Модуль 1

№ п/п	№ раздела дисциплины	Наименование практического занятия	Трудоемкость, час
1	1	Абстрактные классы. Интерфейсы.	4
2	1	Символы. Строки. Регулярные выражения.	4
3	1	Делегаты. События.	4
4	1	Коллекции	4

Модуль 2

№ п/п	№ раздела дисциплины	Наименование практического занятия	Трудоемкость, час
1	3	Анонимные и динамические типы. LINQ.	4
2	3	Многозадачность на основе потоков.	4
3	4	Системы контроля версий.	4
4	4	Основы тестирования ПО.	4
1	5	Порождающие паттерны.	4
2	5	Структурные паттерны.	4
3	5	Поведенческие паттерны.	4
4	6	Разработка оконных приложений Windows Forms	4

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. В.Ф. УТКИНА**

Кафедра «Автоматики и информационных технологий в управлении»

МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ
ПРИКЛАДНОЕ ПРОГРАММИРОВАНИЕ

Специальность 12.05.01
«Электронные и оптико-электронные приборы
и системы специального назначения»

ОПОП
«Оптико-электронные информационно-измерительные приборы и системы»

Квалификация выпускника – инженер

Формы обучения – очная

Рязань 2020

Методические рекомендации студентам по освоению дисциплины

Перед началом изучения дисциплины студенту необходимо ознакомиться с содержанием рабочей программы дисциплины, с целями и задачами дисциплины, ее связями с другими дисциплинами образовательной программы, методическими разработками по данной дисциплине, имеющимися на образовательном портале РГРТУ и сайте кафедры.

Методические рекомендации студентам по работе над конспектом лекции

Основу теоретического обучения студентов составляют лекции. Они дают систематизированные знания студентам о наиболее сложных и актуальных проблемах изучаемой дисциплины. На лекциях особое внимание уделяется не только усвоению студентами изучаемых проблем, но и стимулированию их активной познавательной деятельности, творческого мышления, развитию научного мировоззрения, профессионально-значимых свойств и качеств.

Перед каждой лекцией студенту необходимо просматривать рабочую программу дисциплины, что позволит сэкономить время на записывание темы лекции, ее основных вопросов, рекомендуемой литературы.

Перед очередной лекцией необходимо просмотреть по конспекту материал предыдущей лекции. При затруднениях в восприятии материала следует обратиться к основным литературным источникам. Если разобраться в материале опять не удалось, то обратитесь к лектору (по графику его консультаций) или к преподавателю на практических занятиях. Не оставляйте «белых пятен» в освоении материала.

Во время лекции студенты должны не только внимательно воспринимать действия преподавателя, но и самостоятельно мыслить, добиваться понимания изучаемого предмета. Студенты должны аккуратно вести конспект. В случае недопонимания какой-либо части предмета следует задать вопрос в установленном порядке преподавателю. В процессе работы на лекции необходимо так же выполнять в конспектах модели изучаемого предмета (рисунки, схемы, чертежи и т. д.), которые использует преподаватель.

Слушая лекцию, нужно из всего получаемого материала выбирать и записывать самое главное. Следует знать, что главные положения лекции преподаватель обычно выделяет интонацией или повторяет несколько раз. Именно поэтому предварительная подготовка к лекции позволит студенту

уловить тот момент, когда следует перейти к конспектированию, а когда можно просто внимательно слушать лекцию. В связи с этим нелишне перед началом сессии еще раз бегло просмотреть учебники или прежние конспекты по изучаемым предметам. Это станет первичным знакомством с тем материалом, который прозвучит на лекции, а также создаст необходимый психологический настрой.

Чтобы правильно и быстро конспектировать лекцию важно учитывать, что способы подачи лекционного материала могут быть разными. Преподаватель может диктовать материал, рассказывать его, не давая ничего под запись, либо проводить занятие в форме диалога со студентами. Чаще всего можно наблюдать соединение двух или трех вышеназванных способов.

Эффективность конспектирования зависит от умения владеть правильной методикой записи лекции. Конечно, способы конспектирования у каждого человека индивидуальны. Однако существуют некоторые наиболее употребляемые и целесообразные приемы записи лекционного материала.

