


«УТВЕРЖДАЮ»

И.о. декана факультета
математики и компьютерных наук

 / А.Х.Сташ

30 июня 2020 г.

Рабочая программа дисциплины (модуля)

Б1.О.29 Алгоритмические языки и методы программирования

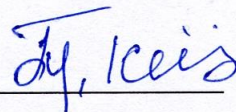
направление подготовки 44.03.01 Педагогическое образование

направленность (профиль) Математическое образование

РПД адаптирована для лиц с ограниченными возможностями здоровья и инвалидов

Кафедра прикладной математики, информационных технологий и информационной безопасности

Составитель: к.п.н., доцент, М.А. Коджешау



Рассмотрена и одобрена на заседании кафедры прикладной математики, информационных технологий и информационной безопасности, протокол №11 от « 30 » июня 2020 г.

Заведующий кафедрой: к.ф.м.н., доцент, М.В. Алиев



Согласовано:

Председатель УМК факультета: доцент кафедры прикладной математики, информационных технологий и информационной безопасности, кандидат пед. наук, доцент Ш.Т. Меретуков



Содержание

	стр.
Пояснительная записка	4
1. Цели и задачи дисциплины (модуля)	6
2. Объём дисциплины (модуля) по видам учебной работы	6
3. Содержание дисциплины (модуля)	8
4. Самостоятельная работа обучающихся	8
5. Учебно-методическое обеспечение дисциплины (модуля)	11
6. Методические рекомендации по дисциплине (модулю)	11
7. Материально-техническое обеспечение дисциплины (модуля)	13
8. Лист регистрации изменений	14

Пояснительная записка

Рабочая программа дисциплины составлена в соответствии с требованиями ФГОС 3++ ВО по направлению подготовки (специальности) 44.03.01. Педагогическое образование (направленность Математическое образование). РП представляет собой совокупность дидактических материалов, направленных на реализацию содержательных, методических и организационных условий подготовки по направлению подготовки (специальности) 44.03.01. Педагогическое образование (направленность Математическое образование). Дисциплина относится к дисциплинам, направленным на подготовку к преподавательской деятельности.

Дисциплина относится к вариативным (указать место дисциплины (модуля) в структуре образовательной программ).

Трудоемкость дисциплины: 3 з.е./108ч.;

контактная работа:

занятия лекционного типа – 12 ч.,

занятия семинарского типа (лабораторные) – 26 ч.,

контроль самостоятельной работы – 0 ч.,

иная контактная работа – 0,3 ч.,

контролируемая письменная работа – 0 ч.,

СР – 34 ч.,

контроль – 35,7 ч.

Ключевые слова: алгоритм, программа, структуры данных, цикл, ветвление, процедура, функция, массив, рекурсия, объектно-ориентированное программирование.

Составитель: Коджешау М.А., к.п.н., доцент, доцент кафедры прикладной математики, информационных технологий и информационной безопасности

1. Цели и задачи дисциплины (модуля).

Цель дисциплины (модуля): изучение языка программирования Питон, методов алгоритмизации, структур данных

Задачи дисциплины (модуля):

1. Изучить теоретические основы программирования на языке Питон;
2. Научить выбору методов и приемов решения задач;
3. Овладеть методами и приемами решения задач предметной области.

Показателями компетенций являются:

Знания: общие принципы построения алгоритмов, основные алгоритмические конструкции; понятие системы программирования; основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти; подпрограммы, составление библиотек программ; объектно – ориентированную модель программирования, понятие классов и объектов, их свойств и методов; основные приемы программирования.; интегрированные среды изучаемых языков программирования.

Умения: использовать языки программирования; строить логически правильные и эффективные программы; составлять простые блок-схемы алгоритмов; составлять программы на алгоритмическом языке высокого уровня; работать в интегрированной среде изучаемых языков программирования; взаимодействовать со специалистами смежного профиля при разработке, внедрению и применению объектов профессиональной деятельности; производить модификацию отдельных модулей программы; производить тестирование программного продукта на выявление ошибок.

