



АННОТАЦИЯ

рабочей программы учебной дисциплины **Б1.В.ДВ.01.01 Языки программирования**

Направление подготовки бакалавриата **09.03.02 Информационные системы и технологии**

| | | | |
|----|--|--|--------------------|
| 1. | Цель изучения 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. | | |
| 2. | Место дисциплины в структуре ОПОП ВО 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. | | |
| | Результаты освоения дисциплины (модуля) «Языки программирования» | | |
| | Код и наименование компетенций | Индикаторы | Дескрипторы |
| | Универсальные компетенции (УК) | | |



**Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Ингушский государственный университет»
ФИЗИКО-МАТЕМАТИЧЕСКИЙ факультет**

| | | | |
|---|--|--|---|
| | <p>УК-9 Способен принимать обоснованные экономические решения в различных областях жизнедеятельности</p> | <p>УК-9.1. Понимает базовые принципы функционирования экономики и 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.</p> | <p>Понимать: базовые принципы функционирования экономики и экономического развития, цели и формы участия государства в 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.</p> |
| Общепрофессиональные компетенции (ОПК) | | | |
| | <p>ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий.</p> | <p>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</p> | <p>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.</p> |
| Профессиональные компетенции (ПК) | | | |



| | | | |
|----|---|---|--|
| | <p>ПК- 3 Способен выполнять интеграцию программных модулей и компонент</p> | <p>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 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>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 среды программирования, средства пакетного выполнения процедур;</p> <p>Уметь: 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>Иметь навыки: разработки процедур сборки модулей и компонент программного обеспечения; разработки процедур развертывания и обновления программного обеспечения; разработки процедур миграции и преобразования конвертации</p> |
| 4. | <p>Структура и содержание дисциплины «Языки программирования»</p> <p>4.1. Структура дисциплины (модуля)</p> | | |



| Вид учебной работы | Всего | Порядковый номер семестра | | | |
|--|------------------|---------------------------|--|--|--|
| | | 2 | | | |
| Общая трудоемкость дисциплины всего (в з.е.), в том числе: | 5 з.е. | | | | |
| Курсовой проект (работа) | не предусмотрено | | | | |
| Аудиторные занятия всего (в акад. часах), в том числе: | 84 | 84 | | | |
| Лекции | 34 | 34 | | | |
| Практические занятия, семинары | | | | | |
| Лабораторные работы | 50 | 50 | | | |
| Самостоятельная работа всего (в акад. часах), в том числе: | 69 | 69 | | | |
| КСР | | | | | |
| Экзамен | 27 | 27 | | | |
| Общая трудоемкость дисциплины | 180 | 180 | | | |
| 4.2. Содержание дисциплины | | | | | |
| <p>Раздел 1. Введение, историческая справка, поколения ЯП. Классификация, общие концепции ЯП</p> <p>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 C. Комментарии. Функции для консольного ввода-вывода (printf, scanf, getc). Современные интегрированные среды разработки программ. Графический интерфейс пользователя.</p> <p>Отладчики. Генераторы кода приложений. Общая характеристика языков ассемблера: назначение, принципы построения и использования; структура языка, основные группы команд, операторы, средства взаимодействия с операционной системой.</p> <p>Раздел 2. Структура программы на языке C. Типы данных</p> <p>2.2. Нотация в форме BNF и ее расширения. Программа на C 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.</p> <p>2.3 Основы консольного ввода-вывода. Переменные в языке C++</p> <p>Раздел 3. Определение констант и типов. Команды</p> <p>3.4.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 C. Неявное приведение типов.</p> <p>3.5.Управляющие конструкции. Условный оператор</p> <p>Раздел 4. Основные команды языка. Функции</p> <p>4.6. 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.</p> <p>Прототипы функций. Тип void. Обработка исключительных ситуаций. Параллельная обработка.</p> <p>4.7.Управляющие конструкции. Оператор цикла с параметром</p> <p>4.8Управляющие конструкции. Оператор цикла с (пред-, пост-) условием</p> <p>4.9Сочетание цикла и вложенного условного оператора</p> <p>Раздел 5. Массивы. Строки</p> | | | | | |



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 C++

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 массивы.

6.14. Вложенные циклы. Работа с таблицами и матрицами



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: 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

9.17.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.

9.18.Структуры данных

9.19.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; fgetc, 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; fgetc, 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 C++: 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 C++: 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 C++.

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

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. Используемые ресурсы информационно-телекоммуникационной сети «Internet»; информационные технологии, программные средства и информационно-справочные системы

Информационное обеспечение

базы данных, информационно-справочные и поисковые системы

| Название ресурса | Ссылка/доступ |
|--|---|
| Электронная библиотека онлайн «Единое окно к Образовательным ресурсам» | http://window.edu.ru |
| «Образовательный ресурс России» | http://school-collection.edu.ru |
| Федеральный образовательный портал: учреждения, программы, стандарты, ВУЗы, тесты ЕГЭ, ГИА | http://www.edu.ru – |
| Федеральный центр информационно-образовательных ресурсов (ФЦИОР) | http://fcior.edu.ru – |
| ЭБС "КОНСУЛЬТАНТ СТУДЕНТА". | http://polpred.com/news |
| Электронная библиотека технического вуза | http://www.studentlibrary.ru – |
| Издательство «Лань». Электронно-библиотечная система | http://rvb.ru – |
| Русская виртуальная библиотека | http://rvb.ru – |



**Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Ингушский государственный университет»
ФИЗИКО-МАТЕМАТИЧЕСКИЙ факультет**

| | | |
|-----------|---|---|
| | Кабинет русского языка и литературы | http://ruslit.ioso.ru – |
| | Национальный корпус русского языка | http://ruscorpora.ru – |
| | Издательство «Лань». Электронно-библиотечная система | http://e.lanbook.com – |
| | Еженедельник науки и образования Юга России «Академия» | http://old.rsue.ru/Academy/Archives/Index.htm |
| | Научная электронная библиотека«e-Library» | http://elibrary.ru/defaultx.asp – |
| | Электронно-библиотечная система IPR books | http://www.iprbookshop.ru – |
| | Электронно-справочная система документов в сфере образования «Информιο» | http://www.informio.ru |
| | Информационно-правовая система «Консультант-плюс» | Сетевая версия, доступна со всех компьютеров в корпоративной сети ИнГУ |
| | Информационно-правовая система «Гарант» | Сетевая версия, доступна со всех компьютеров в корпоративной сети ИнГУ |
| | Электронно-библиотечная система «Юрайт» | https://www.biblio-online.ru |
| | Система электронного документооборота | http://www.directum.ru |
| | Веб-система управления проектами | http://www.advanta-group.ru/ |
| 7. | Формы текущего контроля | |
| | Коллоквиумы, тесты по разделам дисциплины | |
| 8. | Форма промежуточного контроля | |
| | Экзамен | |

Разработчик: ст.преподаватель Ужахова М.М.