



Министерство науки и высшего образования Российской Федерации  
Федеральное Государственное Бюджетное Образовательное  
Учреждение Высшего Образования  
«Ингушский Государственный Университет»

Принята  
решением Ученого совета ИнгГУ

Утверждаю  
И.о. проректора по УР \_\_\_\_\_ Ф.Д. Кодзоева

от «30» июня 2022г.  
Протокол №10

«30» июня 2022г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ  
(Модуля)**

**Б1.В.ДВ.01.01. Языки программирования**

Направление подготовки

**09.03.02 Информационные системы и технологии**

Направленность (*профиль подготовки*)

**Информационные системы и технологии**

Квалификация выпускника

**бакалавр**

Форма обучения

**очная**

## 1. Цели освоения дисциплины

Целями освоения дисциплины «Языки программирования» является— формирование теоретических знаний и практических навыков по использованию современных электронно-вычислительных и программных средств для решения широкого спектра задач в различных областях а именно:

- знакомство студентов с основными видами языков программирования высокого уровня;
- формирование навыков работы с различными средствами программирования и отладки для создания программного обеспечения на языках высокого уровня;
- обучение основным принципам алгоритмического подхода, от этапа формализации до реализации в виде программного кода.

Формируемые дисциплиной знания и умения готовят выпускника данной образовательной программы к выполнению следующих обобщенных трудовых функций (трудовых функций):

Код и наименование профессионального стандарта	Обобщенные трудовые функции			Трудовые функции		
	Код	Наименование	Уровень квалификации	Наименование	Код	Уровень (подуровень) квалификации
06.001 Программист	D	Разработка требований и проектирование программного обеспечения	6	Анализ требований к программному обеспечению	D/01.6	6
				Разработка технических спецификаций на программные компоненты и их взаимодействие	D/02.6	6
				Проектирование программного обеспечения	D/03.6	6

## 2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина «Языки программирования» изучается в блоке Б1.В и является одной из дисциплин вариативной части междисциплинарного профессионального модуля., формируемой участниками образовательных отношений и имеет соответствующий шифр Б1.В.ДВ.01.01 подготовки бакалавриата по направлению 09.03.02 «Информационные системы и технологии». Дисциплина осваивается на 2 курсе в 4 семестра.

Дисциплины и практики, знания и умения по которым необходимы как "входные" при изучении данной дисциплины.	«Математика», «Информатика», «Информационные технологии»
Дисциплины, практики, ГИА, для которых изучение данной дисциплины необходимо как «предшествующее».	Инфокоммуникационные системы и сети», «Технологии обработки информации», «Методы и средства проектирования информационных систем и технологий», «Инструментальные средства информационных систем», «Интеллектуальные системы и технологии»,

Формы работы студентов - в ходе изучения дисциплины предусмотрены семинарские занятия, выполнение домашних работ. Самостоятельная работа студентов, предусмотренная учебным планом, выполняется в ходе семестра в форме выполнения домашних заданий. Отдельные темы теоретического курса прорабатываются студентами самостоятельно в соответствии с планом самостоятельной работы и конкретными заданиями преподавателя с учетом индивидуальных особенностей студентов. Виды текущего контроля - проверка домашних заданий, устный опрос, проверка контрольной работы. Форма итогового контроля – экзамен.

### 3. Результаты освоения дисциплины, Языки программирования

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО по данному направлению:

Код компетенции	Наименование компетенции	Индикатор достижения компетенции	В результате освоения дисциплины обучающийся <b>должен</b> :
УК-9	Способен принимать обоснованные экономические	УК-9.1.Понимает базовые принципы функционирования экономики	Понимать: базовые принципы функционирования экономики и экономического развития, цели и формы участия государства в

	решения различных областях жизнедеятельности.	экономического развития, цели и формы участия государства в экономике. УК-9.2. Применяет методы личного экономического и финансового планирования для достижения текущих и долгосрочных финансовых целей, использует финансовые инструменты для управления личными финансами (личным бюджетом), контролирует собственные экономические и финансовые рынки.	экономике. Применять: методы личного экономического и финансового планирования для достижения текущих и долгосрочных финансовых целей, использует финансовые инструменты для управления личными финансами (личным бюджетом), контролирует собственные экономические и финансовые рынки.
ОПК-6	Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий.	ОПК-6.1. Методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий ОПК-6.2. Применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий; ОПК-6.3. Навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач	Знать: методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий.  Уметь: применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий;  Владеть навыками: программирования, отладки и тестирования прототипов программно-технических комплексов задач.
ПК-3	Способен выполнять интеграцию программных модулей и компонент	ПК-3.1. Знает методы и средства сборки модулей и компонент программного обеспечения; интерфейсы взаимодействия с внешней средой; интерфейсы взаимодействия внутренних модулей системы; методы и средства разработки	Знать методы и средства сборки модулей и компонент программного обеспечения; интерфейсы взаимодействия с внешней средой; интерфейсы взаимодействия внутренних модулей системы; методы и средства разработки процедур для развертывания программного обеспечения; языки, утилиты и средства программирования, средства

