

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Минцаев Магомед Шавалович

Должность: Ректор

Дата подписания: 28.12.2023 13:02:04

Уникальный программный ключ:

236bcc35c296f119d6aafdc22836b21db52dbc07971a86865a5825f9fa4304cc

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

имени академика М.Д. Миллионщикова

«УТВЕРЖДАЮ»

Первый проректор

И. Г. Гирабаев



«22» _____ 2023 г.

РАБОЧАЯ ПРОГРАММА

дисциплины

«Технологии программирования»

Направление подготовки

09.03.04 Программная инженерия

Направленность (профиль)

«Программная инженерия»

Квалификация

бакалавр

Год начала подготовки – 2024

Грозный – 2023

1. Цели и задачи дисциплины

Цель преподавания дисциплины «Технологии программирования» состоит в освоении студентами, получающими квалификацию бакалавра, подходов к составлению алгоритмов решения задач различной степени сложности, а также овладении навыками создания программ на примере среды разработки приложений Microsoft Visual Studio.

Задачами дисциплины являются: изучение основополагающих понятий и правил программирования; разработка алгоритмов обработки данных различной структуры, освоение правил создания и организации пользовательского интерфейса.

2. Место дисциплины в структуре образовательной программы

Учебная дисциплина «Технологии программирования» относится к обязательной части ФГОС ВО по направлению подготовки 09.03.04 Программная инженерия (квалификация «бакалавр»).

Для освоения дисциплины «Технологии программирования» студент должен обладать знаниями и умениями, приобретенными в результате освоения предшествующих дисциплин:

- информатика.

Дисциплина «Технологии программирования» является предшествующей и необходимой для изучения следующих дисциплин:

- программирование;
- объектно-ориентированное программирование;
- технологии обработки информации;
- программирование в MS Office;
- основы программной инженерии.

3. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций

Таблица 1

Код по ФГОС	Индикаторы достижения	Планируемые результаты обучения по дисциплине (ЗУВ)
Общепрофессиональные		
ОПК-1 Способен применять естественнонаучные и общепрофессиональные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной	ОПК-1.1. Знает основы математики, физики, вычислительной техники и программирования ОПК-1.2. Умеет решать стандартные профессиональные задачи с применением естественнонаучных и общепрофессиональных знаний, методов математического анализа и моделирования	Знать: методы и приемы решения практических задач в профессиональной деятельности с помощью информационных систем. Уметь: решать базовые задачи обработки данных в профессиональной деятельности. Владеть: общей подготовкой для решения практических задач в области

деятельности	ОПК-1.3. Имеет навыки теоретического и экспериментального исследования объектов профессиональной деятельности	информационных технологий.
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ОПК-6.1. Знает основы информатики, основные языки программирования и программы, пригодные для практического использования ОПК-6.2. Умеет применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов ОПК-6.2. Имеет навыки в разработке алгоритмов и программ, пригодных для практического использования	Знать: принципы, базовые концепции технологий программирования. Уметь: составлять алгоритмы решения задач различной структуры и оформлять их в соответствии с синтаксическими правилами языка программирования. Владеть: приемами разработки пользовательского интерфейса приложения, обеспечивающего оптимальное функционирование программы и удобство пользователя.

4. Объем дисциплины и виды учебной работы

Таблица 2

Вид учебной работы	Всего часов/ зач. ед.	Семестр 2
	ОФО	ОФО
Контактная работа (всего)	64/1,8	64/1,8
В том числе:		
Лекции	32/0,9	32/0,9
Практические занятия	-	-
Практическая подготовка	-	-
Лабораторные занятия	32/0,9	32/0,9
Самостоятельная работа (всего)	80/2,2	80/2,2
В том числе:		
Курсовая работа (проект)	-	-
Расчетно-графические работы	-	-
ИТР	-	-
Рефераты	-	-
Проектная деятельность	36/1	36/1
<i>И (или) другие виды самостоятельной работы:</i>		
Подготовка к лабораторным работам	18/0,5	18/0,5
Подготовка к практическим занятиям	-	-
Подготовка к зачету	-	-
Подготовка к экзамену	26/0,7	26/0,7

