

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Райхерт Татьяна Николаевна  
Должность: Директор  
Дата подписания: 09.03.2023 13:44:56  
Уникальный программный ключ:  
c914df807d771447164c08ee17f8e2f93dde816b

Министерство образования и науки Российской Федерации  
Нижнетагильский государственный социально-педагогический институт (филиал)  
Федерального государственного автономного образовательного учреждения  
высшего образования  
«Российский государственный профессионально-педагогический университет»

Факультет естествознания, математики и информатики  
Кафедра информационных технологий

УТВЕРЖДАЮ  
Зам. директора по УМР

\_\_\_\_\_ Л. П. Филатова

«\_\_\_» \_\_\_\_\_ 2018 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ  
«ВЫСОКОУРОВНЕВЫЕ МЕТОДЫ ИНФОРМАТИКИ И  
ПРОГРАММИРОВАНИЯ»**

Уровень высшего образования	Бакалавриат
Направление подготовки	09.03.03 Прикладная информатика
Профиль	Прикладная информатика в экономике
Форма обучения	Очная, заочная

Рабочая программа дисциплины «Высокоуровневые методы информатики и программирования». Нижний Тагил: Нижнетагильский государственный социально-педагогический институт (филиал) ФГАОУ ВО «Российский государственный профессионально-педагогический университет», 2018. – 29 с.

Настоящая программа составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.03 Прикладная информатика.

Автор: кандидат педагогических наук, Н.В. Бужинская  
доцент кафедры информационных технологий

Рецензент: кандидат педагогических наук, доцент М.В.Мащенко

Одобрена на заседании кафедры информационных технологий 21 июня 2018 г., протокол № 12.

Заведующая кафедрой М. В. Мащенко

Председатель методической комиссии ФЕМИ В. А. Гордеева

Рекомендована к печати методической комиссией факультета естествознания, математики и информатики 13 сентября 2018 г., протокол № 1.

Декан ФЕМИ Т. В. Жуйкова

Зав отделом АВТ и МТО научной библиотеки О. В. Левинских

© Нижнетагильский государственный социально-педагогический институт (филиал) ФГАОУ ВО «Российский государственный профессионально-педагогический университет», 2018.  
© Бужинская Надежда Владимировна, 2018.

## СОДЕРЖАНИЕ

1. Цель и задачи освоения дисциплины .....	4
2. Место дисциплины в структуре образовательной программы .....	4
3. Результаты освоения дисциплины .....	4
4. Структура и содержание дисциплины.....	5
4.1. Объем дисциплины и виды контактной и самостоятельной работы.....	5
4.2. Содержание и тематическое планирование дисциплины.....	7
4.3. Содержание разделов (тем) дисциплины .....	13
5. Образовательные технологии.....	15
6. Учебно-методические материалы .....	16
6.1. Планирование самостоятельной работы (очная форма обучения).....	16
6.2. Задания и методические указания по организации самостоятельной работы.....	20
7. Учебно-методическое и информационное обеспечение .....	24
8. Материально-техническое обеспечение дисциплины .....	24
9. Текущий контроль качества усвоения знаний .....	25
10. Промежуточная аттестация .....	25

## 1. ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

**Цель дисциплины** – отработка технологии применения основ программирования для решения задач будущей профессиональной деятельности.

**Задачи дисциплины:**

- ознакомление студентов с существующими парадигмами программирования, и областями их применения;
- ознакомление с современными технологиями программирования: объектно-ориентированным программированием, модульным программированием, компонентно-ориентированным программированием;
- ознакомление с жизненным циклом программного продукта, начиная от этапа разработки и заканчивая его адаптацией соответственно требованиям заказчика;
- изучение основных методов решения задач по программированию;
- изучение принципов создания и описания прикладных приложений на языках программирования.

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Данная дисциплина включена в Блок Б1 «Дисциплины (модули)» и является составной частью раздела Б1.В. «Вариативная часть», Б1.В.ОД «Обязательные дисциплины». Дисциплина реализуется в НТГСПИ на кафедре информационных технологий.

Данная дисциплина логически связана с дисциплинами профиля «Прикладная информатика в экономике» («Операционные системы», «Базы данных», «Информатика и программирование»), которые изучаются на первом-третьем курсах. Студентам необходимо знать основы данных дисциплин, чтобы при изучении курса «Высокоуровневые методы информатики и программирования» они могли выбрать наиболее эффективный способ решения задачи, а также использовать знания в области программирования для разработки пользовательского интерфейса.

Кроме того, дисциплина «Высокоуровневые методы информатики и программирования» логически и методически связана с курсами «Проектирование информационных систем» и «Программная инженерия», которые изучаются на четвертом-пятом курсах, так как разработка программных продуктов сопровождается созданием проекта будущего продукта и оформлением необходимой документации.

## 3. РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина направлена на формирование и развитие следующих компетенций:

- ОПК-3 – способностью использовать основные законы естественнонаучных дисциплин и современные информационно-коммуникационные технологии в современной деятельности;
- ПК-2 – способностью разрабатывать, внедрять и адаптировать прикладное программное обеспечение;
- ПК-8 – способностью программировать приложения и создавать программные прототипы решения прикладных задач;
- ПК-23 – способностью применять системный подход и математические методы в формализации решения прикладных задач.

В результате освоения дисциплины обучающийся должен:

**Знать:**

- 31 – жизненный цикл программного приложения и этапы его разработки;

–32 – современные подходы к программированию приложений, существующие парадигмы и технологии программирования, а также перспективы их развития;

–33 – основы системного подхода и математические методы, необходимые для формализации решения прикладных задач;

–34 – требования к разработке, внедрению и адаптации прикладного программного обеспечения;

**уметь:**

– У1 – разрабатывать, внедрять и адаптировать прикладное программное обеспечение;

– У2 – применять системный подход и математические методы в формализации решения прикладных задач;

– У3 – решать типовые задачи проектирования и разработки всех видов интерфейса с использованием современных технологий программирования;

– У4 – использовать информационно-коммуникационные технологии для решения задач в будущей профессиональной деятельности.

**владеть:**

–В1 – способностью применять системный подход в формализации решения прикладных задач и построении алгоритма решения;

–В2 – технологией программирования;

–В3 – способами разработки приложения, внедрением и адаптации.

## 4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

### 4.1. Объем дисциплины и виды контактной и самостоятельной работы

Трудоемкость дисциплины составляет 9 зач. ед. (324 часа) их распределение по видам работ представлено в таблице № 1. Для заочной формы обучения – 324 часа, 9 з.е., контрольная работа во втором семестре, курсовая работа в 3 семестре, зачет во втором семестре и экзамен в 3 семестре.