Запись лекции можно вести в виде тезисов – коротких, простых предложений, фиксирующих только основное содержание материала. Количество и краткость тезисов может определяться как преподавателем, так и студентом. Естественно, что такая запись лекции требует впоследствии обращения к дополнительной литературе. На отдельные лекции можно приносить соответствующий иллюстративный материал на бумажных или электронных носителях, представленный лектором на портале или присланный на «электронный почтовый ящик группы» (таблицы, графики, схемы). Данный материал будет охарактеризован, прокомментирован, дополнен непосредственно на лекции.

Кроме тезисов важно записывать примеры, доказательства, даты и цифры, имена. Значительно облегчают понимание лекции те схемы и графики, которые вычерчивает на доске преподаватель. По мере возможности студенты должны переносить их в тетрадь рядом с тем текстом, к которому эти схемы и графики относятся.

Хорошо если конспект лекции дополняется собственными мыслями, суждениями, вопросами, возникающими в ходе прослушивания содержания лекции. Те вопросы, которые возникают у студента при конспектировании лекции, не всегда целесообразно задавать сразу при их возникновении, чтобы не нарушить ход рассуждений преподавателя. Студент может попытаться ответить на них сам в процессе подготовки к семинарам либо обсудить их с преподавателем на консультации.

Важно и то, как будет расположен материал в лекции. Если запись тезисов ведется по всей строке, то целесообразно отделять их время от времени красной строкой или пропуском строки. Примеры же и

дополнительные сведения можно смещать вправо или влево под тезисом, а также на поля. В тетради нужно выделять темы лекций, записывать рекомендуемую для самостоятельной подготовки литературу, внести фамилию, имя и отчество преподавателя. Наличие полей в тетради позволяет не только получить «ровный» текст, но и дает возможность при необходимости вставить важные дополнения и изменения в конспект лекции.

При составлении конспектов необходимо использовать основные навыки стенографии. Так в процессе совершенствования навыков конспектирования лекций важно выработать индивидуальную систему записи материала, научиться рационально сокращать слова и отдельные словосочетания.

Практика показывает, что не всегда студенту удастся успевать записывать слова лектора даже при использовании приемов сокращения слов. В этом случае допустимо обратиться к лектору с просьбой повторить сказанное. При обращении важно четко сформулировать просьбу, указать какой отрывок необходимо воспроизвести еще раз. Однако не всегда удобно прерывать ход лекции. В этом случае можно оставить пропуск, и после лекции устранить его при помощи конспекта соседа. Важно сделать это в короткий срок, пока свежа память о воспринятой на лекции информации.

Работу над конспектом следует начинать с его доработки, желательно в тот же день, пока материал еще легко воспроизводим в памяти (через 10 часов после лекции в памяти остается не более 30-40 % материала). С целью доработки необходимо прочитать записи, восстановить текст в памяти, а также исправить описки, расшифровать не принятые ранее сокращения, заполнить пропущенные места, понять текст, проникнуть в его смысл. Далее следует прочитать материал по рекомендуемой литературе, разрешая в ходе чтения возникшие ранее затруднения, вопросы, а также дополняя и исправляя свои записи. Записи должны быть наглядными, для чего следует применять различные способы выделений. В ходе доработки конспекта углубляются, расширяются и закрепляются знания, а также дополняется, исправляется и совершенствуется конспект.

Подготовленный конспект и рекомендуемая литература используются при подготовке к семинарским и практическим занятиям. Подготовка сводится к внимательному прочтению учебного материала, к выводу с карандашом в руках всех утверждений и формул, к решению примеров, задач, к ответам на вопросы. Примеры, задачи, вопросы по теме являются средством самоконтроля.

Непременным условием глубокого усвоения учебного материала является знание основ, на которых строится изложение материала. Обычно преподаватель напоминает, какой ранее изученный материал и в какой

степени требуется подготовить к очередному занятию. Обращение к ранее изученному материалу не только помогает восстановить в памяти известные положения, выводы, но и приводит разрозненные знания в систему, углубляет и расширяет их. Каждый возврат к старому материалу позволяет найти в нем что-то новое, переосмыслить его с иных позиций, определить для него наиболее подходящее место в уже имеющейся системе знаний. Неоднократное обращение к пройденному материалу является наиболее рациональной формой приобретения и закрепления знаний.

Методические рекомендации студентам по работе с литературой

В рабочей программе дисциплины для каждого раздела и темы дисциплины указывается основная и дополнительная литература, позволяющая более глубоко изучить данный вопрос. Обычно список всей рекомендуемой литературы преподаватель озвучивает на первой лекции или дает ссылки на ее местонахождение (на образовательном портале РГРТУ, на сайте кафедры и т. д.).