Вид отчетности			ЭКЗ.
Общая трудоемкость дисциплины	ВСЕГО в часах	144	144
	ВСЕГО в зач. ед.	4	4

5. Содержание дисциплины

5.1. Разделы дисциплины и виды занятий

Таблица 3

№ п/п	Наименование раздела дисциплины по семестрам	Часы лекционных занятий	Часы лабораторных занятий	Часы практических (семинарских) занятий	Всего часов
2 семестр					
1.	Основы алгоритмизации и программирования	8	8	-	16
2.	Основы синтаксиса языка программирования С#	14	24	-	38
3.	Методология проектирования программных продуктов	10	-	-	10

5.2. Лекционные занятия

Таблица 4

№ п/п	Наименование раздела дисциплины	Содержание раздела
1.	Основы алгоритмизации и программирования	<p>Понятие алгоритма. Формы представления алгоритмов. Блок-схемы, используемые в графическом представлении. Свойства алгоритмов.</p> <p>Основные этапы программирования решения задачи.</p> <p>Языки программирования низкого и высокого уровня. Трансляторы.</p> <p>Основные понятия ООП: объект, класс, инкапсуляция, наследование, полиморфизм, свойство, метод, событие, оператор, переменная.</p>
2.	Основы синтаксиса языка программирования С#	<p>Типы данных в Visual С#. Объявление переменных. Инкремент и декремент. Операторы сравнения и логические операторы.</p> <p>Условные ветвления в С# (операторы if и switch).</p> <p>Реализация классических и итерационных циклов в С# (операторы for и while).</p> <p>Модальные и немодальные диалоговые окна. Окна вывода сообщений MessageBox.</p>

3.	Методология проектирования программных продуктов	Основные участники и их роли в разработке программного продукта. Модели предметной области в процедурном и объектно-ориентированном программировании. Диаграммы UML в объектно-ориентированном программировании. Понятие системы и среды программирования. Особенности современных сред разработки приложений (IDE).
----	--	--

5.3. Лабораторные занятия

Таблица 5

№ п/п	Наименование раздела дисциплины	Наименование лабораторных работ
1.	Основы алгоритмизации и программирования	Графическая форма представления алгоритмов. Блок-схемы. Составление алгоритмов решения линейных задач. Алгоритмы разветвляющейся структуры (условия). Решение задач с условиями. Алгоритмы циклической структуры. Решение задач с циклами.
2.	Основы синтаксиса языка программирования C#	Знакомство со средой разработки приложений Microsoft Visual Studio. Приемы создания интерфейса приложения. Элементы управления, размещаемые на форме. Переменные и типы данных в Visual C#. Основы синтаксиса языка. Разработка линейных программ на языке C#. Алгоритмы с условными ветвлениями. Конструкция if. Алгоритмы с условными ветвлениями. Конструкция switch. Программирование циклов в языке C#: алгоритмы с повторяющимися действиями. Реализация классических и итерационных циклов в C#. Использование окон вывода сообщений MessageBox.

5.4. Практические (семинарские) занятия: нет

Таблица 6

№ п/п	Наименование раздела дисциплины	Содержание раздела
1.	-	-

6. Самостоятельная работа студентов по дисциплине

Способ организации самостоятельной работы: разработка проектов приложений по заданию преподавателя.

Пример задания

Разработать программу, демонстрирующую работу одного из элементов интерфейса в MS Visual Studio – его назначение, внешний вид, особенности использования и основные свойства:

1. checkedListBox
2. colorDialog
3. contextMenuStrip
4. dateTimePicker
5. domainUpDown
6. menuStrip
7. monthCalendar
8. notifyIcon
9. numericUpDown
10. progressBar
11. statusStrip
12. timer
13. toolStrip
14. trackBar

Учебно-методическое обеспечение для самостоятельной работы студентов:

1. Горелов, С.В. Современные технологии программирования: разработка Windows-приложений на языке C#. В 2 томах. Т.1: учебник / С.В. Горелов; под редакцией П.Б. Лукьянова. – Москва: Прометей, 2019. – 362 с. Режим доступа: <https://www.iprbookshop.ru/94532.html> (ЭБС «IPRbooks»).
2. Лебедева, Т.Н. Технология программирования: учебное пособие / Т.Н. Лебедева, С.С. Юнусова. – 2-е изд. – Челябинск, Саратов: Южно-Уральский институт управления и экономики, Ай Пи Эр Медиа, 2019. – 140 с. – Режим доступа: <https://www.iprbookshop.ru/81500.html> (ЭБС «IPRbooks»).

