



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.О.02.09 Объектно-ориентированные языки и системы программирования

Направление подготовки: 01.04.02 Прикладная математика и информатика

Направленность (профиль): Информационное обеспечение экономической деятельности

Квалификация (степень): магистр

Форма обучения: очная

Институт: математики, естествознания и техники

Кафедра: математического моделирования, компьютерных технологий и информационной безопасности

	очная форма	очно- заочная форма	заочная форма
Курс	1		
Семестр	1		

Лекции	18		
Лабораторные занятия	18		
Практические (семинарские) занятия	18		
в т. ч. практическая подготовка	-		
Форма(ы) промежуточной аттестации	Экзамен - 0,3		
Контроль	9		
Иные формы работы	-		
Самостоятельная работа	116.7		

Всего часов: 180

Трудоемкость: 5 зачетных единиц.

Разработчик(и) рабочей программы:

кандидат физико-математических наук, доцент Д.В. Корниенко

I. ОРГАНИЗАЦИОННО-МЕТОДИЧЕСКИЙ РАЗДЕЛ

Цель изучения дисциплины:

- овладение знаниями и навыками программирования с использованием современных языков программирования;
- формирование компьютерной грамотности и подготовка студентов к использованию современных компьютеров и технологий программирования в качестве инструмента для решения практических задач в своей предметной области;
- формированием представлений о роли программирования в управлении и специфике информационных систем;
- формированием информационной культуры, подготовка будущих выпускников к практической деятельности в условиях широкого использования информационных технологий.

Задачи изучения дисциплины:

- овладение приемами работы в современных средах разработки программного обеспечения, обеспечивающих широкие возможности написания, редактирования и компилирования программ;
- обучение принципам построения программного кода;
- формирование навыков программирования на языке высокого уровня, которые будут использоваться и углубляться при выполнении различных заданий и работ по дисциплинам, изучаемым на последующих курсах;
- развитие навыков системного мышления;
- систематизация и расширение знаний приемов и методов работы с информационно-коммуникационными технологиями, подготовка к их осознанному использованию при решении различного вида прикладных задач.

Место дисциплины в структуре ОПОП: реализуется в рамках обязательной блока Б1. Дисциплины (модули).

Планируемые результаты обучения по дисциплине:

Код компетенции	Индикаторы достижения компетенции	Планируемые результаты обучения по дисциплине
УК-4	Знать: <ul style="list-style-type: none">- компьютерные технологии и информационная инфраструктура в организации;- коммуникации в профессиональной этике;- методы исследования коммуникативного потенциала личности;- современные средства информационно-коммуникационных технологий.	Знает: <ul style="list-style-type: none">– современные средства информационно-коммуникационных технологий;– компьютерные технологии и информационная инфраструктура в организации;

	<p>Уметь:</p> <ul style="list-style-type: none"> - создавать на русском и иностранном языке письменные тексты научного и официально- делового стилей речи по профессиональным вопросам; - исследовать прохождение информации по управленческим коммуникациям; - производить редакторскую и корректорскую правку текстов научного и официально- делового стилей речи на русском и иностранном языке; - анализировать систему коммуникационных связей в организации; - представлять результаты академической и профессиональной деятельности на различных научных мероприятиях, включая международные. 	<p>Умеет:</p> <ul style="list-style-type: none"> - анализировать систему коммуникационных связей в организации; - представлять результаты академической и профессиональной деятельности на различных научных мероприятиях, включая международные.
	<p>Владеть:</p> <ul style="list-style-type: none"> - интегративными умениями, необходимыми для эффективного участия в академических и профессиональных дискуссиях; - использованием современных средств информационно- коммуникационных технологий. 	<p>Владеет:</p> <ul style="list-style-type: none"> - использованием современных средств информационно- коммуникационных технологий.
<p>ОПК-4</p>	<p>Знать:</p> <ul style="list-style-type: none"> - языки программирования, языки баз данных, алгоритмы, библиотеки и пакеты программ; 	<p>Знает:</p> <ul style="list-style-type: none"> - методологии разработки программного обеспечения и технологии программирования; - языки формализации функциональных спецификаций - методы и приемы формализации задач; - методы и средства проектирования программного обеспечения, программных интерфейсов и баз данных; - принципы построения и виды архитектуры программного обеспечения; - типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения;