При работе с рекомендуемой литературой целесообразно придерживаться такой последовательности. Сначала лучше прочитать заданный текст в быстром темпе. Цель такого чтения заключается в том, чтобы создать общее представление об изучаемом материале, понять общий смысл прочитанного. Затем прочитать вторично, более медленно, чтобы в ходе чтения понять и запомнить смысл каждой фразы, каждого положения и вопроса в целом.

Чтение приносит пользу и становится продуктивным, когда сопровождается записями. Это может быть составление плана прочитанного текста, тезисы или выписки, конспектирование и др. Выбор вида записи зависит от характера изучаемого материала и целей работы с ним. Если содержание материала несложное, легко усваиваемое, можно ограничиться составлением плана. Если материал содержит новую и трудно усваиваемую информацию, целесообразно его законспектировать.

План – это схема прочитанного материала, перечень вопросов, отражающих структуру и последовательность материала.

Конспект – это систематизированное, логичное изложение материала источника. Различаются четыре типа конспектов:

- план-конспект – это развернутый детализированный план, в котором по наиболее сложным вопросам даются подробные пояснения,
- текстуальный конспект – это воспроизведение наиболее важных положений и фактов источника,

- свободный конспект – это четко и кратко изложенные основные положения в результате глубокого изучения материала, могут присутствовать выписки, цитаты, тезисы; часть материала может быть представлена планом,
- тематический конспект – составляется на основе изучения ряда источников и дает ответ по изучаемому вопросу.

В процессе изучения материала источника и составления конспекта нужно обязательно применять различные выделения, подзаголовки, создавая блочную структуру конспекта. Это делает конспект легко воспринимаемым и удобным для работы.

Методические рекомендации студентам по подготовке к практическим занятиям

По наиболее сложным вопросам учебной дисциплины проводятся практические занятия. Их главной задачей является углубление и закрепление теоретических знаний у студентов, формирование и развитие у них умений и навыков применения знаний для успешного решения прикладных задач. Практическое занятие проводится в соответствии с планом. В плане указываются тема, время, место, цели и задачи практического занятия, список основной и дополнительной литературы, рекомендованной к практическому занятию. Подготовка студентов к занятию включает:

- заблаговременное ознакомление с планом занятия;
- изучение рекомендованной литературы и конспекта лекций;
- подготовку полных и глубоких ответов по каждому вопросу, выносимому для обсуждения;

При проведении практического занятия уделяется особое внимание заданиям, предполагающим не только воспроизведение студентами знаний, но и направленных на развитие у них практических умений и навыков, а также творческого мышления, научного мировоззрения, профессиональных представлений и способностей.

В ходе практического занятия студент должен опираться на свои конспекты, сделанные на лекции, собственные выписки из учебников по данной теме, примеры решения подобных задач, полученные во время самостоятельной работы.

Самое главное на практическом занятии – уметь решить поставленную на занятии задачу и дать преподавателю и своим коллегам-студентам соответствующие пояснения. Поэтому необходимо обратить внимание на полезные советы:

1. Если студент чувствует, что не владеет навыком устного изложения, необходимо составить подробный план материала, который он будет излагать. Но только план, а не подробный ответ, чтобы избежать зачитывания.

2. Студенту необходимо стараться отвечать, придерживаясь пунктов плана.

3. При устном ответе не волноваться, так как вокруг друзья, а они очень благожелательны к присутствующим.

4. Следует говорить внятно при ответе, не употреблять слова-паразиты.

5. Полезно изложить свои мысли по тому или иному вопросу дома, в общезнании.

При необходимости следует обращаться за консультацией к преподавателю. Идя на консультацию, необходимо хорошо продумать вопросы, которые требуют разъяснения. Работа на всех практических занятиях в течение семестра позволяет подготовиться без трудностей и успешно сдать экзамен или зачет.

Методические рекомендации студентам по подготовке к лабораторным работам

Лабораторная работа — это форма организации учебного процесса, когда обучающиеся по заданию и под руководством преподавателя самостоятельно проводят опыты, измерения, экспериментальные исследования, вычислительные расчеты, разработку программного обеспечения на основе специально разработанных заданий.

Для проведения лабораторных работ используется специальное лабораторное оборудование, измерительная аппаратура, вычислительная техника, которые размещаются в специально оборудованных учебных лабораториях. Перед началом цикла лабораторных работ преподаватель или другое ответственное лицо проводит с обучающимися инструктаж о правилах техники безопасности в данной лаборатории, после чего студенты расписываются в специальном журнале техники безопасности.