		<p>процедур для развертывания программного обеспечения; языки, утилиты и среды программирования, средства пакетного выполнения процедур;</p> <p>ПК-3.2. Умеет писать программный код процедур интеграции программных модулей; использовать выбранную среду программирования для разработки процедур интеграции программных модулей; применять методы и средства сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов;</p> <p>ПК-3.3.: Имеет навыки разработки процедур сборки модулей и компонент программного обеспечения; разработки процедур развертывания и обновления программного обеспечения; разработки процедур миграции и преобразования конвертации</p>	<p>пакетного выполнения процедур;</p> <p><u>Уметь</u> писать программный код процедур интеграции программных модулей; использовать выбранную среду программирования для разработки процедур интеграции программных модулей; применять методы и средства сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов;</p> <p><u>Иметь</u> навыки разработки процедур сборки модулей и компонент программного обеспечения; разработки процедур развертывания и обновления программного обеспечения; разработки процедур миграции и преобразования конвертации</p>
--	--	---	---

#### 4. Структура и содержание дисциплины Языки программирования

##### Структура дисциплины

Общая трудоемкость дисциплины составляет 5 зачетных единиц, 180 часов.

№ п/п	Наименование разделов и тем дисциплины (модуля)	семестр	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в		Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)
			Контактная работа	Самостоятельная работа	









<p>6.13. Память. Размещение переменных в памяти. Адреса, указатели. Получение адреса (&amp;) и разыменование (*). Операции над указателями (арифметика, сравнение). Нетипизированные указатели. Расположение элементов массива в памяти. Имя массива как адрес первого элемента. Операция []. Особенности работы с многомерными массивами. Передача аргументов в функцию. Статическое и динамическое выделение памяти. Динамические массивы. Функции malloc, calloc, realloc, free. Основные ошибки: неинициализированные указатели, висячие ссылки, утечки памяти. Три способа создания многомерных динамических массивов — адресация в одномерном по формуле, адресация в одномерном при фиксированной</p>	4	2	2																	
<p>6.14. Вложенные циклы. Работа с таблицами и матрицами</p>	4	8		8	8				8											
<p><b>Раздел 7. Классы памяти. Рекурсия</b></p>	4																			

<p>7.15.Существование переменных в пространстве и времени. Спецификаторы классов памяти (auto, register, static, extern). Области памяти: регистры процессора, сегмент стека, сегмент данных, динамическая область памяти (куча). Прямая и косвенная рекурсия. Дублирование всех локальных переменных. Пример: вычисление факториала. Хвостовая рекурсия, эквивалентность циклу. Указатели на функции. Определение переменных и типов. Области применения: передача в качестве параметра, массивы указателей на функцию.</p>	4	2	2											
<p><b>Раздел 8. Препроцессор языка С++</b></p>	4													
<p>8.16.Макропроцессоры и макрогенераторы. Предобработка (препроцессинг) исходных текстов. Стадии препроцессорной обработки: объединение строк, удаление комментариев, распознавание директив и лексем, выполнение директив. Директивы #include, #define, #undef. Макроподстановки с параметрами. Цепочка подстановок. Препроцессорные операции: кватирование («преобразование в строку»), склейка препроцессорных лексем. Некоторые predefined макросы. Условное включение текста (#ifdef, #ifndef, #if, #elif, #else, #endif). Препроцессорная</p>	4	2	2											