Навыки: разработка программ с использованием основных алгоритмических конструкций; использования основных методов программирования; использования структур данных при

решении задач и организации данных; поиска и устранения ошибок в разработанных программах.; построения логически правильных и эффективных программ; использования основных приемов программирования.

Таблица 1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций

Компетенция (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Результаты обучения
УК-2. Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений.	УК-2.1. Знает необходимые для осуществления профессиональной деятельности правовые нормы	Знает: приемы и методы решения задач по темам курса, структур данных
	УК-2.2. Умеет определять круг задач в рамках избранных видов профессиональной деятельности, планировать собственную деятельность исходя из имеющихся ресурсов; соотносить главное и второстепенное, решать поставленные задачи в рамках избранных видов профессиональной деятельности.	Умеет: правильно выбирать методы решения задач по темам курса
	УК-2.3. Имеет практический опыт применения нормативной базы и решения задач в области избранных видов профессиональной деятельности	Навыки: использования современных технических и программных средств для решения учебных задач по разделам дисциплины
ПК-1. Способен демонстрировать базовые знания математических и естественных наук, программирования и информационных технологий.	ПК-1.1. Обладает базовыми знаниями, полученными в области математических и (или) естественных наук, программирования и информационных технологий.	Знает теоретические основы алгоритмизации и программирования
	ПК-1.2. Умеет находить, формулировать и решать стандартные задачи в собственной научно-исследовательской деятельности в области программирования и	Умения программировать и использовать методы и приемы программирования

	информационных технологий.	
	ПК-1.3. Имеет практический опыт научно-исследовательской деятельности в области программирования и информационных технологий.	имеет практический опыт реализации алгоритмов и программ

2. Объем дисциплины (модуля) по видам учебной работы.

Таблица 2. Объем дисциплины (модуля) общая трудоемкость: 3 з.е.

Виды учебной работы	Всего часов	Распределение по семестрам в часах			
		I	II	III	IV
Общая трудоемкость дисциплины	108			108	
Контактная работа:					
занятия лекционного типа	12			12	
занятия семинарского типа (лабораторные)	26			26	
контроль самостоятельной работы					
иная контактная работа	0,3			0,3	
Самостоятельная работа (СР)	34			34	
Курсовая работа (проект)					
Контроль	35,7			35,7	
Вид промежуточного контроля	Экзамен			Экзамен	

3. Содержание дисциплины (модуля).

Таблица 3. Распределение часов по темам и видам учебной работы

Номер раздела	Наименование разделов и тем дисциплины (модуля)	Объем в часах					
		Всего	Л	ПЗ	С	ЛР	СР и иная работа
1	Модуль 1. Основные принципы алгоритмизации и программирования		3			6	20
1.1	Введение. Предмет и задачи дисциплины основы алгоритмизации и программирования						4
1.2	Понятие программы и программного обеспечения						4
1.3	Понятие алгоритма. Свойства алгоритмов.						4

1.4	Понятие и элементы блок-схем. Основные алгоритмические конструкции						4
1.5	Проектирование блок-схем.						4
2	Модуль 2. Языки и системы программирования Логические основы алгоритмизации		3			6	20
2.1	Основы алгебры логики. Логические операции с высказываниями: конъюнкция, дизъюнкция, инверсия. Законы логических операций. Таблицы истинности.						4
2.2	Среда программирования. Интерфейс среды программирования.						4
2.3	Линейный алгоритм. Структура оператора на языках Питон.						4
2.4	Разветвляющийся алгоритм. Полная и не полная конструкция. Арифметические выражения. Стандартные функции. Инструкции if/else/elif, логические операторы и выражения сравнения. Функции в Python. Основные понятия						4
2.5	Циклический алгоритм. Полная и не полная конструкция.						2
2.6	Цикл с пред условием. Цикл с пост условием. Циклы в Python						2
3	Модуль 3. Структуры данных. Процедура и функции. Рекурсия.		3			6	20
3.1	Процедуры и функции						4
3.2	Структурированные типы данных массивы						4
3.3	Строковый тип данных						4
3.4	Файловый тип данных						4
3.5	Методы поиска и сортировки данных.						2
3.6	Рекурсия						2
4	Модуль 4. Работа в текстовом и графическом режимах в Turbo Pascal		3			8	10
4.1	Текстовый режим работы модуль CRT.						2