	<p>Уметь:</p> <ul style="list-style-type: none"> - применять в профессиональной деятельности современные информационно-коммуникационные технологии, современные системы программирования, операционные системы, продукты системного и прикладного программного обеспечения, сетевые технологии с учетом требований информационной безопасности; 	<p>Умеет:</p> <ul style="list-style-type: none"> – использовать существующие типовые решения и шаблоны проектирования программного обеспечения; – применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов; – использовать современные системы программирования; – выбирать средства реализации требований к программному обеспечению; – вырабатывать варианты реализации программного обеспечения и требований к нему;
	<p>Владеть:</p> <ul style="list-style-type: none"> - способностью реализовывать решения научно-исследовательских задач с применением современных информационно-коммуникационных технологий с учетом требований информационной безопасности 	<p>Владеет</p> <ul style="list-style-type: none"> – методологией и технологиями проектирования программного обеспечения, программных интерфейсов, структур и баз данных в соответствии с установленными требованиями; – действиями по разработке и согласованию технических спецификаций на программные компоненты; – действиями по согласованию требований к программному обеспечению с заинтересованными сторонами, распределению заданий между программистами в соответствии с техническими спецификациями, осуществлению контроля выполнения заданий, формированию отчетности в соответствии с установленными регламентами.

II. СОДЕРЖАНИЕ И ОБЪЕМ ДИСЦИПЛИНЫ

с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу

Очная форма обучения

№ п/п	Наименование разделов и тем	Всего	Аудиторные занятия			Сам. раб.
			ЛК	ПЗ	ЛБ	
1 семестр						
	Раздел 1. Объектно-ориентрованное программирование	68	9	9	9	41
1	Тема 1. Указатели (Объявления и инициализация переменной-указателя. Операции над указателями. Пере-	5	1	1	1	2

	дача параметра по ссылке. Связь между указателями и массивами. Массивы указателей. Указатели на функции)					
2	Тема 2. Структуры, объединения и перечисления (Описание структур. Инициализация структур. Доступ к элементам структур. Использование структур с функциями. Объединения. Структуры, ссылающиеся на себя. Динамическое распределение памяти. Связанные списки. Стеки. Очереди. Деревья)	5	1	1	1	2
3	Тема 3. Понятия объектно-ориентированного программирования	4	1	1	1	1
4	Тема 4. Классы и абстракция данных (Создание класса. Область действия класса и доступ к элементам класса. Функции доступа и сервисные функции. Инициализация объектов класса: конструкторы. Использование с конструкторами аргументов по умолчанию. Деструкторы. Использование элементов данных и элементов-функций)	9	1	1	1	6
5	Тема 5. Перегрузка операций (Основные принципы перегрузки операций. Запреты на перегрузку операций)	9	1	1	1	6
6	Тема 6. Наследование (Базовые и производные классы. Защищенные элементы. Переопределение элементов базового класса в производном классе. Открытые, защищенные и закрытые базовые классы. Непосредственные и косвенные базовые классы)	9	1	1	1	6
7	Тема 7. Виртуальные функции и полиморфизм (Виртуальные функции. Абстрактные базовые классы и конкретные классы. Полиморфизм. Новые классы и динамическое связывание)	9	1	1	1	6
8	Тема 8. Работа с файлами (Файлы и потоки. Создание файла последовательного доступа. Чтение данных из файла последовательного доступа. Файлы произвольного доступа. Создание файла произвольного доступа. Произвольная запись данных в файл произвольного доступа. Последовательное чтение данных из файла произвольного доступа)	9	1	1	1	6
9	Тема 9. Символы и строки (Библиотека обработки символов. Функции преобразования строк. Функции стандартной библиотеки ввода/вывода)	9	1	1	1	6
Раздел 2. Программирование в системе 1С:Предприятие 8		102,7	9	9	9	75,7