	<b>Раздел 9. Абстрактные типы данных. Примеры составных типов</b>	4																			
	9.17.Определение АД. Структуры. Определение структур, теги и имена, даваемые typedef. Инициализация, доступ к элементам. Наличие присваивания в отличие от массивов. Отсутствие сравнения. Указатели и доступ к полям через указатель. Неоднозначность расположения в памяти. Объединения и перечисления. Особенности расположения в памяти и инициализации. Доступ к полям. Endianness. Битовые поля в структурах и объединениях, зависимость расположения от endian.	4	4	4																	
	9.18.Структуры данных	4	8		8	8			8												
	9.19.Определение АД. Структуры. Определение структур, теги и имена, даваемые typedef. Инициализация, доступ к элементам. Наличие присваивания в отличие от массивов. Отсутствие сравнения. Указатели и доступ к полям через указатель. Неоднозначность расположения в памяти. Объединения и перечисления. Особенности расположения в памяти и инициализации. Доступ к полям. Endianness. Битовые поля в структурах и объединениях, зависимость расположения от endian.	4				6			6												

	<b>Раздел 10. Файловый ввод-вывод. Функции с переменным числом аргументов</b>	4																		
	<p>10.20.Обработка файлов. Файлы как абстракции, связь потоков с файлами (fopen, freopen, fclose). Разница между текстовыми и бинарными файлами. Функции для работы с файлами (getc, fgetc, getchar; putc, fputc, putchar; ungetc; fgets, gets, fputs, puts; fprintf, fscanf; fwrite, fread). Формат бинарных файлов: свобода в интерпретации последовательностей байтов. Традиционные подходы: сырой («raw») формат, однородные файлы записей, блоковая («chunk») структура с заголовками, каталог. Функции для произвольного доступа (ftell, fseek, fgetpos, fsetpos, rewind). Ошибки ввода-вывода и чтение после конца файла, индикация с помощью возвращаемого значения: EOF, NULL, &lt;0, !=0, !=N. Функции feof, ferror. Очистка ошибочных состояний: clearerr или перемотка. Сообщение об ошибке: perror, strerror, errno. Функции с переменным числом аргументов. Макросы для перебора параметров. Пример: минимум из нескольких чисел.</p>	4	4	4																
	10.21.Файловый ввод-вывод	4	6		6	6			6											

<p>10.22.Обработка файлов. Файлы как абстракции, связь потоков с файлами (fopen, freopen, fclose). Разница между текстовыми и бинарными файлами. Функции для работы с файлами (getc, fgetc, getchar; putc, fputc, putchar; ungetc; fgets, gets, fputs, puts; fprintf, fscanf; fwrite, fread). Формат бинарных файлов: свобода в интерпретации последовательностей байтов. Традиционные подходы: сырой («raw») формат, однородные файлы записей, блоковая («chunk») структура с заголовками, каталог. Функции для произвольного доступа (ftell, fseek, fgetpos, fsetpos, rewind). Ошибки ввода-вывода и чтение после конца файла, индикация с помощью возвращаемого значения: EOF, NULL, &lt;0, !=0, !=N. Функции feof, ferror. Очистка ошибочных состояний: clearerr или перемотка. Сообщение об ошибке: perror, strerror, errno. Функции с переменным числом аргументов. Макросы для перебора параметров. Пример: минимум из</p>	4						6				6										
<p><b>Раздел 11. Взаимодействие с ОС</b></p>	4																				



## Содержание дисциплины

### Раздел 1. Введение, историческая справка, поколения ЯП. Классификация, общие концепции ЯП

1.1 Классификация ЯП. Парадигмы программирования. Общие принципы построения и использования языков программирования. Средства описания данных. Средства описания действий. Команды-инструкции (ветвление, циклы). Переменные. Подпрограммы (процедуры, функции). Простейшая программа на языке С. Комментарии. Функции для консольного ввода-вывода (printf, scanf, getc). Современные интегрированные среды разработки программ. Графический интерфейс пользователя.

Отладчики. Генераторы кода приложений. Общая характеристика языков ассемблера: назначение, принципы построения и использования; структура языка, основные группы команд, операторы, средства взаимодействия с операционной системой.

### Раздел 2. Структура программы на языке С. Типы данных

2.2. Нотация в форме BNF и ее расширения. Программа на С как набор описаний (переменных, констант, типов, функций, прототипов функций). Алфавит языка. Идентификаторы. Простые типы (целочисленный, вещественный, перечисляемый, пустой — void, символьный). Представление чисел в различных системах счисления. Представление целочисленного и вещественного типов.

2.3 Основы консольного ввода-вывода. Переменные в языке С++

### Раздел 3. Определение констант и типов. Команды

Определение именованных констант. Определение типов (typedef). Области видимости. Команды (инструкции, операторы). Команды вычисления выражений. Побочные эффекты (side-effects). Операции и операнды. Унарные, бинарные, префиксные, постфиксные и инфиксные операции. Приоритет. Ассоциативность (лево- и право-). Основные операции в С. Неявное приведение типов.