7. Оценочные средства

7.1. Вопросы к рубежным аттестациям

К 1-ой рубежной аттестации:

1. Понятие алгоритма. Формы представления алгоритмов. Блок-схемы, используемые в графическом представлении.
2. Свойства алгоритмов.
3. Основные этапы программирования решения задачи.
4. Типы данных в Visual C#. Объявление переменных. Инкремент и декремент. Операторы сравнения и логические операторы.
5. Условные ветвления в C# (операторы if и switch).
6. Реализация классических и итерационных циклов в C# (операторы for и while).
7. Модальные и немодальные диалоговые окна. Окна вывода сообщений MessageBox.
8. Рассказать о некоторых основных компонентах, применяемых для построения интерфейса в среде Visual Studio (button, label, textBox, comboBox, radioButton, checkBox).

Ко 2-ой рубежной аттестации:

1. Языки программирования низкого и высокого уровня. Трансляторы.
2. Основные понятия ООП: объект, класс, инкапсуляция, наследование, полиморфизм.
3. Основные понятия ООП: свойство, метод, событие, оператор, переменная.
4. Основные участники и их роли в разработке программного продукта.
5. Модели предметной области в процедурном и объектно-ориентированном

программировании.

6. Диаграммы UML в объектно-ориентированном программировании.
7. Понятие системы и среды программирования.
8. Схемы компиляции и интерпретации.
9. Особенности современных сред разработки приложений (IDE).

Помимо проверки знания теоретического материала, на аттестации / экзамене студентам предлагаются практические задания по разделам дисциплины.

Образец билетов рубежной аттестации:

Грозненский Государственный Нефтяной Технический Университет им. акад. М.Д. Миллионщикова Кафедра «Информационные технологии» Дисциплина «Технологии программирования» 1-я рубежная аттестация	
Группа:	Семестр: 2
Билет №	
1. Основные этапы программирования решения задачи.	
2. Вводятся три числа. Программа должна указать, является ли их сумма двузначным числом.	
Преподаватель _____	

Грозненский Государственный Нефтяной Технический Университет им. акад. М.Д. Миллионщикова Кафедра «Информационные технологии» Дисциплина «Технологии программирования» 2-я рубежная аттестация	
Группа:	Семестр: 2
Билет №	
1. Основные понятия ООП: объект, класс, инкапсуляция, наследование, полиморфизм.	
2. Пользователь вводит число N. Найти сумму и произведение чисел в диапазоне от N до 2N.	
Преподаватель _____	

7.2. Вопросы к экзамену

1. Понятие алгоритма. Формы представления алгоритмов. Блок-схемы, используемые в графическом представлении.
2. Свойства алгоритмов.
3. Основные этапы программирования решения задачи.
4. Типы данных в Visual C#. Объявление переменных. Инкремент и декремент. Операторы сравнения и логические операторы.
5. Условные ветвления в C# (операторы if и switch).
6. Реализация классических и итерационных циклов в C# (операторы for и while).
7. Модальные и немодальные диалоговые окна. Окна вывода сообщений MessageBox.
8. Языки программирования низкого и высокого уровня. Трансляторы.
9. Основные понятия ООП: объект, класс, инкапсуляция, наследование, полиморфизм.
10. Основные понятия ООП: свойство, метод, событие, оператор, переменная.

11. Основные участники и их роли в разработке программного продукта.
12. Модели предметной области в процедурном и объектно-ориентированном программировании.
13. Диаграммы UML в объектно-ориентированном программировании.
14. Понятие системы и среды программирования.
15. Схемы компиляции и интерпретации.
16. Особенности современных сред разработки приложений (IDE).