Таблица № 1

#### Распределение трудоемкости дисциплины по видам работ

Вид работы	Форма обучения	
	Очная	Заочная
	3 семестр	2 семестр
<b>Общая трудоемкость дисциплины по учебному плану</b>	<b>144/ 4 з.е.</b>	<b>108 / 3 з.е.</b>
<b>Контактная работа, в том числе:</b>	<b>44</b>	<b>12</b>
Лекции	16	6
Лабораторные работы	28	6
<b>Самостоятельная работа, в том числе:</b>	<b>100</b>	<b>96</b>
Изучение теоретического курса	30	30
Изучение практического курса	30	30
Самоподготовка к текущему контролю знаний	40	32

Подготовка к зачету	9	4
	4 семестр	3 семестр
<b>Общая трудоемкость дисциплины по учебному плану</b>	<b>72 / 2 з.е.</b>	<b>216 / 6 з.е.</b>
<b>Контактная работа, в том числе:</b>	<b>34</b>	<b>36</b>
Лекции	10	8
Лабораторные работы	24	12
<b>Самостоятельная работа, в том числе:</b>	<b>38</b>	<b>196</b>
Изучение теоретического курса	10	66
Изучение практического курса	10	66
Самоподготовка к текущему контролю знаний и зачету	18	51
Экзамен		13
	5 семестр	
<b>Общая трудоемкость дисциплины по учебному плану</b>	<b>108 / 3 з.е.</b>	
<b>Контактная работа, в том числе:</b>	<b>38</b>	
Лекции	12	
Лабораторные работы	26	
<b>Самостоятельная работа, в том числе:</b>	<b>70</b>	
Изучение теоретического курса	10	
Изучение практического курса	10	
Самоподготовка к текущему контролю знаний	14	
Подготовка к экзамену	36	

## 4.2. Содержание и тематическое планирование дисциплины

### 4.2.1. Тематический план дисциплины (очная форма обучения)

Наименование разделов и тем дисциплины (модуля)	Всего, часов	Вид контактной работы, час				Самостоятельная работа, час	Формы текущего контроля успеваемости
		Лекции	Практ. занятия	Лаб. работы	Из них в интерактивной форме		
<b>Раздел 1. Современные направления развития программирования</b>	<b>34</b>	<b>4</b>			<b>2</b>	<b>30</b>	
Тема 1. Парадигмы программирования.	18	2			2	16	Ответы на вопросы, Беседа
Тема 2. Технологии программирования (объектно-ориентированное, модульное, компонентно-ориентированное) и область их применения.	16	2				14	Ответы на вопросы. Беседа
<b>Раздел 2. Реализация основных алгоритмических конструкций на одном из современных высокоуровневых языков программирования.</b>	<b>40</b>	<b>6</b>		<b>14</b>	<b>6</b>	<b>20</b>	
Тема 1. Алфавит и синтаксис языка C#:	8			2		6	Оценивание правильности выполнения заданий лабораторной работы
Тема 2. Управляющие конструкции.	10	2		4	2	4	Оценивание правильности выполнения заданий лабораторной работы
Тема 3. Циклические операторы	12	2		4	2	6	Оценивание правильности выполнения заданий лабораторной работы
Тема 4. Строки.	10	2		4	2	4	Оценивание правильности выполнения заданий лабораторной работы
<b>Раздел 3. Составные типы данных.</b>	<b>61</b>	<b>6</b>		<b>14</b>		<b>41</b>	
Тема 1. Массивы	18	2		6		10	Оценивание правильности выполнения заданий лабораторной

Наименование разделов и тем дисциплины (модуля)	Всего, часов	Вид контактной работы, час				Самостоятельная работа, час	Формы текущего контроля успеваемости
		Лекции	Практ. занятия	Лаб. работы	Из них в интерактивной форме		
							работы
Тема 2. Структуры.	26	2		4		20	Оценивание правильности выполнения заданий лабораторной работы
Тема 3. Файлы	17	2		4		11	Оценивание правильности выполнения заданий лабораторной работы
<b>Зачет с оценкой</b>	<b>9</b>					<b>9</b>	Ответ на зачете
<b>Итого (3 семестр)</b>	<b>144</b>	<b>16</b>		<b>28</b>	<b>8</b>	<b>100</b>	
<b>Раздел 4. Разработка приложений Windows Forms</b>	<b>42</b>	<b>6</b>		<b>14</b>	<b>2</b>	<b>22</b>	
Тема 1. Интерфейс интегрированной среды разработки Visual Studio.	14	2		6		6	Оценивание правильности выполнения заданий лабораторной работы
Тема 2. Обработка событий	12	2		4	2	6	Оценивание правильности выполнения заданий лабораторной работы
Тема 3. Использование объектно-ориентированного языка программирования для работы с базами данных	16	2		4		10	Оценивание правильности выполнения заданий лабораторной работы
<b>Раздел 5. Объектно-ориентированный подход в программировании</b>	<b>30</b>	<b>4</b>		<b>10</b>	<b>2</b>	<b>16</b>	
Тема 1. Реализация объектно-ориентированного подхода на одном из современных высокоуровневых языков программирования.	30	4		10	2	16	Оценивание правильности выполнения заданий лабораторной работы
<b>Итого (4 семестр)</b>	<b>72</b>	<b>10</b>		<b>24</b>	<b>4</b>	<b>38</b>	
<b>Раздел 6. Жизненный цикл программных продуктов</b>	<b>12</b>	<b>4</b>		<b>2</b>	<b>2</b>	<b>6</b>	
Тема 1. Создание и тестирование прикладных приложений на одном из современных высокоуровневых	6	2		2	2	2	Оценивание правильности выполнения



Наименование разделов и тем дисциплины (модуля)	Всего, часов	Вид контактной работы, час				Самостоятельная работа, час	Формы текущего контроля успеваемости
		Лекции	Практ. занятия	Лаб. работы	Из них в интерактивной форме		
языков программирования.							заданий лабораторной работы
Тема 2. Модели разработки программных продуктов.	6	2			2	4	Оценивание правильности выполнения заданий лабораторной работы
<b>Раздел 7. Применения высокоуровневых языков программирования для решения различных задач</b>	<b>37</b>	<b>6</b>		<b>20</b>	<b>2</b>	<b>11</b>	
Тема 1. Искусственный интеллект и язык Prolog	11	2		6	2	3	Оценивание правильности выполнения заданий лабораторной работы
Тема 2. Визуальные среды программирования	12	2		6		4	Оценивание правильности выполнения заданий лабораторной работы
Тема 3. Программирование на одном из языков высокого уровня.	14	2		8		4	Оценивание правильности выполнения заданий лабораторной работы
<b>Раздел 8. Разработка пользовательского интерфейса</b>	<b>23</b>	<b>2</b>		<b>4</b>		<b>17</b>	
Тема 1. Виды интерфейса	10	2		2		6	Оценивание правильности выполнения заданий лабораторной работы
Тема 2. Использование объектно-ориентированного языка для разработки программных продуктов для автоматизации деятельности работников различных сфер	13			2		11	Оценивание правильности выполнения заданий лабораторной работы
Подготовка к экзамену	<b>36</b>					<b>36</b>	
<b>Итого (5 семестр):</b>	<b>108</b>	<b>12</b>		<b>26</b>	<b>6</b>	<b>70</b>	
Итого:	<b>324</b>	<b>38</b>		<b>78</b>	<b>20</b>	<b>208</b>	