По каждой лабораторной работе разрабатываются методические указания по их проведению. Они используются обучающимися при выполнении лабораторной работы.

Применяются разные формы организации обучающихся на лабораторных работах: фронтальная, групповая и индивидуальная. При фронтальной форме организации занятий все обучающиеся выполняют одновременно одну и ту же работу. При групповой форме организации занятий одна и та же работа выполняется группами по 2-5 человек. При

индивидуальной форме организации занятий каждый обучающийся выполняет индивидуальное задание. Выбор метода зависит от учебно-методической базы и задач курса.

До начала лабораторной работы студент должен ознакомиться с теоретическими вопросами, которые будут изучаться или исследоваться в этой работе. Также необходимо познакомиться с принципами работы лабораторного оборудования, используемого в лабораторной работе. Перед началом лабораторной работы преподаватель может провести проверку знаний обучающихся - их теоретической готовности к выполнению задания. По итогам этой проверки студент допускается или не допускается к данной работе. О такой исходной проверке преподаватель информирует студентов заранее. Также возможна ситуация, когда допуском к очередной лабораторной работе является своевременная сдача предыдущей лабораторной работы (или подготовка отчета по ней).

Во время лабораторной работы обучающиеся выполняют запланированное лабораторное задание. Все полученные результаты (числовые данные, графики, тексты программ) необходимо зафиксировать в черновике отчета или сохранить в электронном виде на сменном носителе.

Завершается лабораторная работа оформлением индивидуального отчета и его защитой перед преподавателем.

Приступая к работе в лаборатории студенту следует знать, что в отличие от других видов занятий, пропущенную или некачественно выполненную лабораторную работу нельзя отработать в любое время. Для этого существуют специальные дополнительные дни ликвидации учебных задолженностей. Поэтому пропускать лабораторную работу без уважительной причины крайне нежелательно.

При подготовке к лабораторным работам по Прикладному программированию в 7-м семестре следует использовать методические указания [1].

Методические рекомендации студентам по подготовке к зачету или экзамену

При подготовке к зачету или экзамену студент должен повторно изучить конспекты лекций и рекомендованную литературу, просмотреть решения основных задач, решенных самостоятельно и на семинарах, а также составить письменные ответы на все вопросы, вынесенные на зачет или экзамен.

Необходимо помнить, что практически все зачеты и экзамены в вузе сконцентрированы в течение короткого временного периода в конце семестра

в соответствии с расписанием. Промежутки между очередными зачетами и экзаменами обычно составляют всего несколько дней. Поэтому подготовку к ним нужно начинать заблаговременно в течение семестра. До наступления сессии уточните у преподавателя порядок проведения промежуточной аттестации по его предмету и формулировки критериев для количественной оценивания уровня подготовки студентов. Очень часто для итоговой положительной оценки по предмету необходимо вовремя и с нужным качеством выполнить или защитить контрольные работы, типовые расчеты, лабораторные работы, т. к. всё это может являться обязательной частью учебного процесса по данной дисциплине.

Рекомендуется разработать план подготовки к каждому зачету и экзамену, в котором указать, какие вопросы или билеты нужно выучить, какие задачи решить за указанный в плане временной отрезок.

Также бывает полезно вначале изучить более сложные вопросы, а затем переходить к изучению более простых вопросов. При этом желательно в начале каждого следующего дня подготовки бегло освежить в памяти выученный ранее материал.

В период сдачи зачетов и экзаменов организм студента работает в крайне напряженном режиме и для успешной сдачи сессии нужно не забывать о простых, но обязательных правилах:

- по возможности обеспечить достаточную изоляцию: не отвлекаться на разговоры с друзьями, просмотры телепередач, общение в социальных сетях;
- уделять достаточное время сну;
- отказаться от успокоительных. Здоровое волнение – это нормально. Лучше снимать волнение небольшими прогулками, самовнушением;
- внушать себе, что сессия – это не проблема. Это нормальный рабочий процесс. Не накручивайте себя, не создавайте трагедий в своей голове;
- помогите своему организму – обеспечьте ему полноценное питание, давайте ему периоды отдыха с переменной вида деятельности;
- следуйте плану подготовки.

