

МИНПРОСВЕЩЕНИЯ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования
"Тульский государственный педагогический университет им. Л.Н. Толстого"
(ФГБОУ ВО "ТГПУ им. Л.Н. Толстого")

Алгоритмизация и программирование

рабочая программа дисциплины (модуля)

ОПОП	01.03.01 Математика направленность (профиль) Математика
Квалификация	Бакалавр
Год начала подготовки	2023
Форма обучения	очная
Общая трудоемкость	2 з.е.

Виды контроля по семестрам:
зачет 5

Семестр(Курс.Номер семестра на курсе)	5(3.1)		Итого	
	УП	РПД	УП	РПД
Лекции	18	18	18	18
Лабораторные	18	18	18	18
Итого ауд.	36	36	36	36
КСР	2	2	2	2
Контактная работа	38	38	38	38
Сам. работа	34	34	34	34
Часы на контроль	0	0	0	0
Практическая подготовка	0	0	0	0
Семинары	0	0	0	0
Консультации	0	0	0	0
Итого трудоемкость в часах	72	72	72	72

Программу составил(и):

к.п.н., доцент, Ситникова Людмила Дмитриевна; нет, ассистент, Яковлева Надежда Александровна

Рабочая программа дисциплины

Алгоритмизация и программирование

разработана в соответствии с ФГОС:

Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки **01.03.01 Математика** (приказ Минобрнауки России от 10.01.2018 г. № 8)

составлена на основании учебного плана:

01.03.01 Математика

направленность (профиль) Математика

утвержденного Учёным советом вуза от 27.10.2022 протокол № 13.

РПД утверждена Учёным советом университета
от 27.10.2022 протокол № 13.

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Формирование способности программировать на языке C++, поддерживающем процедурную, объектно-ориентированную и обобщенную парадигмы.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП

Цикл (раздел) ООП:	Б1.В
2.1 Требования к предварительной подготовке обучающегося:	
1.	Практикум на ЭВМ .
2.2 Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:	
1.	Введение в искусственный интеллект
2.	научно-исследовательская работа
3.	Выполнение и защита выпускной квалификационной работы

3. СООТНЕСЕНИЕ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ) С ИНДИКАТОРАМИ ДОСТИЖЕНИЯ КОМПЕТЕНЦИЙ

3.1 Компетенции обучающегося и индикаторы их достижения:

ПК-1: Способен понимать и применять в исследовательской и прикладной деятельности современный математический аппарат, фундаментальные концепции и системные методологии, международные и профессиональные стандарты в области информационных технологий, способность использовать современные инструментальные и вычислительные средства

ПК-1.1	Знать базовый современный математический аппарат, базовые фундаментальные концепции и системные методологии, международные и профессиональные стандарты в области информационных технологий, стандартный функционал современных инструментальных и вычислительных средств
	знает принципы работы современных информационных технологий; современные тенденции развития информатики и вычислительной техники, цифровых компьютерных технологий знает основные алгоритмические конструкции; знает методы программирования, знает методы обработки числовой, текстовой и структурированной информации; знает алгоритмы решения типовых задач по программированию;
ПК-1.2	Уметь использовать при решении конкретных научно-исследовательских и прикладных задач математический аппарат и информационные технологии
	умеет применять современные информационные технологии и использовать возможности программных средств в профессиональной деятельности умеет представлять отчеты по результатам решения задач; умеет выбирать метод программирования в зависимости от типов данных; умеет разрабатывать алгоритм в соответствии с этапами решения задачи; умеет реализовывать алгоритмы средствами языка программирования процедурной парадигмы;
ПК-1.3	Владеть навыками применения математического аппарата и информационных технологий при решении научно-исследовательских и практических задач, в том числе с применением современных инструментальных и вычислительных средств
	владеет технологией разработки алгоритмов типовых задач по программированию владеет основными методами обработки числовых, текстовых и структурированных данных.