#### 4.2.2. Тематический план дисциплины (заочная форма обучения)

Наименование разделов и тем дисциплины (модуля)	Всего, часов	Вид контактной работы, час				Самостоятельная работа, час	Формы текущего контроля успеваемости
		Лекции	Практ. занятия	Лаб. работы	Из них в интерактивной форме		
<b>Раздел 1. Современные направления программирования</b>	<b>24</b>	<b>4</b>			<b>2</b>	<b>20</b>	
Тема 1. Парадигмы программирования.	12	2			2	10	Ответы на вопросы, Беседа
Тема 2. Технологии программирования (объектно-ориентированное, модульное, компонентно-ориентированное) и область их применения.	12	2				10	Ответы на вопросы. Беседа
<b>Раздел 2. Реализация основных алгоритмических конструкций на одном из современных высокоуровневых языков программирования.</b>	<b>38</b>	<b>2</b>		<b>4</b>	<b>6</b>	<b>32</b>	
Тема 1. Алфавит и синтаксис языка C#:	12	2				10	Оценивание правильности выполнения заданий лабораторной работы
Тема 2. Управляющие конструкции.	12			2	2	10	Оценивание правильности выполнения заданий лабораторной работы
Тема 3. Циклические операторы	12			2	2	10	Оценивание правильности выполнения заданий лабораторной работы
Тема 4. Строки.	2				2	2	Оценивание правильности выполнения заданий лабораторной работы
<b>Раздел 3. Составные типы данных.</b>	<b>42</b>			<b>2</b>		<b>40</b>	
Тема 1. Массивы	12			2		10	Оценивание правильности выполнения заданий лабораторной работы

Тема 2. Структуры.	10					10	Оценивание правильности выполнения заданий лабораторной работы
Тема 3. Файлы	20					20	Оценивание правильности выполнения заданий лабораторной работы
<b>Зачет с оценкой</b>	4					<b>4</b>	Ответ на зачете
<b>Итого (2 семестр)</b>	<b>108</b>	<b>6</b>		<b>6</b>	<b>8</b>	<b>96</b>	
<b>Раздел 4. Разработка приложений Windows Forms</b>	<b>114</b>	<b>6</b>		<b>8</b>		<b>100</b>	
Тема 1. Интерфейс интегрированной среды разработки Visual Studio.	32	2				30	Оценивание правильности выполнения заданий лабораторной работы
Тема 2. Обработка событий	34	2		2		30	Оценивание правильности выполнения заданий лабораторной работы
Тема 3. Использование объектно-ориентированного языка программирования для работы с базами данных	48	2		6		40	Оценивание правильности выполнения заданий лабораторной работы
<b>Раздел 5. Разработка пользовательского интерфейса</b>	<b>89</b>	<b>2</b>		<b>4</b>	<b>2</b>	<b>83</b>	
Тема 1. Виды интерфейса	32	2				30	Оценивание правильности выполнения заданий лабораторной работы
Тема 2. Использование объектно-ориентированного языка для разработки программных продуктов для автоматизации деятельности работников различных сфер	57			4	2	53	Оценивание правильности выполнения заданий лабораторной работы
Экзамен	13					13	
<b>Итого (3 семестр):</b>	<b>216</b>	<b>8</b>		<b>12</b>	<b>2</b>	<b>196</b>	
<b>Итого:</b>	<b>324</b>	<b>14</b>		<b>18</b>	<b>10</b>	<b>292</b>	

### 4.2.3. Лабораторные работы (очная форма обучения)

№ раздела	Наименование лабораторных работ	Кол-во ауд. часов
<b>3 семестр</b>		
1	Алфавит и синтаксис языка	2
2,3	Управляющие конструкции.	4
4,5	Циклические операторы	4
6,7	Строки.	4
8	Одномерные и двумерные массивы	2
9	Сортировка массивов	2
10	Решение задач	2
11, 12	Структуры.	4
13	Файлы	2
14	Выполнение проекта	2
<b>4 семестр</b>		
15	Интерфейс интегрированной среды разработки Visual Studio.	2
16	Компиляция кода. Средства отладки. Настройка среды. Управление исходным кодом.	2
17	Разработка форм	2
18	Базовые элементы управления: метка, текстовое поле, кнопка, флажок, переключатель, списки.	2
19	Обработка событий	2
20	Решение задач	2
21	Назначение технологии ADO.	2
22	Использование технологии ADO для организации доступа к данным.	2
23	Создание пользовательских форм и запросов.	2
24	Разработка приложений с использованием технологии ADO.	2
25	Понятие ООП	2
26	Классы и методы	2
<b>5 семестр</b>		
27	Создание и тестирование прикладных приложений на одном из современных высокоуровневых языков программирования.	2
28	Модели разработки программных продуктов.	2
29	Введение в язык Prolog	2
30	Рекурсия	2
31	Обзор визуальных сред программирования и анализ их возможностей	2
32	Разработка приложения в визуальной среде	2
33	Возможности языков высокого уровня	2
34	Выбор темы проекта. Постановка задачи. Обоснование выбора средств	2
35	Разработка приложения	2
36	Тестирование приложения. Составление документации	2
37	Защита проекта	2
38	Виды интерфейса	2
39	Использование объектно-ориентированного языка для разработки программных продуктов для автоматизации деятельности работников различных сфер	2
		78

### 4.2.4. Лабораторные работы (заочная форма обучения)

№ раздела	Наименование лабораторных работ	Кол-во ауд. часов
<b>2 семестр</b>		
1	Управляющие конструкции.	2
2	Циклические операторы	2
3	Одномерные и двумерные массивы	2
<b>3 семестр</b>		
4	Разработка форм	2

5	Базовые элементы управления: метка, текстовое поле, кнопка, флажок, переключатель, списки.	2
6	Обработка событий	2
7	Решение задач	2
8	Назначение технологии ADO.	2
9	Использование технологии ADO для организации доступа к данным.	2
		18

### 4.3. Содержание разделов (тем) дисциплины

#### **Раздел 1. Современные направления развития программирования.**

##### **Парадигмы программирования.**

Понятие парадигмы и технологии программирования. Сравнительная характеристика декларативного и императивного программирования; логической, функциональной, структурной, объектно-ориентированной и параллельной парадигм программирования: достоинства и недостатки, область применения.

**Технологии программирования (объектно-ориентированное, модульное, компонентно-ориентированное) и область их применения.**

Понятие «технология программирования». Новые направления развития программирования.

**Раздел 2. Реализация основных алгоритмических конструкций на одном из современных высокоуровневых языков программирования.**

##### **Управляющие конструкции.**

Управляющие конструкции и их реализация в системе программирования (ветвление, оператор выбора, циклы, функции и процедуры).

##### **Циклические операторы.**

Различные виды циклов и их реализация в программе.

##### **Строки.**

Строковый тип данных. Методы и функции для работы со строками.

#### **Раздел 3. Составные типы данных.**

##### **Массивы.**

Понятие массива. Классификация массивов. Способы реализации массивов на языке C#. Сортировка массивов. Действия над элементами массива. Ступенчатые массивы. Строка как массив символов (слов).

##### **Структуры.**

Составной тип данных. Понятие структуры. Способы задания структуры. Операции над элементами структуры. Сортировка записей в структуре.

##### **Файлы.**

Виды файлов. Открытий и закрытий файла. Операции над элементами файла. Считывание данных из файла. Запись данных в файл.

### 4 семестр

#### **Раздел 4. Разработка приложений Windows Forms.**

Интерфейс интегрированной среды разработки Visual Studio.