Управляющие конструкции. Условный оператор

### Раздел 4. Основные команды языка. Функции

Условный оператор. Составная инструкция. Операторы циклов (for, do...while, while). Инструкции преждевременного выхода и перехода к следующему циклу. Инструкция возврата из функции. Инструкция выбора. Функции. Объявление и определение. Формальные параметры. Неопределенность порядка вычисления аргументов.

Прототипы функций. Тип void. Обработка исключительных ситуаций. Параллельная обработка.

Управляющие конструкции. Оператор цикла с параметром

4.8 Управляющие конструкции. Оператор цикла с (пред-, пост-) условием

4.9 Сочетание цикла и вложенного условного оператора

### Раздел 5. Массивы. Строки

5.10. Массив как набор однотипных данных. Определение. Обращение к отдельным элементам. Многомерные массивы. Инициализация массивов. Автоопределение внешней размерности. Символы как их коды. Строки как массивы типа char. Символы конца строки в разных ОС. Размер буфера и его ограниченность

5.11 Массивы в языке С++

5.12 Работа с текстовыми строками

### Раздел 6. Указатели и ссылки

6.13. Память. Размещение переменных в памяти. Адреса, указатели. Получение адреса (&) и разыменование (\*). Операции над указателями (арифметика, сравнение). Нетипизированные указатели. Расположение элементов массива в памяти. Имя массива как адрес первого элемента. Операция []. Особенности работы с многомерными массивами. Передача аргументов в функцию. Статическое и динамическое выделение памяти. Динамические массивы. Функции malloc, calloc, realloc, free. Основные ошибки: неинициализированные указатели, висячие ссылки, утечки памяти. Три способа создания многомерных динамических массивов — адресация в одномерном по формуле, адресация в одномерном при фиксированной размерности и массив указателей на

## **Раздел 7. Классы памяти. Рекурсия**

7.15. Существование переменных в пространстве и времени. Спецификаторы классов памяти (auto, register, static, extern). Области памяти: регистры процессора, сегмент стека, сегмент данных, динамическая область памяти (куча). Прямая и косвенная рекурсия. Дублирование всех локальных переменных. Пример: вычисление факториала. Хвостовая рекурсия, эквивалентность циклу. Указатели на функции. Определение переменных и типов. Области применения: пере передача в качестве параметра, массивы указателей на функцию.

## **Раздел 8. Препроцессор языка C++**

8.16. Макропроцессоры и макрогенераторы. Предобработка (препроцессинг) исходных текстов. Стадии препроцессорной обработки: объединение строк, удаление комментариев, распознавание директив и лексем, выполнение директив. Директивы #include, #define, #undef. Макроподстановки с параметрами. Цепочка подстановок. Препроцессорные операции: кватирование («преобразование в строку»), склейка препроцессорных лексем. Некоторые предопределенные макросы. Условное включение текста (#ifdef, #ifndef, #if, #elif, #else, #endif). Препроцессорная операция defined. Вспомогательные директивы: #pragma, #line, #error.

## **Раздел 9. Абстрактные типы данных. Примеры составных типов**

Определение АДТ. Структуры. Определение структур, теги и имена, даваемые typedef. Инициализация, доступ к элементам. Наличие присваивания в отличие от массивов. Отсутствие сравнения. Указатели и доступ к полям через указатель. Неоднозначность расположения в памяти. Объединения и перечисления. Особенности расположения в памяти и инициализации. Доступ к полям. Endianness. Битовые поля в структурах и объединениях, зависимость расположения от endian. Инкапсуляция, спецификация, реализация, параметризация, классы и объекты.

### **Структуры данных**

Определение АДТ. Структуры. Определение структур, теги и имена, даваемые typedef. Инициализация, доступ к элементам. Наличие присваивания в отличие от массивов. Отсутствие сравнения. Указатели и доступ к полям через указатель. Неоднозначность расположения в памяти. Объединения и перечисления. Особенности расположения в памяти и инициализации. Доступ к полям. Endianness. Битовые поля в структурах и объединениях, зависимость расположения от endian. Инкапсуляция, спецификация, реализация, параметризация, классы и объекты.