3.2 Результаты обучения по дисциплине:

В результате освоения дисциплины обучающийся должен:

	Знать:
3.1	основные алгоритмические конструкции,
3.2	методы программирования,
3.3	методы обработки числовой, текстовой и структурированной информации;
3.4	алгоритмы решения типовых задач по программированию.
	Уметь:
У.1	выбирать метод программирования в зависимости от типов данных;
У.2	разрабатывать алгоритм в соответствии с этапами решения задачи;
У.3	реализовывать алгоритмы средствами языка программирования процедурной парадигмы;
У.4	представлять отчеты по результатам решения задач.
	Владеть:
В.1	технологией разработки алгоритмов типовых задач по программированию,
В.2	основными методами обработки числовых,
В.3	текстовых и структурированных данных.

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Литература	Содержание
	Алгоритмическая система и ее составные части				
1.1	Алгоритмическая система как совокупность средств и понятий. /Лек/	1	1	Л1.1Л2.1 Л2.2	Понятие алгоритма и исполнителя алгоритма. Представление информации в виде данных: понятие о вводе/выводе; входные, выходные и промежуточные данные. Система команд исполнителя.
1.2	Способы представления алгоритмов. /Лаб/	1	1	Л1.1Л2.1 Л2.2	Свойства алгоритма. Способы записи алгоритма. Основные элементы блок-схемы. Виды алгоритмов. Отличительные особенности алгоритмов с предусловием и постусловием.
1.3	Алгоритмический язык исполнителя. /Ср/	1	3	Л1.1Л2.1 Л2.2	Представление в компьютере целых чисел. Представление в компьютере действительных чисел. Средства записи алгоритмов. Схемы алгоритмов
	Основные понятия и простейшие средства алгоритмических языков программирования				

2.1	Программирование как этап решения задачи на компьютере. /Лек/	1	1	Л1.1Л2.1 Л2.2	Понятие языка программирования высокого уровня. Понятие о исполнителе программы на языке высокого уровня. Составные части языка высокого уровня: алфавит, синтаксис, семантика. Метаязыки описания конструкций языков программирования. Структуры программ и концепции типов данных в алгоритмических языках на примере языков. Оператор как логически завершенная конструкция алгоритмического языка программирования. Различные классификации операторов: исполняемые и неисполняемые, простые и составные. Различные классификации типов данных: простые и структурированные, стандартные и определяемые пользователем. Основные простые типы данных в C/C++. Синтаксис и семантика оператора присваивания, знакомство с операторами ввода/вывода.
2.2	Знакомство со средой Microsoft Visual Studio /Лаб/	1	1	Л1.1Л2.1 Л2.2	Создание простейших проектов в среде программирования Microsoft Visual Studio. Расширения (типы) файлов, создаваемая в среде программирования Microsoft Visual Studio. Структура программы на языке C/C++.
2.3	Типы данных. Операторы ввода и вывода данных. Форматированный ввод/вывод данных различных типов. Стандартные потоки ввода-вывода данных. /Лаб/	1	1	Л1.1Л2.1 Л2.2	Типы данных. Операторы ввода и вывода данных. Форматированный ввод/вывод данных различных типов. Стандартные потоки ввода-вывода данных.
2.4	Константы. Присваивание. Арифметические операции. /Лаб/	1	1	Л1.1Л2.1 Л2.2	Константы. Числовые константы: целочисленные константы, вещественные константы, символьные константы, строковые константы, перечислимые константы. Присваивание. Арифметические операции. Приоритеты операций.
2.5	Логические операции. Математические функции. Преобразование типов. /Лаб/	1	1	Л1.1Л2.1 Л2.2	Логические операции. операции отношений Сдвиговые операции. Математические функции. Преобразование типов.
2.6	Написание, ввод, отладка и тестирование программ линейной структуры. /Ср/	1	3	Л1.1Л2.1 Л2.2	Написание, ввод, отладка и тестирование программ линейной структуры в среде программирования Microsoft Visual Studio.
	Разработка разветвляющихся и циклических алгоритмов и программ				
3.1	Условные конструкции ветвления и циклы в языке C/C++. /Лек/	1	2	Л1.1Л2.1 Л2.2	Условные конструкции ветвления в языке СИ. Конструкции if, if else, switch. Простые и вложенные условные блоки. Порядок выполнения условий. Циклы в языке СИ. Цикл for и его параметры, Цикл while, принципы использования. Выбор конструкции цикла в зависимости от условия задачи.
3.2	Операторы выбора: условный оператор if, оператор-переключатель switch. Операторы перехода. /Лаб/	1	1	Л1.1Л2.1 Л2.2	Операторы выбора: условный оператор if, оператор-переключатель switch. Синтаксис условного оператора с одной ветвью (неполное ветвление). Синтаксис условного оператора с двумя ветвями (полное ветвление). Вложенные условные операторы. Операторы перехода: goto, break, continue, return.