Компиляция кода. Средства отладки. Настройка среды. Управление исходным кодом.

Разработка форм. Класс Form. Базовые элементы управления: метка, текстовое поле, кнопка, флажок, переключатель, списки. Процедуры обработки событий. Функция MsgBox и InputBox.

##### **Обработка событий.**

Обработка событий клавиатуры. Обработка событий мыши. Разработка справочной системы.

## **Использование объектно-ориентированного языка программирования для работы с базами данных.**

Понятие базы данных. Принципы работы поставщиков данных. Назначение технологии ADO. Использование технологии ADO для организации доступа к данным. Создание пользовательских форм и запросов. Разработка приложений с использованием технологии ADO.

### **Раздел 5. Объектно-ориентированный подход в программировании**

#### **Реализация объектно-ориентированного подхода на одном из современных высокоуровневых языков программирования.**

Сущность объектно-ориентированного подхода. Объектный тип данных. Переменные объектного типа.

Классы и объекты. Классификация объектов. Ограничение доступа к полям класса с помощью свойств.

Методы и события. События в среде Windows. Объявление класса. Создание экземпляра класса. Описание классов. Пространства имен. Конструкторы и деструкторы. Виды конструкторов. Свойства классов: наследование, инкапсуляция, полиморфизм. Перегрузка и переопределение методов на основании наследования.

Работа с динамическими массивами. Класс Array. Указатели. Работа с памятью компьютера. Реализация динамических структур данных с помощью массивов и указателей. Стеки. Очереди. Деревья.

Работа с XML-документами.

## **5 семестр**

### **Раздел 6. Жизненный цикл программных продуктов**

#### **Создание и тестирование прикладных приложений на одном из современных высокоуровневых языков программирования.**

Понятие тестирования. Виды тестирования. Подходы к тестированию. Сервисы для тестирования программных продуктов.

#### **Модели разработки программных продуктов.**

Жизненный цикл программного продукта. Способы описания жизненного цикла программного продукта. Оформление документации. Техническое задание. Руководство пользователя.

### **Раздел 7. Применения высокоуровневых языков программирования для решения различных задач**

#### **Искусственный интеллект и язык Prolog.**

Понятие искусственного интеллекта. Особенности языка Пролог. Правила сопоставления термов в системе Пролог. Общие принципы поиска ответов на вопросы системой Пролог. Объекты данных. Структурированные объекты. Структура программы. Арифметические выражения. Сравнение результатов арифметических выражений. Понятие списков.

#### **Визуальные среды программирования**

Парадигма визуального программирования. Визуальное и объектно-ориентированное программирование. Среда визуального программирования.

#### **Программирование на одном из языков высокого уровня**

Возможности языка . Знакомство со средой разработки. Условный оператор if. Циклы. Строки. Списки (массивы). Индексы и срезы. Кортежи. Словари. Множества. Функции. Исключения и их обработка. Байтовые строки. Файлы. With ... as – менеджеры контекста. Создание и подключение модулей

Объектно-ориентированное программирование. Основы. Инкапсуляция, наследование, полиморфизм. Перегрузка операторов. Декораторы. Требования к проекту. Проектная деятельность по программированию.

### **Раздел 8. Разработка пользовательского интерфейса**

#### **Виды интерфейса**

Понятие интерфейса. Особенности пользовательского интерфейса. Виды интерфейсов (командный, графический, звуковой) и способы их создания (MDI-технология, SDI-технология). Элементы интерфейса: окно, форма, меню, панель (лента). Использование библиотеки графических объектов для создания интерфейса. Создание MDI и SDI интерфейсов: разработка меню, панели инструментов, стандартных диалоговых окон, строки состояния, контекстного меню. Технология WPF и ее использование для создания пользовательских форм.

## 5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Данный курс предусматривает наличие теоретических лекционных занятий, на которых студенты знакомятся с общими современными направлениями в программировании, и практических – на которых осваивается конкретный язык программирования и соответствующие ему технологии.

Основными методами, используемыми при объяснении теоретического материала, будут: проблемное изложение; кейс-метод (с использованием элементов технологии опережающего развития); дискуссия (пресс-конференция; форум).

Основными методами, используемыми для практических занятий, будут: практикум, метод проектов, деловая игра и тренинг. Их соотношение может быть следующим:

№	Тема занятия	Вид занятия	Используемые методы
1	Парадигмы программирования	Лекция	Проблемное изложение
2	Технологии программирования (объектно-ориентированное, модульное, компонентно-ориентированное) и область их применения.	Лекция	Лекция-конференция
3	Строки. Циклические операторы Управляющие конструкции.	Лабораторные работы	Проект «Создание банковского калькулятора для пересчета валют, расчета по кредитам и вкладам»
4	Реализация объектно-ориентированного подхода на одном из современных высокоуровневых языков программирования	Лабораторная работа	Метод проблемного изложения Метод проектов «Создание индивидуального мини-проекта, представляющего собой решение практико-ориентированной задачи»
5	Создание и тестирование прикладных приложений на одном из современных высокоуровневых языков программирования.	Лекция	Мозговой штурм «В каком направлении будет развиваться рынок программного обеспечения и зависит ли это развитие от возможностей языков программирования?»
6	Обработка событий.	Лабораторная работа	Проект Разработка игры для детей младшего школьного возраста
7	Модели разработки программных продуктов.	Лекция	Шесть шляп мышления «О каких факторов зависит эффективность разработки программных

			продуктов». Взгляд на проблему с разных точек зрения: с точки зрения системного администратора, дизайнера, заказчика, программиста.
8	Искусственный интеллект и язык Prolog	Лекция	Мозговой штурм «Смогут ли машины заменить человека?»

## 6. УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

### 6.1. Планирование самостоятельной работы (очная форма обучения)

Темы занятий	Количество часов			Содержание самостоятельной работы	Формы контроля СРС
	Всего	Аудиторных	Самостоят. работы		
3 семестр					
<b>Раздел 1. Современные направления развития программирования</b>	<b>34</b>	<b>4</b>	<b>30</b>		
Тема 1. Парадигмы программирования.	18	2	16	Подготовка презентаций по теме «Новые направления развития программирования».	Проверка таблиц преподавателем, демонстрация презентаций с обсуждением на форуме.
Тема 2. Технологии программирования (объектно-ориентированное, модульное, компонентно-ориентированное) и область их применения.	16	2	14	Краткий конспект по вопросам самостоятельной работы	Проверка на занятии
<b>Раздел 2. Реализация основных алгоритмических конструкций на одном из современных высокоуровневых языков программирования.</b>	<b>40</b>	<b>20</b>	<b>20</b>		
Тема 1. Алфавит и синтаксис языка	8	2	6	Решение задач, представленных в лабораторной работе	Проверка на занятии
Тема 2. Управляющие конструкции.	10	6	4	Решение задач, представленных в лабораторной работе	Проверка на занятии
Тема 3. Циклические операторы	12	6	6	Решение задач для самостоятельной работы в практикуме «Создание банковского калькулятора для пересчета валют, расчета по кредитам	Проверка отчета и задач для сам. работы преподавателем