## **Раздел 10. Файловый ввод-вывод. Функции с переменным числом аргументов**

10.20. Обработка файлов. Файлы как абстракции, связь потоков с файлами (fopen, freopen, fclose). Разница между текстовыми и бинарными файлами. Функции для работы с файлами (getc, fgetc, getchar; putc, fputc, putchar; ungetc; fgets, gets, fputs, puts; fprintf, fscanf; fwrite, fread). Формат бинарных файлов: свобода в интерпретации последовательностей байтов. Традиционные подходы: сырой («raw») формат, однородные файлы записей, блоковая («chunk») структура с заголовками, каталог. Функции для произвольного доступа (ftell, fseek, fgetpos, fsetpos, rewind). Ошибки ввода-вывода и чтение после конца файла, индикация с помощью возвращаемого значения: EOF, NULL, <0, !=0, !=N. Функции feof, ferror. Очистка ошибочных состояний: clearerr или перемотка. Сообщение об ошибке: perror, strerror, errno. Функции с переменным числом аргументов.

Макросы для перебора параметров. Пример: минимум из нескольких чисел.

### **10.21. Файловый ввод-вывод**

10.22. Обработка файлов. Файлы как абстракции, связь потоков с файлами (fopen, freopen, fclose). Разница между текстовыми и бинарными файлами. Функции для работы с файлами (getc, fgetc, getchar; putc, fputc, putchar; ungetc; fgets, gets, fputs, puts; fprintf, fscanf; fwrite, fread). Формат бинарных файлов: свобода в интерпретации последовательностей байтов. Традиционные подходы: сырой («raw») формат, однородные файлы записей, блоковая («chunk») структура с заголовками, каталог. Функции для произвольного доступа (ftell, fseek, fgetpos, fsetpos, rewind). Ошибки ввода-вывода и чтение после конца файла, индикация с помощью возвращаемого значения: EOF, NULL, <0, !=0, !=N. Функции feof, ferror. Очистка ошибочных состояний: clearerr или перемотка. Сообщение об ошибке: perror, strerror, errno. Функции с переменным числом аргументов. Макросы для перебора параметров. Пример: минимум из нескольких чисел.

## **Раздел 11. Взаимодействие с ОС**



11.23.Функция main(). Параметры командной строки. Возвращаемое значение. Трансляторы с языков программирования. Компиляторы и интерпретаторы. Стадии трансляции программы на языке С++: препроцессинг, компиляция, компоновка. Программы из нескольких модулей. Создание и использование библиотек функций.

11.24.Функция main(). Параметры командной строки. Возвращаемое значение. Трансляторы с языков программирования. Компиляторы и интерпретаторы. Стадии трансляции программы на языке С++: препроцессинг, компиляция, компоновка. Программы из нескольких модулей. Создание и использование библиотек функций.

## **Раздел 12. Обзор стандартных библиотек функций**

12.25.Библиотеки программ и классов. Стандарты языков программирования. Обзор стандартных библиотек функций языка С++.

12.26.Библиотеки программ и классов. Стандарты языков программирования. Обзор стандартных библиотек функций языка С++.

## **5. Образовательные технологии**

На каждом практическом занятии проводится разбор кодов конкретных программ, написанных на современном языке программирования. По существу, каждое занятие является мастер-классом по соответствующей теме дисциплины.

По пройденному материалу проводится контрольная проверка, результаты которой входят в накопленную оценку модуля.

Задания в тестовой форме применяются для обучения студентов и проведения промежуточных и итогового контролей.

## **6. Учебно-методическое обеспечение самостоятельной работы студентов. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.**

Самостоятельная работа студентов включает текущую и творческую проблемно-ориентированную самостоятельную работу (ТСР).

**Текущая СРС** – работа с лекционным материалом, подготовка к лабораторным работам; опережающая самостоятельная работа; изучение тем, вынесенных на самостоятельную проработку; подготовка к контрольным работам и к экзамену.

**Творческая проблемно-ориентированная самостоятельная работа (ТСР)** – поиск, анализ, структурирование и презентация информации по теме раздела дисциплины, применительно к индивидуальному проекту.