Методические рекомендации студентам по проведению самостоятельной работы

Самостоятельная работа студента над учебным материалом является неотъемлемой частью учебного процесса в вузе.

В учебном процессе образовательного учреждения выделяются два вида самостоятельной работы:

1) аудиторная – выполняется на учебных занятиях, под непосредственным руководством преподавателя и по его заданию), студентам могут быть предложены следующие виды заданий:

- выполнение самостоятельных работ;
- выполнение контрольных и лабораторных работ;
- составление схем, диаграмм, заполнение таблиц;
- решение задач;
- работу со справочной, нормативной документацией и научной литературой;
- защиту выполненных работ;
- тестирование и т. д.

2) *внеаудиторная* – выполняется по заданию преподавателя, но без его непосредственного участия, включает следующие виды деятельности.

- подготовку к аудиторным занятиям (теоретическим, практическим занятиям, лабораторным работам);
- изучение учебного материала, вынесенного на самостоятельную проработку: работа над определенными темами, разделами, вынесенными на самостоятельное изучение в соответствии с рабочими программами учебной дисциплины или профессионального модуля;
- выполнение домашних заданий разнообразного характера;
- выполнение индивидуальных заданий, направленных на развитие у студентов самостоятельности и инициативы;
- подготовку к учебной и производственной практикам и выполнение заданий, предусмотренных программами практик;
- подготовку к контрольной работе, зачету, экзамену;
- написание курсовой работы, реферата и других письменных работ на заданные темы;
- подготовку к ГИА, в том числе выполнение ВКР;
- другие виды внеаудиторной самостоятельной работы, специальные для конкретной учебной дисциплины или профессионального модуля.

Внеаудиторные самостоятельные работы представляют собой логическое продолжение аудиторных занятий, проводятся по заданию преподавателя, который инструктирует студентов и устанавливает сроки выполнения задания.

При планировании заданий для внеаудиторной самостоятельной работы используются следующие типы самостоятельной работы:

- воспроизводящая (репродуктивная), предполагающая алгоритмическую деятельность по образцу в аналогичной ситуации. Включает следующую основную деятельность: самостоятельное прочтение, просмотр, конспектирование учебной литературы, прослушивание

записанных лекций, заучивание, пересказ, запоминание, Internet–ресурсы, повторение учебного материала и др.

– реконструктивная, связанная с использованием накопленных знаний и известного способа действия в частично измененной ситуации, предполагает подготовку сообщений, докладов, выступлений на семинарских и практических занятиях, подбор литературы по дисциплинарным проблемам, написание рефератов, контрольных, курсовых работ и др.

– эвристическая (частично-поисковая) и творческая, направленная на развитие способностей студентов к исследовательской деятельности. Включает следующие виды деятельности: написание рефератов, научных статей, участие в научно–исследовательской работе, подготовка дипломной работы (проекта), выполнение специальных заданий и др., участие в студенческой научной конференции.

Одной из важных форм самостоятельной работы студента является работа с литературой ко всем видам занятий: лабораторным, семинарским, практическим, при подготовке к зачетам, экзаменам, тестированию, участию в научных конференциях.

Один из методов работы с литературой – повторение: прочитанный текст можно заучить наизусть. Простое повторение воздействует на память механически и поверхностно. Полученные таким путем сведения легко забываются.

Более эффективный метод – метод кодирования: прочитанный текст нужно подвергнуть большей, чем простое заучивание, обработке. Чтобы основательно обработать информацию и закодировать ее для хранения, важно провести целый ряд мыслительных операций: прокомментировать новые данные; оценить их значение; поставить вопросы; сопоставить полученные сведения с ранее известными. Для улучшения обработки информации очень важно устанавливать осмысленные связи, структурировать новые сведения.

Изучение научной учебной и иной литературы требует ведения рабочих записей. Форма записей может быть весьма разнообразной: простой или развернутый план, тезисы, цитаты, конспект.

План – структура письменной работы, определяющая последовательность изложения материала. Он является наиболее краткой и потому самой доступной и распространенной формой записей содержания исходного источника информации. По существу, это перечень основных вопросов, рассматриваемых в источнике. План может быть простым и развернутым. Их отличие состоит в степени детализации содержания и, соответственно, в объеме.

Преимущество плана состоит в том, что план позволяет наилучшим образом уяснить логику мысли автора, упрощает понимание главных моментов произведения. Кроме того, он позволяет быстро и глубоко проникнуть в сущность построения произведения и, следовательно, гораздо легче ориентироваться в его содержании и быстрее обычного вспомнить прочитанное. С помощью плана гораздо удобнее отыскивать в источнике нужные места, факты, цитаты и т. д.