				и вкладам».	
Тема 4. Строки.	10	6	4	Создание презентации по теме «Основные методы и функции для работы со строками»	Проверка на занятии
<b>Раздел 3. Составные типы данных.</b>	<b>61</b>	<b>20</b>	<b>41</b>		
Тема 1. Массивы	18	8	10	Ступенчатые массивы. Их применение в решении задач	Обсуждение на занятии
Тема 2. Структуры.	26	6	20	Составные типы данных в C#. Структура как тип данных.	Обсуждение на занятии
Тема 3. Файлы	17	6	11	Типы файлов и их реализация.	Обсуждение на занятии
Зачет с оценкой	<b>9</b>		9		
<b>Раздел 4. Разработка приложений Windows Forms</b>	<b>42</b>	<b>20</b>	<b>22</b>		
Тема 1. Интерфейс интегрированной среды разработки Visual Studio.	14	8	6	Решение задач, представленных в лабораторной работе	Обсуждение на занятии
Тема 2. Обработка событий	12	6	6	Проект Разработка игры для детей младшего школьного возраста	Обсуждение на занятии
Тема 3. Использование объектно-ориентированного языка программирования для работы с базами данных	16	6	10	Решение задач, представленных в лабораторной работе	Проверка правильности выполнения задания
<b>Раздел 5. Объектно-ориентированный подход в программировании</b>	<b>30</b>	<b>14</b>	<b>16</b>		
Тема 1. Реализация объектно-ориентированного подхода на одном из современных высокоуровневых языков программирования.	30	14	16	Метод проектов «Создание индивидуального мини-проекта, представляющего собой решение практико-ориентированной задачи»	Проверка правильности выполнения задания
<b>Раздел 6. Жизненный цикл программных продуктов</b>	<b>12</b>	<b>6</b>	<b>6</b>		
Тема 1. Создание и тестирование прикладных приложений на одном из современных высокоуровневых языков программирования.	<b>6</b>	4	2	Решение задач, представленных в лабораторной работе Подготовка к мозговому штурму	Проверка правильности выполнения задания
Тема 2. Модели разработки программных продуктов.	<b>6</b>	2	4	Обзор и анализ моделей разработки программных	Проверка правильности выполнения

				продуктов Изучение вопроса «О каких факторов зависит эффективность разработки программных продуктов». Взгляд на проблему с разных точке зрения: с точки зрения системного администратора, дизайнера, заказчика, программиста.	задания
<b>Раздел 7. Применения высокоуровневых языков программирования для решения различных задач</b>	<b>37</b>	<b>26</b>	<b>11</b>		
Тема 1. Искусственный интеллект и язык Prolog	<b>11</b>	8	3	Решение задач, представленных в лабораторной работе, подготовка к мозговому штурму	Проверка правильности выполнения задания
Тема 2. Визуальные среды программирования	<b>12</b>	8	4	Обзор и анализ различных сред программирования	Проверка правильности выполнения задания
Тема 3. Программирование на одном из языков высокого уровня. Выполнение проекта	<b>14</b>	10	4	Решение задач, представленных в лабораторной работе	Проверка правильности выполнения задания
<b>Раздел 8. Разработка пользовательского интерфейса</b>	<b>23</b>	6	<b>17</b>		
Тема 1. Виды интерфейса.	<b>10</b>	4	6		Проверка правильности выполнения задания
Тема 2. Использование объектно-ориентированного языка для разработки программных продуктов для автоматизации деятельности работников различных сфер	<b>13</b>	2	11	Решение задач, представленных в лабораторной работе	Проверка правильности выполнения задания
Подготовка к экзамену	<b>36</b>		<b>36</b>		
<b>Итого</b>	<b>324</b>	<b>116</b>	<b>208</b>		

## 6.2. Планирование самостоятельной работы (заочная форма обучения)

Темы занятий	Количество часов			Содержание самостоятельной работы	Формы контроля СРС
	Всего	Аудиторных	Самостоят. работы		
3 семестр					
<b>Раздел 1. Современные направления развития программирования</b>	24	4	20		
Тема 1. Парадигмы программирования.	12	2	10	Подготовка презентаций по	Проверка таблиц преподавателем,

				теме «Новые направления развития программирования»	демонстрация презентаций с обсуждением на форуме.
Тема 2. Технологии программирования (объектно-ориентированное, модульное, компонентно-ориентированное) и область их применения.	12	2	10	Краткий конспект по вопросам самостоятельной работы	Проверка на занятии
<b>Раздел 2. Реализация основных алгоритмических конструкций на одном из современных высокоуровневых языков программирования.</b>	38	6	32		
Тема 1. Алфавит и синтаксис языка	12	2	10	Решение задач, представленных в лабораторной работе	Проверка на занятии
Тема 2. Управляющие конструкции.	12	2	10	Решение задач, представленных в лабораторной работе	Проверка на занятии
Тема 3. Циклические операторы	12	2	10	Решение задач для самостоятельной работы в практикуме «Создание банковского калькулятора для пересчета валют, расчета по кредитам и вкладам».	Проверка отчета и задач для сам. работы преподавателем
Тема 4. Строки.	2	0	2	Создание презентации по теме «Основные методы и функции для работы со строками»	Проверка на занятии
<b>Раздел 3. Составные типы данных.</b>	42	2	40		
Тема 1. Массивы	12	2	10	Ступенчатые массивы. Их применение в решении задач	Обсуждение на занятии
Тема 2. Структуры.	10	0	10	Составные типы данных в C#. Структура как тип данных.	Обсуждение на занятии
Тема 3. Файлы	4	0	4	Типы файлов и их реализация.	Обсуждение на занятии
<b>Раздел 4. Разработка приложений Windows Forms</b>	114	14	100		
Тема 1. Интерфейс интегрированной среды	32	2	30	Решение задач, представленных в	Обсуждение на занятии

разработки Visual Studio.				лабораторной работе Самостоятельное изучение вопросов 1-2	
Тема 2. Обработка событий	34	4	30	Проект Разработка игры для детей младшего школьного возраста	Обсуждение на занятии
Тема 3. Использование объектно-ориентированного языка программирования для работы с базами данных	48	8	40	Решение задач, представленных в лабораторной работе	Проверка правильности выполнения задания
<b>Раздел 5. Разработка пользовательского интерфейса</b>	<b>89</b>	<b>6</b>	<b>83</b>		
Тема 1. Виды интерфейса	32	2	30	Самостоятельное изучения вопросов 3-4	Проверка правильности выполнения задания
Тема 2. Использование объектно-ориентированного языка для разработки программных продуктов для автоматизации деятельности работников различных сфер	57	4	53	Решение задач, представленных в лабораторной работе	Проверка правильности выполнения задания
Экзамен	<b>13</b>	<b>0</b>	<b>13</b>		
Итого	<b>216</b>	<b>20</b>	<b>196</b>		
<b>Итого</b>	<b>324</b>	<b>32</b>	<b>292</b>		

## 6.2. Задания и методические указания по организации самостоятельной работы

### Семестр 3

#### Тема 1. Алфавит и синтаксис языка

Вопросы для отчета:

- 1) Какие существуют операторы?
- 2) Встроенные функции языка?
- 3) В чем различие операторов и операций в данном языке программирования?

#### Тема 2. Управляющие конструкции

Вопросы для отчета:

- 1) Оператор выбора.
- 2) Условный оператор и его типы.