	Работа с текстом в графическом режиме.						
4.2	Графический режим работы модуль GRAPH. Запуск графической системы						2
4.3	Процедуры и функции рисования объектов.						2
4.4	Программирование окон, звука, цвета символов и фона.						2
4.5	Имитация анимированных изображений						2

4. Самостоятельная работа обучающихся.

Таблица 4. Содержание самостоятельной работы обучающихся

№ п/п	Вид самостоятельной работы	Разделы или темы рабочей программы	Форма отчетности
	Модуль 1		
1	Подготовка к лабораторным занятиям	1.2.-1.5	Фронтальная беседа
2	Подготовка материалов по вопросам для самостоятельного изучения, указанным в каждой лабораторной работе	1.4.-1.5	Презентация по теме
3	Подготовка ответов на вопросы по содержанию лабораторной работы	1.2.-1.5	Собеседование
4	Отработка навыков решения задач		Выполнение тестов
5	Подготовка отчетов в электронном виде		Предъявление всех выполненных тестов по главам
	Модуль 2		
1	Подготовка к лабораторным занятиям	2.1. -2.6	Фронтальная беседа
2	Подготовка материалов по вопросам для самостоятельного изучения, указанным в каждой лабораторной работе	2.1. -2.6	Собеседование
3	Подготовка ответов на вопросы по содержанию лабораторной работы		Защита результатов, оформленных в письменном виде
4	Отработка навыков решения задач		Выполнение тестов
5	Подготовка отчетов в электронном виде		Предъявление всех выполненных тестов по главам
Модуль 3			

1	Подготовка к лабораторным занятиям	3.1. -3.6	Индивидуальная беседа
2	Подготовка материалов по вопросам для самостоятельного изучения, указанным в каждой лабораторной работе	3.1. -3.6	Отчеты по разработанным программам
3	Подготовка ответов на вопросы по содержанию лабораторной работы		Собеседование
4	Отработка навыков решения задач		Выполнение тестов
5	Подготовка отчетов в электронном виде		Предъявление всех выполненных тестов по главам
Модуль 4			
1	Подготовка к лабораторным занятиям	4.1.-5.5.	Фронтальная беседа
2	Подготовка материалов по вопросам для самостоятельного изучения, указанным в каждой лабораторной работе	4.1.-5.5.	Отчеты по разработанным программам
3	Подготовка ответов на вопросы по содержанию лабораторной работы	4.1.-5.5.	Собеседование
4	Отработка навыков решения задач		Выполнение тестов
5	Подготовка отчетов в электронном виде		Предъявление всех выполненных тестов по главам

4.1. Темы курсовых работ (проектов).

Курсовые работы или семестровые задания не предусмотрены

4.2. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся.

Самостоятельная работа студентов осуществляется с использованием:

1. учебно-методического обеспечения дисциплины;
2. заданий к лабораторным работам для самостоятельного изучения и решения задач по разделам курса;
3. ресурсов информационно-телекоммуникационной сети «Интернет».

5. Учебно-методическое обеспечение дисциплины (модуля).