10	Тема 10. Основные объекты конфигурации (назначение, свойства, использование).	8,7	1	1	1	5,7
11	Тема 11. Работа с документами.	13	1	1	1	10
12	Тема 12. Работа с регистрами.	13	1	1	1	10
13	Тема 13. Дополнительные возможности платформы.	8	1	1	1	5
14	Тема 14. Особенности расчета заработной платы в системе 1С:Предприятие 8. Расчетные объекты конфигурации.	13	1	1	1	10
15	Тема 15. Базовая зависимость. Зависимость по периоду действия.	13	1	1	1	10
16	Тема 16. Механизм Сторно.	13	1	1	1	10
17	Тема 17. Получение данных о фактическом периоде действия записи для расчета. Получение данных графика для расчета записи.	13	1	1	1	10
18	Тема 18. Получение базы для расчета записей. Перерасчеты.	8	1	1	1	5
	<i>Контроль</i>	9				
	<i>Экзамен</i>	0,3				
	<i>Итого за 1 семестр</i>	180	18	18	18	116.7
	в т.ч. практическая подготовка	-				
	ИТОГО:	180	18	18	18	116.7

Очно-заочная форма обучения не реализуется

Заочная форма обучения не реализуется

III. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Текущая аттестация проводится в форме контрольной работы, теста, реферата.

Типовой вариант контрольной работы

Контрольная работа (5 вариантов)

1. Класс и объект. Назначение элементов и методов класса.
2. Конструкторы и деструктор класса.
3. Дружественные функции и классы.
4. Наследование класса. Пример простого наследования класса.
5. Множественное наследование. Базовый и производный классы.

Типовой вариант тестов (1 вариант)

С1. Какие методы компилятор может автоматически скомпилировать для класса:

```
class A
{
};
```

- a) конструктор по умолчанию
деструктор
- b) конструктор копирования оператор - operator= (const A&)
деструктор
- c) конструктор по умолчанию
конструктор копирования оператор - operator= (const A&) деструктор
- d) конструктор по умолчанию
конструктор копирования

C2. Как можно вызвать метод следующего класса:

```
class Child
{
public:
    void Count(){ }
};
```

- a) Child * obj = new Child;
obj->Count;
- b) Child obj = new Child;
obj.Count();
- c) Child * obj = new Child;
obj.Count();
- d) Child obj = new Child;
(*obj).Count();

C3. Какое из перечисленных утверждений не верно по отношению к друзьям класса:

- a) Другом класса может быть объявлена любая функция;
- b) Для объявления класса В другом класса А достаточно внутри описания класса А добавить инструкцию : friend class B;
- c) Внутри описания класса может быть любое количество инструкций friend;
- d) Другом класса может быть объявлен любой другой класс;
- e) Для объявления глобальной функции f другом класса А достаточно внутри описания класса А добавить инструкцию: friend f;

C4. Допустимо ли в C++ определение следующего чисто виртуального метода:

```
class Abstract
{
public:
    virtual voidPure ()= 0
    {
    }
};
```

- a) Да, метод доступен по имени PureVirtual();
- b) Да, метод доступен по имени Abstract: PureVirtual();
- c) Да, однако метод не доступен для вызова;
- d) Нет, определение не допустимо.

C5. Что напечатает следующий код:

```
#include "stdio.h"
```

```

class super1
{
public:
    virtual void cout()
    {
        printf("%d",1);
    }
}

class super2
{
public:
    virtual void cout()
    {
        printf("%d",2);
    }
}

class Child:public super1,public super2
{
public:
    void cout()
    {
        printf("%d",3);
    }
};

int main()
{
    super1 * obj = new Child;
    obj-> count();
    return 0;
}

```

- a) 1;
- b) 2;
- c) 3;
- d) Результат не определен;
- e) Возникнет ошибка компиляции.

С6. Что напечатает код при создании экземпляра класса X:

```

class Y
{
public:
    Y() {cout<<"Y";}
};

class Z
{

```

```

public:
    Z() {cout<<"Z";}
};
class X:public Z
{
private:
    Y m_objY;
public:
    X() {cout<<"X";}
};

```

- a) XY;
- b) YZ;
- c) XYZ;
- d) YXZ;
- e) ZYX.