#### Тема 3. Циклические операторы

Вопросы для отчета:

- 1) Какие существуют циклы?
- 2) Когда применяется цикл foreach?
- 3) В чем различие цикла foreach и for?

#### Тема 4. Строки

Вопросы для отчета:

- 1) Какие существуют методы для работы со строками?
- 2) Какие функции существуют функции для работы со строками?
- 3) В чем различие между методами и функциями для работы со строками?

#### Тема 5. Массивы

Вопросы для отчета:

- 1) Что такое одномерные массив?

- 2) Как задается одномерный массив?
- 3) Что такое двумерный массив?
- 4) Как задается двумерный массив?
- 5) Какие действия существуют над элементами массива?
- 6) Какие методы сортировки массивов существуют?
- 7) В чем заключается сущность каждого метода сортировки массивов?

#### **Тема 6. Структуры**

Вопросы для отчета:

- 1) Что такое структура?
- 2) Как задается структура?
- 3) Какие действия над элементами можно выполнять?
- 4) Как осуществляется сортировка записей структуры?

#### **Тема 7. Файлы**

Вопросы для отчета:

- 1) Как осуществляется запись в файл?
- 2) Какое пространство имен отвечает за работу с файлами?
- 3) Как осуществляется считывание записей из файлов?

### **4 семестр**

#### **Тема 1. Интерфейс интегрированной среды разработки Visual Studio.**

Вопросы для отчета:

- 1) Как осуществляется настройка среды?
- 2) Какие компоненты являются базовыми?
- 3) Как пишутся методы к элементам?

#### **Тема 2. Обработка событий.**

Вопросы для отчета:

- 1) Как осуществляется обработка событий клавиатуры?
- 2) Как осуществляется обработка событий мыши?

#### **Тема 3. Использование объектно-ориентированного языка программирования для работы с базами данных.**

Вопросы для отчета:

- 1) В чем заключается назначение технологии ADO?
- 2) Как используется технология ADO для организации доступа к данным?
- 3) Как осуществляется создание пользовательских форм и запросов?

#### **Тема 4. Реализация объектно-ориентированного подхода на одном из современных высокоуровневых языков программирования.**

Вопросы для отчета:

- 1) Что такое объектный тип данных? Переменные объектного типа.
- 2) Как задаются классы и объекты?
- 3) В чем заключается ограничение доступа к полям класса с помощью свойств?
- 4) Что такое методы и события?
- 5) Что такое конструкторы и деструкторы?
- 6) Какие виды конструкторов существуют?
- 7) Как реализуются свойства классов: наследование, инкапсуляция, полиморфизм?
- 8) В чем суть перегрузки и переопределение методов на основании наследования?
- 9) Что такое динамические массивы?
- 10) Работа с динамическими массивами и Класс Array?
- 11) Что такое указатели?
- 12) Как реализуются стеки, очереди и деревья?

### **5 семестр**

#### **Тема 1. Создание и тестирование прикладных приложений на одном из современных высокоуровневых языков программирования.**

Вопросы для отчета:

- 1) Что такое тестирование?
- 2) Какие виды тестирования существуют?.
- 3) Какие выделяются подходы к тестированию?
- 4) В чем особенность сервисов для тестирования программных продуктов?

## **Тема 2. Модели разработки программных продуктов.**

Вопросы для отчета:

- 1) Что такое жизненный цикл программного продукта?
- 2) Как описывается жизненный цикл программного продукта?
- 3) Что такое Техническое задание?
- 4) Как правильно оформляется руководство пользователя?

## **Тема 3. Искусственный интеллект и язык Prolog.**

1) В чем особенность программы на языке Prolog.

- 1) Что такое отношения?
- 2) Как задается база фактов?

## **Тема 4. Визуальные среды программирования**

- 1) Как разрабатывать приложения в среде Scratch?
- 1) Как разрабатывать приложения в среде Hiasm?
- 2) Каковы особенности визуального программирования в Google Blockly?

## **Тема 5. Программирование на одном из языков высокого уровня. Выполнение проекта**

- 1) Возможности языка.
- 2) Отличие данного языка от других языков высокого уровня.
- 3) Какие задачи можно решать с помощью данного языка?
- 4) Для разработки каких проектов применяется данный язык?

## **Тема 6. Виды интерфейса**

Вопросы для отчета:

- 1) Что такое интерфейс?
- 2) Какие выделяются виды интерфейсов?
- 3) Какие выделяются элементы интерфейса?
- 4) В чем суть технологии WPF и ее использования для создания пользовательских форм?

## **Тема 7. Использование объектно-ориентированного языка для разработки программных продуктов для автоматизации деятельности работников различных сфер**

- 1) Какие требования выделяются к программным продуктам для автоматизации деятельности работников различных сфер.
- 2) Автоматизированное рабочее место и его структура.
- 3) Место продуктов, разработанных с помощью программирования, в структуре автоматизированных рабочих мест.

В рамках самостоятельной работы студентов предусмотрено выполнение мини-проектов, которые рассчитаны на самостоятельную внеаудиторную работу студентов. Цель заданий – закрепление знаний, полученных на лабораторных занятиях. Задания можно выполнять дома или в компьютерном классе вуза. В процессе работы можно пользоваться методическими указаниями для выполнения лабораторных работ.

*Проект 1. Создание банковского калькулятора для пересчета валют, расчета по кредитам и вкладам*

Студент должен показать владение основными конструкциями программирования. В проекте должно быть предусмотрен расчет кредита методом простых и сложных процентов.

*Проект 2. Создание индивидуального мини-проекта, представляющего собой решение практико-ориентированной задачи*

Тему проекта студент формулирует сам, например «Разработка виртуальной экскурсии по г. Нижний Тагил», «Разработка базы данных по результативности игроков чемпионата мира по футболу». В процессе разработки студент должен показать владение технологиями программирования.

### **Примерная тематика индивидуальных проектов**

1. Разработка электронного календаря-еженедельника, позволяющего планировать текущие дела и напоминать о них согласно текущей дате.
2. Разработка электронного приложения «Склад», позволяющего заносить записи о товарах, их ценах и количестве, сортировать, находить нужный товар, иллюстрировать текущее положение в виде диаграммы и удалять купленные товары.
3. Разработка налогового калькулятора индивидуального предпринимателя.
4. Разработка валютного калькулятора с возможностью добавления новых валют и постоянного обновления курсов, а также хранением курсов в течение недели и иллюстрации их изменения в виде графика.
5. Разработка туристического калькулятора для расчета стоимости путевки в зависимости от входных параметров.
6. Разработка текстового редактора.
7. Разработка графического редактора.
8. Разработка мультимедиа плеера, поддерживающего различные форматы файлов.
9. Разработка математического редактора для работы с квадратными матрицами.
10. Разработка стратегической игры для двух игроков.

### *Проект 3. Разработка игры для детей младшего школьного возраста*

Проект подразумевает разработку игры для детей младшего школьного возраста, иллюстрирующей события мыши и клавиатуры. При разработке игры необходимо учесть особенности детей данного возраста.