3.3	Операторы цикла: цикл с предусловием, с постусловием, с параметром. /Лаб/	1	1	Л1.1Л2.1 Л2.2	Операторы цикла: цикл с предусловием, с постусловием, с параметром. Цикл с предусловием while. Цикл с постусловием do...while. Цикл с параметром for. Рекомендации по выбору цикла.
3.4	Алгоритмические структуры ветвлений и циклов. /Ср/	1	4	Л1.1Л2.1 Л2.2	Базовые алгоритмические структуры ветвления. Базовая алгоритмическая структура цикл с предусловием. Цикл с постусловием. Проблема заикливания. Цикл с параметром.
	Функции пользователя и рекурсия				
4.1	Функции пользователя. Рекурсивные функции. Указатели. /Лек/	1	2	Л1.1Л2.1 Л2.2	Описание функций пользователя: объявление и вызов. Перегрузка функций. Рекурсивные функции. Рекурсивная триада. Указатели: на объект, на функцию, на указатель.
4.2	Функции пользователя. /Лаб/	1	1	Л1.1Л2.1 Л2.2	Описание функций пользователя: объявление и вызов. Перегрузка функций. Заголовочные файлы.
4.3	Рекурсивные функции. /Лаб/	1	1	Л1.1Л2.1 Л2.2	Рекурсивные функции. Рекурсивная триада. Решение задач рекурсивным способом. Примеры решения задач рекурсивным способом.
4.4	Указатели /Лаб/	1	1	Л1.1Л2.1 Л2.2	Указатели: на объект, на функцию, на указатель. Указатели на функции как параметры.
4.5	Использование функций пользователя и рекурсивных алгоритмов в языке C/C++. /Ср/	1	4	Л1.1Л2.1 Л2.2	Примеры использования функций пользователя. Основные функции пользователя. Рекурсивные алгоритмы.
	Массивы				
5.1	Одномерные и двумерные массивы /Лек/	1	2	Л1.1Л2.1 Л2.2	Одномерные массивы: объявление, инициализация, задачи поиска, замены и перестановок элементов массива, задачи сортировок элементов массива. Двумерные массивы: задачи поиска, замены и суммирования элементов двумерного массива. Двумерные массивы: задачи сортировок и перестановок в двумерных массивах.
5.2	Одномерные массивы /Лаб/	1	1	Л1.1Л2.1 Л2.2	Одномерные массивы: объявление, инициализация, задачи поиска, замены и перестановок элементов массива, задачи сортировок элементов массива.
5.3	Двумерные массивы /Лаб/	1	1	Л1.1Л2.1 Л2.2	Двумерные массивы: задачи поиска, замены и суммирования элементов двумерного массива. Двумерные массивы: задачи сортировок и перестановок в двумерных массивах.
5.4	Применение массивов /Ср/	1	4	Л1.1Л2.1 Л2.2	Примеры использования массивов для решения задач.
5.5	/КСР/	1	1		КСР
	Символьные данные и строки				
6.1	Символьные данные и строки. Функции работы с ними /Лек/	2	2	Л1.1Л2.1 Л2.2	Символьные данные и строки. Функции работы с ними
6.2	Обработка символьных данных и строк /Лаб/	2	1	Л1.1Л2.1 Л2.2	Символьные данные и строки. Функции работы с ними
6.3	Символьные данные и строки /Ср/	2	4	Л1.1Л2.1 Л2.2	Примеры технологий обработки символьных данных и строк для решения задач.
	Структуры				
7.1	Структуры /Лек/	2	2	Л1.1Л2.1 Л2.2	Объявления и определение структурных объектов, инициализация, присваивание структур, доступ к элементам структур