C7. Что не верно в следующем коде:

```

class Parent
{
public:
    ~Parent(){}
    virtual void method() {}
};

class Child : public Parent
{
public:
    Child() { /*Захват ресурсов*/ }
    ~Child() { /*Освобождение ресурсов*/ }
    void method() { /*программный код*/ }
};

main()
{
    Parent *obj=new Child;
    delete obj;
    return 0;
}

```

- a) Код полностью корректен;
- b) Деструктор базового класса необходимо объявить как virtual;
- c) Код по очистке ресурсов необходимо перенести в базовый класс;
- d) Метод method в базовом классе не нужно объявлять как virtual;
- e) Деструктор и метод класса- наследника необходимо объявить как virtual.

C8. Что напечатает следующая программа:

```

#include <iostream>
using namespace std;
class A
{

```

```

    int I;
public:
    A()
    {
        cout<< "in A::A()" << endl;
        I = 0;
    }
    A operator = (A a)
    {
        cout << "in A::operator=(A)"<<endl;
        I = a.i;
    }
};

int main()
{
    A a;
    A b=a;
    return 0;
}

```

- a) In A::A();
- b) In A::operator=(A);
- c) In A::A()
In A::operator=(A);
- d) Возникнет ошибка компиляции;
- e) Возникнет ошибка времени выполнения.

С9. Что напечатает следующий код:

```

#include <iostream>
using namespace std;
class B
{
public:
    B(char c)
    {
        cout <<c;
    };
};

class D
{
public:
    B bB, bA, bC;

    D(void):bA('a'),bB('b'),bC('c');
};

int main()
{
    D dInstance;
}

```

```
    ...  
    return 0;  
}
```

- a) abc;
- b) cba;
- c) bac;
- d) Вывод зависит от версии компилятора

C10. Какие из следующих прототипов функций записаны некорректно:

```
template <class T> void func1();  
template <typename T> void func2(T);  
template <unsigned N> void func3();  
template <unsigned N> void func4(N);  
template <class T> void func5(typename T::f);
```

- a) 1
- b) 2
- c) 3
- d) 4
- e) 5

Примерная тематика рефератов

1. Распределенные и параллельные методы программирования
2. Современные методы Web-ориентированного программирования на языке C#
3. Web-технологии в разработке удаленных баз данных
4. Применение case-технологий к проектированию базы данных
5. Программные методы коррекции изображения
6. Способы обработки изображений с использованием библиотеки компьютерного зрения OpenCV
7. Построение многомодульных приложений
8. Разработка Web-приложений с применением технологии ASP.NET
9. Разработка клиент-серверного приложения средствами языка C#

Промежуточная аттестация обучающихся осуществляется в форме экзамена с использованием следующих оценочных материалов:

Вопросы к экзамену (1 семестр, очная форма обучения)

1. Создание класса. Область действия класса и доступ к элементам класса. Функции доступа и сервисные функции.
2. Инициализация объектов класса: конструкторы. Использование с конструкторами аргументов по умолчанию. Деструкторы.
3. Основные принципы перегрузки операций. Запреты на перегрузку операций.
4. Базовые и производные классы. Переопределение элементов базового класса в производном классе.

5. Виртуальные функции.
6. Полиморфизм.
7. Стек. Очередь. Дерево.
8. Особенности расчета заработной платы в системе 1С:Предприятие 8. Расчетные объекты конфигурации.
9. Регистр расчета и его применение.
10. План видов расчета и его применение.
11. Базовая зависимость.
12. Зависимость по периоду действия.
13. Внедрение расчетных механизмов в прикладное решение.
14. Механизм Сторно.
15. Применение запросов при решении расчетных задач. Получение данных из регистров расчета.
16. Получение данных о фактическом периоде действия записи для расчета.
17. Получение данных графика для расчета записи.
18. Получение базы для расчета записей.
19. Перерасчеты.

Задачи к экзамену (1 семестр, очная форма обучения)