### **Критерии оценки проектов**

- соответствие предложенной тематике;
- обоснованность структуры и выбранных форм;
- полнота;
- результативность;
- оригинальность;
- оптимальность используемых алгоритмов;
- оптимальность использования базовых элементов;
- удобство интерфейса пользователя.
- наглядность представления проекта;
- убедительность при обосновании и защите проекта;
- уровень владения материалом;
- умение вести дискуссию.

2 – компонент присутствует в полном объеме, 1- компонент присутствует, но есть ошибки, 0- компонент отсутствует. Оценка рассчитывается соответственно %, представленным в положении о НБРС НТСГИ (ф) РГППУ.

Общее количество баллов – 24.

«Отлично» – > 21 балла.

«Хорошо» – 17 – 20 баллов.

«Удовлетворительно» – 12 – 16 баллов.

«Не удовлетворительно» – < 12 баллов.

### **Перечень вопросов для самостоятельного изучения для заочной формы обучения**

1. Визуальные среды программирования.
2. Модели разработки программных продуктов.
3. Создание и тестирование прикладных приложений на одном из современных высокоуровневых языков программирования.
4. Виды интерфейса.

## 7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

### *Основная литература*

1. Бабушкина, И.А. Практикум по объектно-ориентированному программированию [Электронный ресурс] : учебное пособие. М.: Издательство «Лаборатория знаний», 2015. 369 с. Режим доступа: <https://e.lanbook.com/book/66121>
2. Бужинская, Н.В., Кудиенко, О.И. Высокоуровневые методы информатики и программирования. Язык С# [Текст]: Учебно-методическое пособие. Нижний Тагил: НТГСПИ (ф) РГППУ, 2016. 83 с. Режим доступа: <https://e.lanbook.com/book/66121>
3. Биллиг В.А. Основы программирования на С# [Электронный ресурс]. М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. 574 с. /Режим доступа: <http://www.iprbookshop.ru/73695.html>
4. Лубашева Т.В. Основы алгоритмизации и программирования [Электронный ресурс]: учебное пособие/ Минск: Республиканский институт профессионального образования (РИПО), 2016. 379 с. Режим доступа: <http://www.iprbookshop.ru/67689.html>
5. Медведев М.А. Программирование на СИ# [Электронный ресурс]: учебное пособие/ Екатеринбург: Уральский федеральный университет, ЭБС АСВ, 2015. 64 с. Режим доступа: <http://www.iprbookshop.ru/69667.html>
6. Снетков В.М. Практикум прикладного программирования на С# в среде VS.NET 2008 [Электронный ресурс]. М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. 1691 с.

### *Дополнительная литература*

7. Зайдельман, Я. Н. Эффективность алгоритмов: простые задачи и наглядные примеры. М.: Чистые пруды, 2006. 1 экз. Режим доступа: <https://e.lanbook.com/book/70753>
8. Зыков С.В. Введение в теорию программирования [Электронный ресурс]. М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. 188 с. Режим доступа: <http://www.iprbookshop.ru/73675.html>
9. Литвиненко, Н. А. Технология программирования на С++ Win32 API-приложения: учебное пособие для вузов. СПб.: БХВ-Петербург, 2010. Режим доступа: <http://www.iprbookshop.ru/69449.html>
10. Малиновская Е.А. Языки программирования. Часть 1 [Электронный ресурс]: лабораторный практикум/ Ставрополь: Северо-Кавказский федеральный университет, 2016. 103 с. Режим доступа: <http://www.iprbookshop.ru/62967.html>
11. Николаев Е.И. Объектно-ориентированное программирование [Электронный ресурс]: учебное пособие . Ставрополь: Северо-Кавказский федеральный университет, 2015. 225 с.
12. Плещев В. В. Высокоуровневые методы информатики и программирования с примерами и упражнениями: Учебник. – Екатеринбург: Изд-во Урал. гос. экон. ун-та, 2002.

## 8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

1. Лекционная аудитория –201А.
  2. Компьютеры (ноутбук).
  3. Мультимедиапроектор.
  4. Презентации к лекциям.
  5. Лабораторные работы.
- Учебная аудитория 201А  
35 посадочных мест для студентов,



11 рабочих мест для студентов,  
рабочее место преподавателя, маркерная доска, интерактивная доска,  
12 компьютеров, стационарный мультимедиакомплекс, учебный сервер  
Пакет офисных программ: Office Standard 2016 Russian OLP NL Academic Edition.  
Акт предоставления прав № IT021617 от 12.02.2016 г.  
Microsoft Visual Studio,  
Expressions и  
Embedded.  
Microsoft Visio,  
OneNote,  
Project.  
Серверы Microsoft SQL,  
BizTalk  
SharePoint  
Сублицензионный договор № Tr000142285 от 16.02.2017 г., продление 02.08.2018 г.  
№ счета 5024818829  
1С: Предприятие 8.3  
Лицензионный договор №Л-2015/42 от 05.11.2015 г.  
MathCad 14  
проприетарная  
код лицензии PKG-7517-FN от 31.12.2008 г.  
Бесплатное ПО:  
GIMP, Inkscape, Paint Net  
7-Zip  
Blender  
Hot Potatoes  
Nvu, Ebook Maestro FREE  
Ramus Educational  
Python, Dev C++  
Net Beans IDE

## **9. ТЕКУЩИЙ КОНТРОЛЬ КАЧЕСТВА УСВОЕНИЯ ЗНАНИЙ**

Текущий контроль сводится к систематической оценке аудиторной и самостоятельной работы студентов, в процессе которой выявляется полнота выполнения предлагаемых заданий, а также степень владения материалом.

## **10. ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ**

Промежуточная аттестация по данной дисциплине проводится в форме зачета с оценкой в 3 семестре, зачете – в 4 семестре и экзамена в 5 семестре.

На зачет выносятся один теоретический вопрос и практическое задание. Кроме того, оценка может быть поставлена по итогам работы студента в течении семестра.

### **Вопросы для зачета в третьем семестре**

1. Структуры.
2. Классификация массивов. Способы реализации массивов на языке программирования. Сортировка массивов.
3. Применение строк в системе программирования.
4. Алфавит и синтаксис языка:: структура программы, модуля, типы данных область видимости переменных. Константы. Основные операторы.
5. Управляющие конструкции и их реализация в системе программирования (ветвление, оператор выбора, циклы, функции и процедуры).

### **Примерные задания**

1. Составить программу, которая позволяет вычислять единый вмененный налог. Определите входные и выходные параметры для этой задачи.
2. Составьте программу «Конвертор валют».
3. Составьте программу, которая позволяет в виде класса задавать список сотрудников и выполнять сортировку сотрудников по размеру заработной платы.
4. Составьте в виде структуры список сотрудников и отсортируйте сотрудников по алфавиту. Выведите сотрудников с окладом больше 8000.
5. Введите строку и удалите из нее все гласные буквы.
6. Удалите в строке повторяющиеся пробелы.
7. Отсортируйте двумерный массив 4\*4 по возрастанию элементов в строках.

#### **Критерии оценки ответа на вопрос**

- полнота;
- фактическое знание материала;
- логичность;
- наличие практических примеров;
- умение поддержать диалог.

#### **Критерии выполнения практического задания**

- точность и рациональность алгоритмов;
- отсутствие ошибок;
- соответствие расчетных значений эталонным.