7.2	Структуры /Лаб/	2	1	Л1.1Л2.1 Л2.2	Объявления и определение структурных объектов, инициализация, присваивание структур, доступ к элементам структур
7.3	Структуры /Ср/	2	4	Л1.1Л2.1 Л2.2	Примеры использования С++ для объявления и определения структурных объектов, инициализации, присваивания структур, доступа к элементам структур
	Файлы и работа с ними				
8.1	Файлы и работа с ними /Лек/	2	2	Л1.1Л2.1 Л2.2	Стандартные файлы и функции по работе с ними: открытие файла, закрытие, удаление, переименование, функция контроля конца файла, ввод и вывод данных (символьный, строковый, блоковый, форматированный)
8.2	Файлы и работа с ними /Лаб/	2	2	Л1.1Л2.1 Л2.2	Стандартные файлы и функции по работе с ними: открытие файла, закрытие, удаление, переименование, функция контроля конца файла, ввод и вывод данных (символьный, строковый, блоковый, форматированный)
8.3	Файлы и работа с ними /Ср/	2	4	Л1.1Л2.1 Л2.2	Стандартные файлы и функции по работе с ними: открытие файла, закрытие, удаление, переименование, функция контроля конца файла, ввод и вывод данных (символьный, строковый, блоковый, форматированный)
	Динамическое программирование				
9.1	Динамическое программирование /Лек/	2	4	Л1.1Л2.1 Л2.2	Разделение памяти, динамическое выделение памяти. Одномерные и двумерные динамические массивы. Динамические структуры данных. Однонаправленные и двунаправленные списки. Стеки и очереди
9.2	Динамическое программирование /Лаб/	2	2	Л1.1Л2.1 Л2.2	Разделение памяти, динамическое выделение памяти. Одномерные и двумерные динамические массивы. Динамические структуры данных. Однонаправленные и двунаправленные списки. Стеки и очереди
9.3	Динамическое программирование /Ср/	2	4	Л1.1Л2.1 Л2.2	Разделение памяти, динамическое выделение памяти. Одномерные и двумерные динамические массивы. Динамические структуры данных. Однонаправленные и двунаправленные списки. Стеки и очереди
9.4	/КСР/	2	1	Л1.1Л2.1 Л2.2	КСР

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

5.1. Типовые задания для проведения текущего контроля

Вариант контрольной работы.

Контрольная работа

1. Форматированный ввод/вывод данных.

2. Структура программы в С++. Препроцессорные директивы.

3. Условный оператор.

4. Вычислите объем шара ($\frac{4}{3}\pi R^3$), реализуя решение данной задачи с использованием константы. Напишите код программы.

5. Вычислите значение выражения $\sqrt{(a+b^2) / (\ln(c))}$ с учетом области определения функции. Изобразите блок-схему решения данной задачи и напишите код программы.

Вариант индивидуального задания.

1. Тело движется по закону $S = t^3 - 3t^2 + 2$. Вычислить скорость тела в момент времени t . Значение t ввести с клавиатуры (Функция скорости есть производная от функции расстояния по времени).

2. Дано пятизначное число. Определить: а) входит ли в него цифра 5; б) входит ли в него цифра а.