### **План самостоятельной работы студентов**

№ нед.	Тема	Вид самостоятельной работы	Задание	Рекомендуемая литература	Количество часов
	Тема-1.1.-2.2.	Лабораторные	Основы консольного ввода-вывода. Переменные в языке С++	Л2.3, Л1.1, Л2.4	8
	Тема 3.4	Лабораторные	Управляющие конструкции. Условный оператор	Л2.1, Л2.2,	2

				Л1.1	
	Тема 4.6	Лабораторные	Управляющие конструкции. Оператор цикла с параметром	Л2.2, Л1.1	2
	Тема 4.6	Лабораторные	Управляющие конструкции. Оператор цикла с (пред-, пост-) условием	Л2.2, Л1.1	4
	Тема 4.6.	Лабораторные	Сочетание цикла и вложенного условного оператора	Л2.1, Л2.2, Л1.1	4
	Тема 5.10.	Лабораторные	Массивы в языке С++	Л2.2, Л1.1	4
	Тема 5.10.	Лабораторные	Работа с текстовыми строками	Л2.2, Л1.1	4
	Тема 6.13.	Лабораторные	Вложенные циклы. Работа с таблицами и матрицами	Л2.2, Л1.1	8
	Тема 9.17.	Лабораторные	Структуры данных	Л2.1, Л2.2, Л2.3, Л1.1, Л2.4	8
	Тема 9.17.	Самостоятельная работа	Определение АД. Структуры. Определение структур, теги и имена, даваемые typedef. Инициализация, доступ к элементам. Наличие присваивания в отличие от массивов. Отсутствие сравнения. Указатели и доступ к полям через указатель. Неоднозначность расположения в памяти. Объединения и перечисления. Особенности расположения в памяти и инициализации. Доступ к полям. Endianness. Битовые поля в структурах и объединениях, зависимость расположения от endian. Инкапсуляция, спецификация, реализация, параметризация, классы и объекты.	Л2.1, Л2.2, Л2.3, Л1.2, Л2.4	6
	Тема 10.20.	Лабораторные	Файловый ввод-вывод	Л1.1, Л2.4	6
	Тема 10.20.	Самостоятельная работа	Обработка файлов. Файлы как абстракции, связь потоков с файлами (fopen, freopen, fclose). Разница между текстовыми и бинарными файлами. Функции для работы с файлами (getc, fgetc, getchar; putc, fputc, putchar; ungetc; fgets, gets, fputs, puts; fprintf, fscanf; fwrite, fread). Формат бинарных файлов: свобода в интерпретации последовательностей байтов. Традиционные подходы: сырой («raw») формат, однородные файлы записей, блоковая («chunk») структура с заголовками, каталог. Функции для произвольного доступа (ftell, fseek, fgetpos, fsetpos, rewind). Ошибки ввода-вывода и чтение после конца файла, индикация с помощью возвращаемого значения: EOF, NULL, <0, !=0, !=N. Функции feof, ferror. Очистка ошибочных состояний: clearerr или перемотка. Сообщение об ошибке: perror, strerror, errno. Функции с переменным числом аргументов. Макросы для перебора параметров. Пример: минимум из нескольких чисел.	Л1.1, Л2.4	6
	Тема 11.23.	Самостоятельная работа	Функция main(). Параметры командной строки. Возвращаемое значение. Трансляторы с языков программирования. Компиляторы и интерпретаторы. Стадии трансляции программы на языке С: препроцессинг, компиляция, компоновка. Программы из нескольких модулей. Создание и	Л1.1, Л1.2, Л2.4	4

			использование библиотек функций		
	Тема 12.25	Самостоятельная работа	Библиотеки программ и классов. Стандарты языков программирования. Обзор стандартных библиотек функций языка С.	Л1.1, Л1.2, Л2.4	3

### **Методические указания по организации самостоятельной работы студентов**

Самостоятельная работа студентов организуется в компьютерном классе с развернутой ЛВС, имеющей подключение к сети Интернет и обеспечивающей доступ к ресурсам электронного обучения, современным профессиональным базам данных и информационным справочным системам.

Рекомендуется проведение следующих видов самостоятельной работы:

- подготовка к практическим занятиям: изучить теоретический материал по теме практического занятия, ответить на контрольные вопросы;

- подготовка статьи на студенческую конференцию ФВТ: изучить литературу по выбранной теме, обобщить материал, изучить требования к оформлению статьи, представить оформленную статью;

- работа с конспектом лекций и изучение рекомендованной литературы: изучить конспект лекций, ответить на контрольные вопросы, изучить разделы рекомендованной литературы;

- подготовка к экзамену: повторить материал, изученный в течение семестра, студентам из числа лиц с ограниченными возможностями здоровья могут быть предложены электронные образовательные ресурсы в формах, адаптированных к ограничениям их здоровья.