Таблица 5. Основная литература

№ п/п	Наименование, библиографическое описание
1	Электронное учебное пособие «Основы программирования. PYTHON»/ Т.Ф. Долгих, А.П. Мелехов, Н.М.Поляков, М.Н. Романов.- 2017, 36с.- [Электронный ресурс]. - URL: https://elibrary.ru/item.asp?id=30562738

2	Федоров Д.Ю. Программирование на языке высокого уровня Python : Учебное пособие 2-е изд., пер. и доп. - Сер. 60 Бакалавр. Прикладной курс / Д.Ю. Федоров.- СПбГЭУ, Санкт-Петербург. - 2019 -[Электронный ресурс]. - URL: https://elibrary.ru/item.asp?id=41246219
3	Макарова, Н.П. Методы программирования и информатика. Лабораторный практикум. В 2-х ч / Н.П. Макарова. - Гродно : ГрГУ, 2012. - Ч. I. - 54 с. - ISBN 978-985-515-562-2, 978-985-515-561-5 ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=134239
4	Комлева, Н.В. Методы программирования. Учебно-методический комплекс / Н.В. Комлева, Е.В. Ковалевская. - М. : Евразийский открытый институт, 2011. - 319 с. - ISBN 978-5-374-00356-7 ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=90390

Таблица 6. Дополнительная литература

№ п/п	Наименование, библиографическое описание
1.	Голицына О П, Попов И.И. Основы алгоритмизации и программирования: Учебное пособие.- М.: Форум: Инфра-М, 2004.
2.	Кирнос, В.Н. Информатика II. Основы алгоритмизации и программирования на языке C++ : учебно-методическое пособие / В.Н. Кирнос ; Министерство образования и науки Российской Федерации, Томский Государственный Университет Систем Управления и Радиоэлектроники (ТУСУР). - Томск : Эль Контент, 2013. - 160 с. : ил.,табл., схем. - ISBN 978-5-4332-0068-5 ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=208651
3.	Фаронов В. В. Delphi. Программирование на языке высокого уровня : учебник для вузов / В. В. Фаронов. – СПб. : Питер, 2009. – 639 с.: ил.. – ISBN 978-5-8046-0008-3.
4.	Семакин И.Г., Шестаков А.П. Основы программирования: Учебник. - М.: Академия 2004.
5.	Баженова И. Delphi6: Самоучитель программиста. - М.: КУ-ДИЦ-ОБРАЗ, 2002.
6.	Грызлов В.И., Грызлова Т.П. Турбо Паскаль 7.0. - М.: ДМК, 2000.
7.	Немнюгин С.А. Turbo Pascal. - СПб.: Питер, 2000.
8.	Фаронов В. Delphi6: Учебный курс. - СПб: Питер, 2006.
9.	Фаронов В Турбо Паскаль 7.0 – М: КНОРУС, 2006

Таблица 7. Ресурсы информационно-телекоммуникационной сети «Интернет»

№ п/п	Название (адрес) ресурса
1.	Методическая копилка учителя информатики - http://www.metod-kopilka.ru/
2.	Методическая копилка учителя информатики http://nikaschool3.ucoz.ru/index/0-5
3.	Методическая копилка учителя информатики - http://www.videouroki.net/filecatalog.php
4.	Методическая копилка учителя информатики - http://86sch6-kogalym.edusite.ru/p48aa1.html
5.	Методическая копилка учителя информатики - http://comp-science.narod.ru/
6.	Методическая копилка учителя информатики - http://panoramayrokov.narod.ru/metodik.htm

6. Методические рекомендации по дисциплине (модулю).

Методические рекомендации преподавателю по подготовке и проведению вузовской лекции.

Традиционно подготовка вузовской лекции строится по схеме:

- определение цели изучения материала по данной теме;
- составление плана изложения материала;
- определение основных понятий темы;
- подбор основной литературы к теме.

При подготовке лекции необходимо учитывать следующее:

1. Большое значение имеет временное планирование каждой структурной части лекции и строгое следование такому плану.

2. Необходимо максимально использовать современные технические средства обучения.

3. В случае отсутствия технических средств обучения рационально часть изучаемого материала давать через схемы, начерченные (лучше заранее) на доске. Схемы необходимо использовать для лучшего усвоения, они несут большую смысловую нагрузку.

4. Определить в процессе подготовки лекции отдельные вопросы изучаемой темы, которые будут предлагаться студентам для самостоятельного изучения.

Методические рекомендации преподавателю по подготовке и проведению лабораторных занятий.