#### **Критерии оценивания ответов на зачете**

Оценка «**Отлично**» выставляется студентам, показавшим глубокое знание теоретической части курса, при развернутом ответе на теоретический вопрос, умение проиллюстрировать изложение материала практическими приемами, грамотных ответах на дополнительные вопросы преподавателя, а также выполнившим практическое задание.

Оценка «**Хорошо**» выставляется студентам, показавшим достаточное знание теоретического вопроса, умение проиллюстрировать часть изложенного материала примерами, отвечать на дополнительные вопросы преподавателя и выполнившим практическое задание без существенных ошибок. При ответе на теоретический вопрос и представлении практического задания допускаются незначительные ошибки.

Оценка «**Удовлетворительно**» выставляется студентам, показавшим знание основных положений теории при наличии существенных пробелов в деталях, испытывающим затруднения при практическом применении теории, допустившим существенные ошибки при ответе на вопрос билета и дополнительные вопросы преподавателя.

Оценка «**Неудовлетворительно**» выставляется, если студент показал существенные пробелы в знаниях основных положений теории (не ответил на теоретический вопрос), не умеет применять теоретические знания на практике, не показал знания основных понятий курса или не приступил к разработке практического задания.

Промежуточная аттестация в 4 семестре проводится в форме зачета.

На зачете студенту предлагается ответить на один теоретический вопрос и выполнить практическое задание. Оценку «Зачтено» студент может получить автоматически по результатам работы в течении семестра.

#### **Вопросы для зачета в четвертом семестре**

1. Разработка форм. Класс Form. Базовые элементы управления: метка, текстовое поле, кнопка, флажок, переключатель, списки.
2. Назначение технологии ADO. Использование технологии ADO для организации доступа к данным.
3. Сущность объектно-ориентированного подхода. Объектный тип данных. Переменные объектного типа.

4. Классы и объекты. Классификация объектов. Ограничение доступа к полям класса с помощью свойств.
5. Методы и события.
6. Конструкторы и деструкторы. Виды конструкторов.
7. Свойства классов: наследование, инкапсуляция, полиморфизм.
8. Перегрузка и переопределение методов на основании наследования.
9. Реализация динамических структур данных с помощью массивов и указателей.
10. Стеки. Очереди. Деревья.
11. Работа с XML-документами.

### Примеры практических заданий

1. Задайте два массив из 10 элементов каждый. Сформируйте из него новый массив четных элементов. Отсортируйте полученный массив и выведите его на экран.
2. С помощью делегатов вычислите значение  $\cos(x)$ ,  $\sin(x)$ ,  $\operatorname{tg}(x)$  и  $\operatorname{ctg}(x)$ . При этом переведите  $x$  из радиан в градусы.
3. С помощью делегатов оформите вопрос теста. В качестве значения строковой переменной можно указать `string s=""`.
4. Напишите программу, будут меняться значения  $a$  и  $b$ . Если  $a$  положительное число – то его значения меняется на отрицательное, для  $b$  – наоборот. Добавьте событие с обработчиком, которое выводит на экран надпись, что пользователь ввел 0. Указание – для изменения значения  $a$  или  $b$  воспользуйтесь методом `set`, а для возврата значения – методом `get`.
5. Напишите методы для проверки, правильно ли введены стороны (они должны быть  $>0$ ) и вывода значения этих сторон на экран.
6. Создайте класс для ввода длин сторон треугольника. Напишите конструктор для данного класса, создайте экземпляр класса и выведите информацию на экран.

### Критерии оценивания ответов на зачете

Оценка «**Зачтено**» выставляется студентам, показавшим знание теоретической части курса, при развернутом ответе на теоретический вопрос, умение проиллюстрировать изложение материала практическими приемами, ответах на дополнительные вопросы преподавателя, а также выполнившим практическое задание (допускаются незначительные ошибки).

Оценка «**Незачтено**» выставляется, если студент показал существенные пробелы в знаниях основных положений теории (не ответил на теоретический вопрос), не умеет применять теоретические знания на практике, не показал знания основных понятий курса или не приступил к разработке практического задания.

На экзамене в 5 семестре студенты должны ответить на один теоретический вопрос и представить проект.

Вопросы для экзамена в 5 семестре

### Список вопросов к экзамену

1. Перечислите основные панели инструментов, предназначенные для работы в среде Visual Studio. Расскажите об их элементах.
2. Из каких частей состоит проект, созданный в среде Visual Studio на языке C#. В какие файлы проект компилируется при сохранении. Виды проектов.
3. Указатели. Работа с памятью компьютера. Реализация динамических структур данных с помощью массивов и указателей. Указатели на функции и их использование.
4. Понятие компонента. Проектирование и использование компонентов.
5. Макросы и шаблоны.
6. Жизненный цикл программного продукта. Способы описания жизненного цикла программного продукта.

7. Особенности использования среды визуального программирования. Реализация концепции визуального программирования в Visual Studio.

8. Алгоритмы стандартной библиотеки. Программирование с использованием стандартной библиотеки.

9. Виды интерфейсов.

10. Методы для работы с коллекциями.

11. Конструкторы и деструкторы.

12. Абстрактные методы.

13. Жизненный цикл программного продукта.

14. Искусственный интеллект

15. Визуальная среда программирования.

### **Критерии оценки ответа на вопрос**

- полнота;
- фактическое знание материала;
- логичность;
- наличие практических примеров;
- умение поддерживать диалог.

### **Примерные практические задания**

1. Создайте класс для вывода и обработки информации о сотрудниках предприятия. Напишите методы для сортировки сотрудников по алфавиту, окладу. Выведите всех сотрудников пенсионного возраста.

2. Создайте класс «Геометрические фигуры». Найдите площади геометрических фигур.

3. Создайте детскую игру для демонстрации сложения и вычитания чисел.

4. Создайте базу данных по сотрудникам предприятия. Напишите методы для обработки информации об этих сотрудниках.

5. Проиллюстрируйте на примере абстрактные методы для класса.

### **Критерии оценивания ответов на экзамене**

Оценка **«Отлично»** выставляется студентам, показавшим глубокое знание теоретической части курса, при развернутом ответе на теоретический вопрос, умение проиллюстрировать изложение материала практическими приемами, грамотных ответах на дополнительные вопросы преподавателя, а также разработав проект соответственно представленным требованиям.

Оценка **«Хорошо»** выставляется студентам, показавшим достаточное знание теоретического вопроса, умение проиллюстрировать часть изложенного материала примерами, отвечать на дополнительные вопросы преподавателя и выполнившим практическое задание без существенных ошибок. При ответе на теоретический вопрос и демонстрации проекта допускаются незначительные ошибки.

Оценка **«Удовлетворительно»** выставляется студентам, показавшим знание основных положений теории при наличии существенных пробелов в деталях, испытывающим затруднения при практическом применении теории, допустившим существенные ошибки при ответе на вопрос билета и дополнительные вопросы преподавателя.

Оценка **«Неудовлетворительно»** выставляется, если студент показал существенные пробелы в знаниях основных положений теории (не ответил на теоретический вопрос), не умеет применять теоретические знания на практике, не показал знания основных понятий курса или не приступил к разработке индивидуального проекта.