Образец билета к экзамену:

<p>Грозненский Государственный Нефтяной Технический Университет им. акад. М.Д. Миллионщикова Кафедра «Информационные технологии» Дисциплина «Технологии программирования» Группа: _____ Семестр: 2</p>	
<p>Билет № _____</p>	
<p>1. Понятие алгоритма. Формы представления алгоритмов. Блок-схемы, используемые в графическом представлении.</p>	
<p>2. Вводится порядковый номер месяца в году. Определить, является ли этот месяц зимним или летним.</p>	
<p>Подпись преподавателя _____ Подпись заведующего кафедрой _____</p>	

7.3. Текущий контроль

Образец типового задания для лабораторных занятий

Лабораторная работа на тему «Программирование циклов в языке C#»

Известно, что ежемесячная стипендия студента составляет A руб., а расходы на проживание превышают стипендию и составляют B руб. в месяц. Рост цен ежемесячно увеличивает расходы на 3%.

Составьте программу расчета суммы денег, которую необходимо единовременно попросить у родителей, чтобы можно было прожить учебный год (10 месяцев), используя только эти деньги и стипендию.

Программный код

```
private void button1_Click(object sender, EventArgs e)
{
```

A – стипендия, B – расходы в месяц, R – разница между стипендией и расходами в месяц, S – общая сумма, которую нужно просить. Тип данных – double, дробные числа.

```
double A, B, R, S;
int i;
A = Convert.ToDouble(textBox1.Text);
B = Convert.ToDouble(textBox2.Text);
```



```
S = 0;
```

Вычисления производятся за каждый месяц в отдельности с помощью цикла for. Счетчик цикла меняется в диапазоне от 1 до 10; i++ – операция инкремента, дающая увеличение значения счетчика на 1 после каждого прохождения цикла.

За каждый месяц определяется разница между стипендией и расходами, которая прибавляется к итоговой сумме S. Перед переходом на следующий месяц расходы B увеличиваются на 3%.

```
for (i = 1; i<=10; i++)  
{  
    R = B - A;  
    S = S + R;  
    B = B * 1.03;  
}
```

Полученное значение суммы округляется до двух знаков после запятой и для вывода в текстовое поле конвертируется в формат string.

```
S = Math.Round(S, 2);  
textBox3.Text = Convert.ToString(S);  
}
```

7.4. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкалы оценивания

Таблица 7

Планируемые результаты освоения компетенции	Критерии оценивания результатов обучения				Наименование оценочного средства
	менее 41 баллов (неудовлетворительно)	41-60 баллов (удовлетворительно)	61-80 баллов (хорошо)	81-100 баллов (отлично)	
ОПК-1: Способность применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности					
Знать: методы и приемы решения практических задач в профессиональной деятельности с помощью информационных систем.	Фрагментарные знания	Неполные знания	Сформированные, но содержащие отдельные пробелы знания	Сформированные систематические знания	Комплект заданий для выполнения лабораторных работ, темы докладов с презентациями, вопросы по темам / разделам дисциплины
Уметь: решать базовые задачи обработки данных в профессиональной деятельности.	Частичные умения	Неполные умения	Умения полные, допускаются небольшие ошибки	Сформированные умения	
Владеть: общей подготовкой для решения практических задач в области информационных технологий.	Частичное владение навыками	Несистематическое применение навыков	В систематическом применении навыков допускаются пробелы	Успешное и систематическое применение навыков	
ОПК-6: Способность разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов					
Знать: принципы, базовые концепции технологий программирования.	Фрагментарные знания	Неполные знания	Сформированные, но содержащие отдельные пробелы знания	Сформированные систематические знания	Комплект заданий для выполнения лабораторных работ, темы докладов с презентациями, вопросы по темам / разделам дисциплины
Уметь: составлять алгоритмы решения задач различной структуры и оформлять их в соответствии с синтаксическими правилами языка программирования.	Частичные умения	Неполные умения	Умения полные, допускаются небольшие ошибки	Сформированные умения	