Ведущей дидактической целью лабораторных работ является экспериментальное подтверждение и проверка существенных теоретических положений теории алгоритмизации и программирования, изучаемых в дисциплине «Алгоритмические языки и методы программирования». Лабораторные работы должны развивать мышление студентов, самостоятельность при решении практической задачи, формировать глубоких и прочные знания.

Методические рекомендации преподавателю по организации самостоятельной работы студентов.

В изучении курса особое место занимает самостоятельная работа слушателей. Самостоятельность в учебной работе способствует развитию заинтересованности студента в изучаемом материале, вырабатывает у него умение и потребность самостоятельно получать знания.

Используются различные формы самостоятельной работы:

- работа с источниками в читальном зале;
- анализ литературы по теме и составление конспектов, докладов, рефератов;
- практическое выполнение предложенных заданий на ПК.

Выполнение практических заданий предполагает много возможностей применения активных методов обучения и организации самостоятельной работы на основе индивидуального подхода. Поэтому при выполнении работы необходимо:

1. Провести экспресс-опрос (устно или в тестовой форме) по теоретическому материалу, необходимому для выполнения работы (с оценкой).

2. Оценить работу студента в лаборатории и полученные им данные (оценка).

3. Проверить и выставить оценку за выполнение самостоятельного задания.

Запись лекции – одна из форм активной самостоятельной работы студентов, требующая навыков и умения кратко, схематично, последовательно и логично фиксировать основные положения, выводы, обобщения, формулировки. Культура записи лекции – один из важнейших факторов успешного и творческого овладения знаниями. Последующая работа над текстом лекции воскрешает в памяти ее содержание, позволяет развивать аналитическое мышление. В конце лекции преподаватель оставляет время (5-10 минут) для того, чтобы студенты имели возможность задать уточняющие вопросы по изучаемому материалу.

Работа с конспектом лекций предполагает просмотр конспекта в тот же день после занятий, пометку материала конспекта, который вызывает затруднения для понимания. Попытайтесь найти ответы на затруднительные вопросы, используя рекомендуемую литературу. Если самостоятельно не удалось разобраться в материале, сформулируйте вопросы и обратитесь за помощью к преподавателю на консультации или ближайшей лекции. Регулярно отводите время для повторения пройденного материала, проверяя свои знания, умения и навыки по контрольным вопросам.

Для выполнения письменных домашних заданий студентам необходимо внимательно прочитать соответствующий раздел учебника и проработать аналогичные задания, рассматриваемые преподавателем на лекционных занятиях.

Основным методом обучения является самостоятельная работа студентов с учебно-методическими материалами, научной литературой, статистическими данными, в том числе из сети Интернет.

Постоянная активность на занятиях, готовность ставить и обсуждать актуальные проблемы курса - залог успешной работы и положительной оценки.

Вопросы к экзамену

1. Понятие алгоритмических языков программирования. Интерпретаторы, компиляторы и псевдокомпиляторы. Общая схема формирования .EXE файла из исходного кода. Этапы трансляции и компиляции.
2. Интегрированная среда программирования на примере Turbo Pascal или Borland Pascal. Написание текста программ, многооконный редактор.
3. Основные возможности интегрированной среды программирования Turbo Pascal: компиляция с последующим выполнением программы; использование контрольных точек и «пошагового» выполнения программы в режимах “Step over” и “Trace into”; выполнение программы «до курсора»; использование окна “Watches”.
4. Структура программы на языке Pascal (все разделы). Назначение каждого раздела. Очередность описания разделов программы.
5. Алфавит языка. Имена. Правила формирования имен.
6. Переменные: определение, назначение и типы. Предопределенные (стандартные) типы. Раздел описания переменных. Область действия переменных. Абсолютная адресация в памяти.
7. Использование комментариев в программе на языке Pascal. Символы обозначения комментариев. Вложенные комментарии.
8. Предопределенные (стандартные) типы языка Pascal: основные и производные от них. Выбор типа переменных при создании программ.
9. Арифметические операции над целыми и вещественными данными. Приоритеты операций. Изменение приоритета. Математические и тригонометрические функции. Целочисленные операции.
10. Логические операции “и”, “или”, “не”, “или–не”. Таблицы истинности для этих операций. Использование данных логических операций в условных выражениях и при «побитовой» обработке.