- 1.** Создайте класс TIME (время), который содержит функции позволяющие установить часы, минуты и секунды (SET функции), а также функции возвращающие значения часов, минут и секунд (GET функции). Объекты этого класса должны выводиться в военном (23:45:32) и стандартном (11:45:32) форматах.
- 2.** Создайте класс DATE (дата), который содержит функции позволяющие установить год, месяц и день (SET функции), а также функции возвращающие значения года, месяца и дня (GET функции). Объекты этого класса должны выводить дату.
- 3.** Создайте класс EMPLOYEE (служащий), который содержит в себе объекты класса DATE (дата). Один объект класса DATE символизирует дату рождения служащего, второй объект класса DATE символизирует дату поступления на работу. Объекты класса EMPLOYEE выводят имя, фамилию, дату рождения и дату поступления на работу служащего.
- 4.** Создайте класс PhoneNumber (телефонный номер), закрытые элементы которого есть код города и номер. Также создайте дружественные функции, которые перегружают операции передачи и извлечения из потока и позволяющие обрабатывать данные класса телефонного номера.
- 5.** Создайте класс Point (точка), содержащий координаты точки. Создайте класс Circle (окружность) производный от класса Point, содержащий элемент данных радиус. Программа должна выводить объект класса Circle в виде - Центр = [x; y]; Радиус = r .

6. Создайте класс Shape (форма). Создайте от него производные классы Circle (окружность) и Cylinder(цилиндр). Класс Shape должен содержать чисто виртуальную функцию printShapeName (печатать имя формы). Эта функция должна переопределяться в каждом производном классе. Имена объектов производных классов должны выводиться на печать через указатель на базовый класс, то есть использовать свойства полиморфизма.

7. Создайте класс стек, который вставляет переменную в стек и выталкивает ее.

8. Создайте класс сотовый телефон. Создайте класс прилавок. Причем класс прилавок должен содержать не менее 3 объектов класса сотовый телефон. При описании классов используйте следующую информацию. Класс сотовый телефон должен содержать следующую информацию о сотовом телефоне:

- 1) наименование фирмы производителя
- 2) количество часов в режиме разговора
- 3) количество часов в режиме ожидания
- 4) число 1 если цветной дисплей и 0 если не цветной дисплей
- 5) цена

Класс прилавок должен содержать следующую информацию:

- 1) порядковый номер прилавка
- 2) количество сотовых телефонов на прилавке
- 3) информацию об имеющихся телефонах

10. Создайте класс группа. Создайте класс факультет. Причем класс факультет должен содержать не менее 3 объектов класса группа. При описании классов используйте следующую информацию. Класс группа должен содержать следующую информацию о группе:

- 1) наименование группы (к примеру, ПМ-11, ПМ-21)
- 2) количество человек в группе
- 3) фамилию старосты

Класс факультет должен содержать следующую информацию:

- 1) название факультета
- 2) количество групп
- 3) информацию о каждой группе

11. Создайте класс Student (Студент). Класс должен содержать в закрытых элементах-данных переменную, которая отвечает за фамилию студента, переменную, которая отвечает за имя студента, переменную, которая отвечает за название группы в которой обучается студент и переменную в которой хранится возраст студента. В открытом интерфейсе класса должны присутствовать функция, которая устанавливает значения для студента и функция, которая печатает информацию о студенте.

12. Создайте класс Преподаватель. Создайте класс Факультет. Причем класс Факультет должен содержать объекты класса Преподаватель. При описании классов используйте следующую информацию. Класс Преподаватель должен содержать следующую информацию о преподавателе:

- 1) Фамилию
- 2) Имя
- 3) Ученую степень

4) Звание

5) Должность

Класс Факультет должен содержать следующую информацию о факультете:

1) Название факультета

2) Декан факультета

3) Преподаватели факультета (2 человека).

13. Создайте класс Многочлен (3 степени). В классе должны присутствовать следующие функции: инициализация многочлена, вывод на экран многочлена, перегруженная операция «+».

14. Создайте класс Комплексное Число. В классе должны присутствовать функции, которые перегружают следующие операции: «+», «-», «*».

15. Создайте класс Матрица и реализуйте в нем функции, отвечающие за сложение матриц, умножения матрицы на число.