Выписки представляют собой небольшие фрагменты текста (неполные и полные предложения, отдельные абзацы, а также дословные и близкие к дословным записи об излагаемых в нем фактах), содержащие в себе квинтэссенцию содержания прочитанного. Выписки представляют собой более сложную форму записи содержания исходного источника информации. По сути, выписки – не что иное, как цитаты, заимствованные из текста. Выписки позволяют в концентрированной форме и с максимальной точностью воспроизвести наиболее важные мысли автора. В отдельных случаях – когда это оправдано с точки зрения продолжения работы над текстом – вполне допустимо заменять цитирование изложением, близким дословному.

Тезисы – сжатое изложение содержания изученного материала в утвердительной (реже опровергающей) форме. Отличие тезисов от обычных выписок состоит в том, что тезисам присуща значительно более высокая степень концентрации материала. В тезисах отмечается преобладание выводов над общими рассуждениями. Записываются они близко к оригинальному тексту, т. е. без использования прямого цитирования.

Аннотация – краткое изложение основного содержания исходного источника информации, дающее о нем обобщенное представление. К написанию аннотаций прибегают в тех случаях, когда подлинная ценность и пригодность исходного источника информации исполнителю письменной работы окончательно неясна, но в то же время о нем необходимо оставить краткую запись с обобщающей характеристикой.

Резюме – краткая оценка изученного содержания исходного источника информации, полученная, прежде всего, на основе содержащихся в нем выводов. Резюме весьма сходно по своей сути с аннотацией. Однако, в отличие от последней, текст резюме концентрирует в себе данные не из основного содержания исходного источника информации, а из его заключительной части, прежде всего выводов. Но, как и в случае с аннотацией, резюме излагается своими словами – выдержки из оригинального текста в нем практически не встречаются.

Конспект представляет собой сложную запись содержания исходного текста, включающая в себя заимствования (цитаты) наиболее

примечательных мест в сочетании с планом источника, а также сжатый анализ записанного материала и выводы по нему.

При выполнении конспекта требуется внимательно прочитать текст, уточнить в справочной литературе непонятные слова и вынести справочные данные на поля конспекта. Нужно выделить главное, составить план. Затем следует кратко сформулировать основные положения текста, отметить аргументацию автора. Записи материала следует проводить, четко следуя пунктам плана и выражая мысль своими словами. Цитаты должны быть записаны грамотно, учитывать лаконичность, значимость мысли.

В тексте конспекта желательно приводить не только тезисные положения, но и их доказательства. При оформлении конспекта необходимо стремиться к емкости каждого предложения. Мысли автора книги следует излагать кратко, заботясь о стиле и выразительности написанного. Число дополнительных элементов конспекта должно быть логически обоснованным, записи должны распределяться в определенной последовательности, отвечающей логической структуре произведения. Для уточнения и дополнения необходимо оставлять поля. Необходимо указывать библиографическое описание конспектируемого источника.

Типовые задания для самостоятельной работы

Модуль 1 (семестр 7)

1. История платформы .NET.
2. Встроенные типы данных на языке C#.
3. Виды условных операторов в C#.
4. Обеспечение инкапсуляции при написании программ на языке C#.
5. Проявление полиморфизма в C#.
6. Сравнение интерфейсов и абстрактных классов в C#.
7. Статические члены в C#.
8. Классы для работы со строками в C#.
9. Сравнение структур и классов в C#.
10. Классификация коллекций в C#.

Модуль 2 (семестр 8)

1. Понятие динамической типизации.
2. Работа с потоками в C#.

3. Модели жизненного цикла ПО.
4. Обзор одного из паттернов проектирования.
5. Обзор метода управления проектами – SCRUM.
6. Создание модулей в Паскале.
7. Принцип работы с оконными приложениями Windows Forms.
8. Обзор видов тестирования ПО.
9. Обзор пространства имен System.Drawing.
10. Правила формирования запроса LINQ.

Библиографический список

1. Корепанов С.Е., Стротов В.В. Основы программирования на языке С# (часть 1): методические указания к лабораторным работам / Рязан. гос. радиотехн. ун-т; Рязань, 2020. 80 с. Электронный каталог РГРТУ, <https://elib.rsreu.ru/ebs/download/2576>