11. Простой и составной операторы. Операторные скобки: назначение, правила и порядок их использования.
12. Оператор присваивания. Назначение, синтаксис и порядок использования. Совместимость типов данных в левой и правой частях оператора присваивания.
13. Управляющие операторы языка Pascal. Определение, назначение. Примеры управляющих операторов.
14. Условный оператор. Назначение, синтаксис и порядок использования. Принципиальные отличия от оператора выбора (варианта).
15. Оператор выбора (варианта). Назначение, синтаксис и порядок использования. Принципиальные отличия от условного оператора.
16. Циклический процесс. Определение, назначение. Оператор цикла с заранее известным количеством повторений (синтаксис и порядок работы).
17. Циклический процесс. Определение, назначение. Оператор цикла с предусловием (синтаксис и порядок работы).
18. Циклический процесс. Определение, назначение. Оператор цикла с постусловием (синтаксис и порядок работы).
19. Понятие файла с точки зрения его использования в программе. Типы файлов в языке Pascal. Дескриптор файла. Общая схема работы с файлом.
20. Файлы с прямым и последовательным доступом. Понятие указателя файла. Действия над указателем файла в зависимости от способа доступа к файлу.
21. Понятие «буфера обмена» для файла. Назначение буфера обмена, его использование. Зависимость типа буфера обмена при работе с текстовыми, типизированными и нетипизированными файлами.
22. Текстовые файлы: процедуры и функции обработки данных; действия над указателем файла.
23. Типизированные файлы: процедуры и функции обработки данных; действия над указателем файла.
24. Нетипизированные файлы: процедуры и функции обработки данных; действия над указателем файла.
25. Операции ввода-вывода при работе с текстовыми файлами. Обработка ошибок файлового ввода-вывода.
26. Операции ввода-вывода при работе с типизированными файлами. Обработка ошибок файлового ввода-вывода.
27. Операции ввода-вывода при работе с нетипизированными файлами. Обработка ошибок файлового ввода-вывода.
28. Использование стандартных имен устройств (консоль, принтер) при работе с текстовыми файлами. Переназначение потоков ввода-вывода.
29. Подпрограммы. Виды подпрограмм. Структура подпрограмм.
30. Использование директивы Forward при описании подпрограмм.
31. Сходства и отличия процедур и функций. Порядок описания и вызова подпрограмм.
32. Механизм передачи параметров. Параметры – значения и параметры – переменные.
33. Типы, определяемые пользователем. Порядок описания и использования.
34. Комбинированный тип. Назначение, порядок описания и использования.
35. Модули. Назначение, порядок подключения и использования.
36. Структура модуля. Назначение областей Interface и Implementation.

7. Материально-техническое обеспечение дисциплины (модуля).

Теоретические и практические занятия должны проводиться в специализированной аудитории, оснащенной современными персональными компьютерами и программным обеспечением в соответствии с тематикой изучаемого материала. Число рабочих мест в аудитории должно быть таким, чтобы обеспечивалась индивидуальная работа студента на отдельном персональном компьютере. Аудитория также должна быть оснащенной современным видеопроектором.

Для обеспечения процесса обучения необходимо использовать помещение, рассчитанное на 12-15 слушателей и соответствующее количество лабораторных компьютеров (один компьютер на каждого учащегося).

Для выполнения практических лабораторных занятий курса требуются компьютеры и периферийное оборудование с установленным программным обеспечением, необходимым для освоения дисциплины: наличие операционных систем Microsoft Windows XP или выше, Delphi

Рекомендуется: выделенное подключение к Интернету для каждого студента, необходимое для контроля за выполнением поиска в Интернете.

8. Лист регистрации изменений

[illegible]