3. Написать программу поиска суммы последовательности положительных чисел, вводимых с клавиатуры, предшествующих первому введенному нулю. Контрольный пример: 1,2,3,-4,5,-2,0.

4. Даны два натуральных числа. Выяснить, в каком из них сумма цифр больше. (Определить функцию для расчета суммы цифр натурального числа.)

5. Найти n -ый член последовательности: $1/3, 2/5, 3/7, 4/9, \dots$

Примеры заданий из лабораторных работ:

1. Составьте программу, в которой запрашиваются радиус окружности основания и высота конуса и вычисляются характеристики: площадь полной поверхности, объем..
2. Дано целое число x . Составьте программу, которая выясняет, верно ли, что $0 \leq x < 1$. На экран вывести <<Да>> или <<Нет>>.
3. Даны действительные числа x, y, z . Составьте программу, которая находит наибольшее из них.
4. Мастям игральных карт присвоены следующие порядковые номера: пики — 1, трефы — 2, бубны — 3, червы — 4; а достоинству карт: валету — 11, даме — 12, королю — 13, тузу — 14 (порядковые номера остальных достоинств соответствуют их названию: шестерке — 6, семерке — 7 и т.д.) Составьте программу, которая по заданному номеру масти (от 1 до 4) и достоинству карты (от 6 до 14) определяет полное название (масть и достоинство) соответствующей карты, например, «Король бубен», «Десятка пик» и т.п.

5.2. Типовые задания для проведения промежуточной аттестации

Вопросы к зачету по дисциплине «Алгоритмизация и программирование» 1 семестр

1. Алгоритмы: определение, свойства, способы записи и виды.
2. Блок-схемы: способы представления, основные элементы. Примеры для разных алгоритмических конструкций.
3. Возникновение и развитие языка C. Стандарты языка. Язык C++.
4. Общие сведения о среде программирования. Отличительные особенности среды программирования Microsoft Visual Studio 2013.
5. Основные объекты языка: алфавит, лексемы, данных.
6. Константы, идентификаторы, переменные.
7. Типы данных. Множество значений и множество операций на типе данных. Операции замещения.
8. Правила записи программ на языке C. Структура программы на языке C.
9. Форматированный ввод/вывод данных различных типов. Спецификаторы. Примеры.
10. Стандартные потоки ввода-вывода данных. Манипуляторы потока. Примеры.
11. Комментарии. Управляющие последовательности. Примеры.
12. Оператор присваивания. Арифметические операции. Приоритеты выполнения операций. Примеры.
13. Логические операции: условные и побитовые. Операции отношений. Сдвиговые операции. Примеры.
14. Математические функции. Преобразование типов. Примеры.
15. Простой оператор. Составной оператор. Примеры.
16. Операторы выбора: условный оператор if. Примеры.
17. Операторы выбора: оператор-переключатель switch. Примеры.
18. Оператор цикла с предусловием. Примеры.
19. Оператор цикла с постусловием. Примеры.
20. Оператор цикла с параметром. Примеры.
21. Функции пользователя: описание, вызов. Локальные и глобальные переменные. Примеры.
22. Функция пользователя. Перегрузка функций. Заголовочные файлы. Примеры.
23. Рекурсия: понятие, рекурсивная триада. Примеры решения задач рекурсивным способом.
24. Указатели: указатели на объекты, указатели на функции, указатель на указатель.
25. Одномерные массивы: объявление, инициализация, генерация и вывод. Определения размера памяти для одномерного массива. Указатели и массивы.
26. Одномерные массивы: задачи поиска, замены и перестановок элементов массива.
27. Одномерные массивы: задачи сортировок элементов массива. Сортировка методом "пузырька" (простого обмена).
28. Одномерные массивы: задачи сортировок элементов массива. Сортировка методом простого выбора (простой перебор).
29. Одномерные массивы: задачи сортировок элементов массива. Сортировка методом простого включения (сдвиг-вставка, вставками, вставка и сдвиг).
30. Двумерные массивы: объявление, инициализация, генерация и вывод.
31. Двумерные массивы: задачи поиска, замены и суммирования элементов двумерного массива.
32. Двумерные массивы: задачи сортировок и перестановок в двумерных массивах.