Владеть: приемами разработки пользовательского интерфейса приложения, обеспечивающего оптимальное функционирование программы и удобство пользователя.	Частичное владение навыками	Несистематическое применение навыков	В систематическом применении навыков допускаются пробелы	Успешное и систематическое применение навыков
--	-----------------------------	--------------------------------------	--	---

8. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

Для осуществления процедур текущего контроля успеваемости и промежуточной аттестации обучающихся созданы фонды оценочных средств, адаптированные для инвалидов и лиц с ограниченными возможностями здоровья и позволяющие оценить достижение ими запланированных в основной образовательной программе результатов обучения и уровень сформированности всех компетенций, заявленных в образовательной программе. Форма проведения текущей аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.). При тестировании для слабовидящих студентов используются фонды оценочных средств с укрупненным шрифтом. На экзамен приглашается сопровождающий, который обеспечивает техническое сопровождение студенту. При необходимости студенту-инвалиду предоставляется дополнительное время для подготовки ответа на экзамене (или зачете). Обучающиеся с ограниченными возможностями здоровья и обучающиеся инвалиды обеспечиваются печатными и электронными образовательными ресурсами (программы, учебные пособия для самостоятельной работы и т.д.) в формах, адаптированных к ограничениям их здоровья и восприятия информации:

1) для инвалидов и лиц с ограниченными возможностями здоровья **по зрению:**

- **для слепых:** задания для выполнения на семинарах и практических занятиях оформляются рельефно-точечным шрифтом Брайля или в виде электронного документа, доступного с помощью компьютера со специализированным программным обеспечением для слепых, либо зачитываются ассистентом; письменные задания выполняются на бумаге рельефно-точечным шрифтом Брайля или на компьютере со специализированным программным обеспечением для слепых либо надиктовываются ассистенту; обучающимся для выполнения задания при необходимости предоставляется комплект письменных принадлежностей и бумага для письма рельефно-точечным шрифтом Брайля, компьютер со специализированным программным обеспечением для слепых;

- **для слабовидящих:** обеспечивается индивидуальное равномерное освещение не менее 300 люкс; обучающимся для выполнения задания при необходимости предоставляется увеличивающее устройство; возможно также использование собственных увеличивающих устройств; задания для выполнения заданий оформляются увеличенным шрифтом;

2) для инвалидов и лиц с ограниченными возможностями здоровья **по слуху:**

- **для глухих и слабослышащих:** обеспечивается наличие звукоусиливающей аппаратуры коллективного пользования, при необходимости обучающимся предоставляется звукоусиливающая аппаратура индивидуального пользования; предоставляются услуги сурдопереводчика;

- **для слепоглухих** допускается присутствие ассистента, оказывающего услуги тифлосурдопереводчика (помимо требований, выполняемых соответственно для слепых и глухих);

3) для лиц с тяжелыми нарушениями речи, глухих, слабослышащих лекции и семинары, проводимые в устной форме, проводятся в письменной форме;

4) для инвалидов и лиц с ограниченными возможностями здоровья, **имеющих**

нарушения опорно-двигательного аппарата:

- для лиц с нарушениями опорно-двигательного аппарата, нарушениями двигательных функций верхних конечностей или отсутствием верхних конечностей: письменные задания выполняются на компьютере со специализированным программным обеспечением или надиктовываются ассистенту; выполнение заданий (тестов, контрольных работ), проводимые в письменной форме, проводятся в устной форме путем опроса, беседы с обучающимся.

9. Учебно-методическое и информационное обеспечение дисциплины

1. Горелов, С.В. Современные технологии программирования: разработка Windows-приложений на языке C#. В 2 томах. Т.І: учебник / С.В. Горелов; под редакцией П.Б. Лукьянова. – Москва: Прометей, 2019. – 362 с. Режим доступа: <https://www.iprbookshop.ru/94532.html> (ЭБС «IPRbooks»).

2. Горелов, С.В. Современные технологии программирования: разработка Windows-приложений на языке C#. В 2 томах. Т.ІІ: учебник / С.В. Горелов; под редакцией П.Б. Лукьянова. – Москва: Прометей, 2019. – 378 с. – Режим доступа: <https://www.iprbookshop.ru/94533.html> (ЭБС «IPRbooks»).