IV. ПЕРЕЧЕНЬ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

4.1. Основная литература

1. Гунько, А.В. Программирование : учебно-методическое пособие : [16+] / А.В. Гунько ; Новосибирский государственный технический университет. – Новосибирск : Новосибирский государственный технический университет, 2019. – 74 с. : табл. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book_view_red&book_id=576267 (дата обращения: 01.09.2022). – Библиогр. в кн. – ISBN 978-5-7782-3961-6. – Текст : электронный.
2. Горелов, С.В. Современные технологии программирования: разработка Windows-приложений на языке C#: учебник для студентов, обучающихся по дисциплине «Современные технологии программирования», направление «Прикладная информатика» (09.03.03 — для бакалавров, 09.04.03 — для магистров) : в 2 томах : [16+] / С.В. Горелов ; под науч. ред. П.Б. Лукьянова ; Финансовый университет при Правительстве Российской Федерации. – Москва : Прометей, 2019. – Том 1. – 363 с. : ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book_view_red&book_id=576037 (дата обращения: 01.09.2022). – Библиогр. в кн. – ISBN 978-5-907100-09-1. – Текст : электронный.
3. Скороход, С.В. Программирование на платформе 1С: предприятие 8.3 : [16+] / С.В. Скороход ; Южный федеральный университет. – Ростов-на-Дону ; Таганрог : Южный федеральный университет, 2019. – 136 с. : ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book_view_red&book_id=577921 (дата обращения: 21.07.2022). – Библиогр.: с. 132. – ISBN 978-5-9275-3315-2. – Текст : электронный.

4.2. Дополнительная литература

1. Практикум по администрированию программного обеспечения: лабораторный практикум : [16+] / авт.-сост. И.В. Анзин ; Северо-Кавказский федеральный университет. – Ставрополь : Северо-Кавказский Федеральный университет (СКФУ), 2017. – 85 с. : ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book_view_red&book_id=483755 (дата обращения: 01.09.2022). – Библиогр. в кн. – Текст : электронный.
2. Марченко, И.О. Разработка системы управления предприятием на платформе «1С: Предприятие 8.3» : учебно-методическое пособие : [16+] / И.О. Марченко, М.Л. Перевертайло ; Новосибирский государственный технический университет. – Новосибирск : Новосибирский государственный технический университет, 2018. – 116 с. : ил., табл. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book_view_red&book_id=574864 (дата обращения: 12.07.2022). – Библиогр. в кн. – ISBN 978-5-7782-3714-8. – Текст : электронный

V. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

№ пп	Ссылка на информационный ресурс	Наименование разработки в электронной форме	Доступность
1.	http://edu.ru/	Российское образование: Федеральный портал. Включает ссылки на порталы и сайты образовательных учреждений; государственные образовательные стандарты; нормативные документы; каталог экскурсий и обучающих программ.	Свободный доступ
2.	http://citforum.ru/database/osbd/contents.shtml	Информационно-аналитические материалы	Свободный доступ
3.	https://its.1c.ru/	Информационно-технологическое сопровождение пользователей 1С	Доступ по регистрации

VI. СОВРЕМЕННЫЕ ПРОФЕССИОНАЛЬНЫЕ БАЗЫ ДАННЫХ И ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ СИСТЕМЫ

1.	http://www.biblioclub.ru	Электронно-библиотечная система (ЭБС) Университетская библиотека онлайн	Регистрация через любой университетский компьютер.
----	---	--	--

			В дальнейшем предоставляется неограниченный индивидуальный доступ из любой точки, в которой имеется доступ к сети Интернет
2.	www.garant.ru	Информационно-правовой портал	Свободный доступ
3.	www.elibrary.ru	Российский информационный портал в области науки, технологии, медицины и образования	Свободный доступ
4.	www.consultant.ru	Российская компьютерная справочно-правовая система	Свободный доступ

VII. ЛИЦЕНЗИОННОЕ И СВОБОДНО РАСПРОСТРАНЯЕМОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

При реализации учебной дисциплины применяется следующее лицензионное и свободно распространяемое программное обеспечение:

- Microsoft Windows;
- Microsoft Office;
- LibreOffice;
- и др.

VIII. ОБОРУДОВАНИЕ И ТЕХНИЧЕСКИЕ СРЕДСТВА ОБУЧЕНИЯ, НЕОБХОДИМЫЕ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Учебные занятия проводятся в аудиториях, укомплектованных специализированной мебелью, в том числе стационарными или переносными техническими средствами обучения (проектор, экран, компьютер/ноутбук).

Лабораторные занятия, групповые и индивидуальные консультации, текущая и промежуточная аттестации проводятся в специализированных классах, оснащенных автоматизированными рабочими местами с компьютерами.

Самостоятельная работа проводится в кабинетах, оснащенных компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.