5.3. Перечень видов оценочных средств

1. лабораторные работы;
2. контрольные работы;
3. индивидуальные задания;
4. зачет.

5.4. Процедура применения оценочных материалов

Промежуточная аттестация может проводиться с применением электронного обучения и (или) дистанционных образовательных технологий в соответствии с "Порядком проведения промежуточной аттестации с применением электронного обучения и /или дистанционных образовательных технологий".

Оценочные материалы представлены в Приложении файл

Описание балльно-рейтинговой системы по дисциплине "Алгоритмизация и программирование".

Составляющие итоговой оценки за дисциплину:

- 1) Текущий контроль (общий вес 70 баллов):
 - до 60 баллов - выполнение лабораторных работ, самостоятельная работа;
 - до 10 баллов - выполнение индивидуальных заданий и контрольной работы.

2) Промежуточная аттестация заключается в проведении экзамена (общий вес - 30 баллов): ответы на два вопроса в билете и выполнение практического задания.
При этом, для получения положительной итоговой оценки на экзамене необходимо получить не менее 50% по каждой составляющей и выполнить все практические задания.

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

6.1. Рекомендуемая литература

6.1.1. Основная литература

	Авторы, составители	Заглавие	Издательство, год (кол-во экземпляров для печатных изданий)	Ссылка на электронное издание
Л1.1	Трофимов В. В.	Алгоритмизация и программирование: учебник для академического бакалавриата	, 2017	https://www.biblio-online.ru/book/B08DB966-3F96-4B5A-B030-E3CD9085CED4

6.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год (кол-во экземпляров для печатных изданий)	Ссылка на электронное издание
Л2.1	Р. Ю. Царев [и др.]	Информатика и программирование: учебное пособие	, 2014	http://biblioclub.ru/index.php?page=book_red&id=364538&sr=1
Л2.2	Царев Р. Ю.	Программирование на языке Си: учебное пособие	, 2014	http://biblioclub.ru/index.php?page=book_red&id=364601&sr=1

6.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	Официальный сайт ФГБОУ ВО «Тульский государственный педагогический университет им. Л.Н. Толстого»
Э2	Среда электронного обучения LMS Moodle

6.3. Информационные технологии

6.3.1 Перечень лицензионного и свободно распространяемого программного обеспечения

1.	Операционная система Microsoft Windows XP Professional Russian. Лицензия № 16698685 от 08.08.2003 г.
2.	Операционная система Microsoft Windows Professional 7 Russian. Лицензия №48497058 от 13.05.2011 г., договор № Пр/16/6 от 05 апреля 2016 г.
3.	Операционная система Microsoft Windows 10 Professional Russian. Контракт № ПР/ФЕН/15/18 от 23.10.2015 г., договор № Пр/16/6 от 05 апреля 2016 г.
4.	Программное обеспечение Microsoft Office Enterprise 2007 Russian. Лицензия №46138962 от 16.11.2009
5.	Программное обеспечение Microsoft Office 2013 Professional. Контракт № 405535 от 2 ноября 2015 года, контракт № ПР/ФЕН/15/18 от 23.10.2015 г.
6.	Оболочка программирования Code: Blocks 17.12. Свободно распространяемое ПО
7.	Файловый архиватор 7z. Свободно распространяемое ПО
8.	Браузеры Google Chrome, Mozilla, Opera. Свободно распространяемое ПО