3. Лебедева, Т.Н. Технология программирования: учебное пособие / Т.Н. Лебедева, С.С. Юнусова. – 2-е изд. – Челябинск, Саратов: Южно-Уральский институт управления и экономики, Ай Пи Эр Медиа, 2019. – 140 с. – Режим доступа: <https://www.iprbookshop.ru/81500.html> (ЭБС «IPRbooks»).

4. Згуральская, Е.Н. Технологии программирования: учебное пособие / Е.Н. Згуральская. – Ульяновск: Ульяновский государственный технический университет, 2020. – 72 с. – Режим доступа: <https://www.iprbookshop.ru/106123.html> (ЭБС «IPRbooks»).

5. Златопольский, Д.М. Программирование: типовые задачи, алгоритмы, методы / Д.М. Златопольский. – 4-е изд. – Москва: Лаборатория знаний, 2020. – 224 с. – Режим доступа: <https://www.iprbookshop.ru/12264.html> (ЭБС «IPRbooks»).

10. Материально-техническое обеспечение дисциплины

10.1. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Перечень материально-технических средств учебной аудитории для проведения занятий по дисциплине:

- учебная аудитория, доска;
- стационарные компьютеры;
- мультимедийный проектор;
- настенный экран.

10.2. Помещения для самостоятельной работы

Учебная аудитория для самостоятельной работы – 4-01.

Методические указания по освоению дисциплины «Технологии программирования»

1. Методические указания для обучающихся по планированию и организации времени, необходимого для освоения дисциплины

Изучение рекомендуется начать с ознакомления с рабочей программой дисциплины, ее структурой и содержанием разделов (модулей), фондом оценочных средств, ознакомиться с учебно-методическим и информационным обеспечением дисциплины.

Дисциплина «Технологии программирования» состоит из трех связанных между собой разделов, обеспечивающих последовательное изучение материала.

Обучение дисциплине «Технологии программирования» осуществляется в следующих формах:

1. Аудиторные занятия (лекции, лабораторные занятия).
2. Самостоятельная работа студента (подготовка к лекциям, лабораторным занятиям, доклады с презентациями, индивидуальная консультация с преподавателем).

Учебный материал структурирован, и изучение дисциплины производится в тематической последовательности. Каждому лабораторному занятию и самостоятельному изучению материала предшествует лекция по данной теме. Обучающиеся самостоятельно проводят предварительную подготовку к занятию, принимают активное и творческое участие в обсуждении теоретических вопросов, разборе проблемных ситуаций и поисков путей их решения.

Описание последовательности действий обучающегося:

При изучении курса следует внимательно слушать и конспектировать материал, излагаемый на аудиторных занятиях. Для его понимания и качественного усвоения рекомендуется следующая последовательность действий:

1. После окончания учебных занятий для закрепления материала просмотреть и обдумать текст лекции, прослушанной сегодня, разобрать рассмотренные примеры (10- 15 минут).
2. При подготовке к лекции следующего дня повторить текст предыдущей лекции, подумать о том, какая может быть следующая тема (10-15 минут).
3. В течение недели выбрать время для работы с литературой в электронной библиотечной системе (по 1 часу).
4. При подготовке к лабораторному занятию повторить основные понятия по теме, изучить примеры. Решая конкретную ситуацию, – предварительно понять, какой теоретический материал нужно использовать. Наметить план решения, попробовать на его основе решить 1-2 задачи.

2. Методические указания по работе обучающихся во время проведения лекций

Лекции дают обучающимся систематизированные знания по дисциплине, концентрируют их внимание на наиболее сложных и важных вопросах. Лекции обычно излагаются в традиционном или в проблемном стиле. Для студентов в большинстве случаев в проблемном стиле. Проблемный стиль позволяет стимулировать активную познавательную деятельность обучающихся и их интерес к дисциплине, формировать творческое мышление, прибегать к противопоставлениям и сравнениям, делать обобщения, активизировать внимание обучающихся путем постановки проблемных вопросов, поощрять дискуссию.