### **Материалы для проведения текущего и промежуточного контроля знаний студентов**

#### ***Контроль освоения компетенций***

№ п\п	Вид контроля	Контролируемые темы (разделы)	Компетенции, компоненты которых контролируются
1	Контроль усвоения теоретического материала – коллоквиумы, тесты по разделам дисциплины	Раздел 1.Тема 1.1 Раздел 2. Тема 2.2 Раздел 3. Тема:3.4	УК-9, ОПК-6, ПК-3
2	Контроль усвоения теоретического материала – коллоквиумы, тесты по разделам дисциплины	Раздел 4.Тема 4.6 Раздел 5. Тема 5.10 Раздел 6. Тема:6.13	УК-9, ОПК-6, ПК-3
3	Контроль усвоения теоретического	Раздел 7.Тема 7.15 Раздел 8. Тема 8.16	УК-9, ОПК-6, ПК-3

	материала – коллоквиумы, тесты по разделам дисциплины	Раздел 9. Тема:9.17	
4	Контроль усвоения теоретического материала – устный опрос	Раздел 10. Тема 10.20 Раздел 11.Тема 11.23 Раздел 12. Тема:12.25	УК-9, ОПК-6, ПК-3
5.	Полнота выполнения практического задания	Раздел 1 -12 Основы консольного ввода-вывода. Переменные в языке С Управляющие конструкции. Условный оператор Управляющие конструкции. Оператор цикла с параметром Управляющие конструкции. Оператор цикла с (пред-, пост-) условием. Сочетание цикла и вложенного условного оператора Массивы в языке С. Работа с текстовыми строками. Вложенные циклы. Работа с таблицами и матрицами. Структуры данных Файловый ввод-вывод	УК-9, ОПК-6, ПК-3

## 7. Учебно-методическое и материально-техническое обеспечение дисциплины

При изучении дисциплины для проработки всех тем и выполнения заданий по всем темам студенты могут использовать различные учебно-методические материалы, размещаемые в электронном виде преподавателями, которая предполагает также возможность обмена информацией с преподавателем для подготовки заданий.

Материально-техническое обеспечение образовательного процесса по дисциплине **Языки программирования** включает в себя следующие компоненты:

- Учебные аудитории для контактной работы с преподавателем, укомплектованных специализированной мебелью (столы и стулья).компьютерные классы с набором лицензионного базового программного обеспечения для проведения лабораторных занятий с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду;
- Дополнительные мультимедийные материалы, мультимедийная аудитория; Skype, для проведения дистанционного обучения и консультаций

### Учебная литература:

1. Основная литература				
	Авторы	Заглавие	Издательство, год	Эл. адрес
1.	Царев Р. Ю.	Программирование на языке Си: Учебники и учебные пособия для ВУЗов	Сибирский федеральный университет, 2014 // ЭБС Университетская библиотека Online	<a href="http://biblioclub.ru/index.php?page=book&amp;id=364601">http://biblioclub.ru/index.php?page=book&amp;id=364601</a>

2.	Трофимов В.В. - отв. ред.	Алгоритмизация и программирование: Учебник для академического бакалавриата	М.: Издательство Юрайт, 2018 // ЭБС "Юрайт"	<a href="https://www.biblionline.ru/book/algoritmizaciya-i-programmirovaniye-414652">https://www.biblionline.ru/book/algoritmizaciya-i-programmirovaniye-414652</a>
2. Дополнительная литература				
	Авторы	Заглавие	Издательство, год	Эл. адрес
3.	Юдинцев А.Ю., Трошкина Г.Н., Драгун И.А.	Основы алгоритмизации и языки программирования: учеб.-метод. пособие	Барнаул: Изд-во АлтГУ, 2011	<a href="http://elibrary.asu.ru/handle/asu/651">http://elibrary.asu.ru/handle/asu/651</a>
4.	Незнанов А.А.	Программирование и алгоритмизация: учебник	М.: Академия, 2010	
5.	Синицын С.В., Михайлов А.С., Хлытчиев О.И.	Программирование на языке высокого уровня: учебник	М.: Академия, 2010	
6.	Гниденко И. Г., Павлов Ф. Ф., Федоров Д. Ю.	Технология разработки программного обеспечения: Учебное пособие	Юрайт, 2017 // ЭБС "Юрайт"	<a href="https://www.biblionline.ru/book/tehnologiya-razrabotki-programmnogo-obespecheniya-415606">https://www.biblionline.ru/book/tehnologiya-razrabotki-programmnogo-obespecheniya-415606</a>