6.3.2 Перечень информационных справочных систем и профессиональных баз данных

1.	Компьютерная информационно-правовая система «Гарант»
----	--

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Ауд.	Назначение	Оборудование и технические средства обучения	Вид
4-316	Учебная аудитория	Учебная аудитория для проведения учебных занятий, оснащенная оборудованием и техническими средствами обучения, персональные компьютеры Asus Intel(R), переносной проектор Acer, комплект учебной мебели	Лек, Лаб, Ксп, зачет
4-305	Помещение для самостоятельной работы	Помещение для самостоятельной работы обучающихся, оснащенное компьютерной техникой, подключенной к сети Интернет, обеспечен доступ к электронно-образовательной среде Университета: комплект учебной мебели, персональные компьютеры (ноутбуки) с подключением к сети Интернет и обеспечением доступа к электронным библиотекам и в электронную информационно-образовательную среду Университета, доска, компьютер стационарный (моноблок)	Ср

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

Приступая к изучению новой учебной дисциплины, обучающиеся должны ознакомиться с рабочей программой, учебной, научной и методической литературой, осуществить запись на соответствующий курс в среде электронного обучения университета.

Глубина усвоения дисциплины зависит от активной и систематической работы обучающегося на лекциях и лабораторных занятиях, а также в ходе самостоятельной работы, по изучению рекомендованной литературы.

На лекциях важно сосредоточить внимание на ее содержании. Это поможет лучше воспринимать учебный материал и уяснить взаимосвязь проблем по всей дисциплине. Основное содержание лекции целесообразнее записывать в тетради в виде ключевых фраз, понятий, тезисов, обобщений, схем, опорных выводов. Необходимо обращать внимание на термины, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации. Желательно оставлять в конспектах поля, на которых делать пометки из рекомендованной литературы, дополняющей материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений. С целью уяснения теоретических положений, разрешения спорных ситуаций необходимо задавать преподавателю уточняющие вопросы. Для закрепления содержания лекции в памяти, необходимо во время самостоятельной работы внимательно прочесть свой конспект и дополнить его записями из учебников и рекомендованной литературы. Конспектирование читаемых лекций и их последующая доработка способствует более глубокому усвоению знаний, и поэтому являются важной формой учебной деятельности обучающихся.

Прочное усвоение и долговременное закрепление учебного материала невозможно без продуманной самостоятельной работы. Такая работа требует от обучающегося значительных усилий, творчества и высокой организованности. В ходе самостоятельной работы обучающиеся выполняют следующие задачи: дорабатывают лекции, изучают рекомендованную литературу, готовятся к лабораторным работам, к тестированию по отдельным темам дисциплины. При этом эффективность учебной деятельности обучающегося во многом зависит от того, как он распорядился выделенным для самостоятельной работы бюджетом времени.

Результатом самостоятельной работы является прочное усвоение теоретического материала по предмету согласно программы дисциплины. В итоге этой работы формируются профессиональные умения и компетенции, развивается творческий подход к решению возникших в ходе учебной деятельности проблемных задач, появляется самостоятельность мышления.

Целью лабораторных занятий по данной дисциплине является закрепление теоретических знаний, полученных при изучении дисциплины. При подготовке к лабораторной работе целесообразно выполнить следующие рекомендации: изучить основную литературу; ознакомиться с дополнительной литературой, новыми публикациями в периодических изданиях: журналах, рекомендуемыми электронными ресурсами и т. д.; при необходимости доработать конспект лекций. При этом учесть рекомендации преподавателя и требования рабочей программы.

При выполнении лабораторных заданий основным методом обучения является самостоятельная работа обучающегося под управлением преподавателя. На них пополняются теоретические знания обучающихся, их умение творчески мыслить, анализировать, обобщать изученный материал, проверяется отношение обучающихся к будущей профессиональной деятельности.

Оценка выполненной работы осуществляется преподавателем комплексно: по результатам выполнения заданий, устному сообщению и оформлению индивидуального проекта. После подведения итогов занятия обучающийся обязан устранить недостатки, отмеченные преподавателем при оценке его работы.