Во время лекционных занятий рекомендуется вести конспектирование учебного материала, обращать внимание на формулировки и категории, раскрывающие суть того или иного явления, выводы и практические рекомендации.

Работая над конспектом лекций, необходимо использовать также литературу, которую дополнительно рекомендовал преподаватель. Тематика лекций дается в рабочей

программе дисциплины.

3. Методические указания обучающимся по подготовке к лабораторным занятиям

На лабораторных занятиях приветствуется активное участие в обсуждении конкретных ситуаций, способность на основе полученных знаний находить наиболее эффективные решения задач, уметь находить полезный дополнительный материал по тематике занятий.

Студенту рекомендуется следующая схема подготовки к лабораторному занятию:

1. Ознакомиться с планом занятия, который отражает содержание предложенной темы.

2. Проработать конспект лекций.

3. Прочитать основную и дополнительную литературу.

В процессе подготовки к лабораторным занятиям, необходимо обратить особое внимание на самостоятельное изучение рекомендованной литературы. Самостоятельная работа с учебниками, учебными пособиями, научной, справочной литературой, материалами периодических изданий и Интернета является наиболее эффективным методом получения дополнительных знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у студентов отношение к конкретной проблеме.

4. Выполнить домашнее задание.

5. При затруднениях сформулировать вопросы к преподавателю.

Результат такой работы должен проявиться в способности студента свободно ответить на теоретические вопросы, выступать и участвовать в коллективном обсуждении вопросов изучаемой темы, правильно выполнять практические задания, которые даются в фонде оценочных средств дисциплины.

4. Методические указания обучающимся по организации самостоятельной работы

Цель организации самостоятельной работы по дисциплине «Технологии программирования» – это углубление и расширение знаний в области алгоритмизации и программирования; формирование навыка и интереса к самостоятельной познавательной деятельности.

Самостоятельная работа обучающихся является важнейшим видом освоения содержания дисциплины, подготовки к практическим занятиям и к рубежной аттестации. Самостоятельная работа носит исследовательский характер, что послужит в будущем основанием для написания выпускной квалификационной работы, практического применения полученных знаний.

Организация самостоятельной работы обучающихся ориентируется на активные методы овладения знаниями, развитие творческих способностей, переход от поточного к индивидуализированному обучению, с учетом потребностей и возможностей личности.

Правильная организация самостоятельных учебных занятий, их систематичность, целесообразное планирование рабочего времени позволяет студентам развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивать высокий уровень успеваемости в период обучения, получить навыки повышения профессионального уровня.

Подготовка к лабораторному занятию включает, кроме проработки конспекта и презентации лекции, поиск литературы (по рекомендованным спискам и самостоятельно).

При подготовке к контрольной работе (рубежной аттестации) обучающийся должен повторять пройденный материал в строгом соответствии с учебной программой, используя конспект лекций и литературу, рекомендованную преподавателем. При необходимости можно обратиться за консультацией и методической помощью к преподавателю.

Самостоятельная работа реализуется:

- непосредственно в процессе аудиторных занятий – на лекциях, лабораторных занятиях;
- в контакте с преподавателем вне рамок расписания – на консультациях по учебным вопросам, в ходе творческих контактов, при ликвидации задолженностей, при выполнении индивидуальных заданий и т.д.
- в библиотеке, дома, на кафедре при выполнении обучающимся учебных и практических задач.

Виды СРС и критерии оценок


(по балльно-рейтинговой системе ГГНТУ, СРС оценивается в 15 баллов)

1. Работа над проектами
2. Подготовка к лабораторным занятиям

Темы для самостоятельной работы прописаны в рабочей программе дисциплины. Эффективным средством осуществления обучающимся самостоятельной работы является электронная информационно-образовательная среда университета, которая обеспечивает доступ к учебным планам, рабочим программам дисциплин (модулей), практик, к изданиям электронных библиотечных систем.

Составитель:

Доцент кафедры
«Информационные технологии»


 / Мачуева Д.А. /

СОГЛАСОВАНО:

Зав. выпускающей кафедрой
«Информационные технологии»

 / Моисеенко Н.А. /

Директор ДУМР

 / Магомаева М.А. /