### Интернет-ресурсы

Название ресурса	Ссылка/доступ
Электронная библиотека онлайн «Единое окно к Образовательным ресурсам»	<a href="http://window.edu.ru">http://window.edu.ru</a>
«Образовательный ресурс России»	<a href="http://school-collection.edu.ru">http://school-collection.edu.ru</a>
Федеральный образовательный портал: учреждения, программы, стандарты, ВУЗы, тесты ЕГЭ, ГИА	<a href="http://www.edu.ru">http://www.edu.ru</a> –
Федеральный центр информационно-образовательных ресурсов (ФЦИОР)	<a href="http://fcior.edu.ru">http://fcior.edu.ru</a> -
ЭБС "КОНСУЛЬТАНТ СТУДЕНТА". Электронная библиотека технического вуза	<a href="http://polpred.com/news">http://polpred.com/news</a>
Издательство «Лань». Электронно-библиотечная система	<a href="http://www.studentlibrary.ru">http://www.studentlibrary.ru</a> -
Русская виртуальная библиотека	<a href="http://rvb.ru">http://rvb.ru</a> –
Кабинет русского языка и литературы	<a href="http://ruslit.ioso.ru">http://ruslit.ioso.ru</a> –
Национальный корпус русского языка	<a href="http://ruscorpora.ru">http://ruscorpora.ru</a> –
Издательство «Лань». Электронно-библиотечная система	<a href="http://e.lanbook.com">http://e.lanbook.com</a> -
Еженедельник науки и образования Юга России «Академия»	<a href="http://old.rsue.ru/Academy/Archives/Index.htm">http://old.rsue.ru/Academy/Archives/Index.htm</a>
Научная электронная библиотека «e-Library»	<a href="http://elibrary.ru/defaultx.asp">http://elibrary.ru/defaultx.asp</a> -
Электронно-библиотечная система IPR books	<a href="http://www.iprbookshop.ru">http://www.iprbookshop.ru</a> -
Электронно-справочная система документов в сфере образования «Информо»	<a href="http://www.informio.ru">http://www.informio.ru</a>

Информационно-правовая система «Консультант-плюс»	Сетевая версия, доступна со всех компьютеров в корпоративной сети ИнГГУ
Информационно-правовая система «Гарант»	Сетевая версия, доступна со всех компьютеров в корпоративной сети ИнГГУ
Электронно-библиотечная система «Юрайт»	<a href="https://www.biblio-online.ru">https://www.biblio-online.ru</a>
Система электронного документооборота	<a href="http://www.directum.ru">http://www.directum.ru</a>
Веб-система управления проектами	<a href="http://www.advanta-group.ru/">http://www.advanta-group.ru/</a>

### **Программное обеспечение**

Для проведения лекционных и лабораторных занятий рекомендуется использовать программное обеспечение: операционная система ОС Windows 7 и выше, пакет Microsoft Office 2010 и выше, обслуживающие программы и среды разработки программ по выбору преподавателей.

### **Материально-техническое обеспечение**

Для проведения лекций по дисциплине используются специализированные аудитории с мультимедийным оборудованием или с возможностями подключения к такому оборудованию, позволяющему демонстрировать на большом экране приемы работы с персональным компьютером и другой лекционный материал (технические характеристики компьютера, входящего в состав мультимедийного оборудования или используемого совместно с таким оборудованием, должны обеспечивать возможность работы с современными версиями ОС Windows, пакета Microsoft Office, обслуживающих, прикладных программ и другого ПО).

Для проведения лабораторных занятий по дисциплине и для самостоятельной работы студентов используются специализированные аудитории, оснащенные персональными компьютерами, при проведении лабораторных занятий используются современное программное обеспечение (операционную систему Windows 7 и выше, пакет Microsoft Office 2010 и выше, а также обслуживающие программы и среды разработки программ по выбору преподавателей).

Рабочая программа дисциплины **Б1.В.10 Инструментальные средства информационных систем** составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.02 Информационные системы и технологии, утвержденного приказом Министерства образования и науки Российской Федерации от «19» сентября 2017 г. №926, с учетом примерной программы учебной дисциплины из ПООП.

:

Программу составила :  
Старший преподаватель А.М.Даурбекова

Программа одобрена на заседании кафедры  
»Информационные системы и технологии»  
Протокол №10 от «20» июня 2022г.

Программа одобрена Учебно-методическим советом физико-математическим факультетом  
Протокол № 10 от «22» июня 2022г.

Программа одобрена на заседании Учебно-методического совета университета  
Протокол № 10 от «29» июня 2022г.

**Сведения о переутверждении программы на очередной учебный год и регистрации изменений**

Учебный год	Решение кафедры (№ протокола, дата)	Внесенные изменения	Подпись зав. кафедрой