Министерство образования и науки Российской Федерации
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кубанский государственный университет»
факультет математики и компьютерных наук



УТВЕРЖДАЮ
Проректор по учебной работе,
качеству образования – первый
проректор
_____Иванов А.Г.
подпись
30 июня 2017 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.В.20 ПРОГРАММИРОВАНИЕ

Направление подготовки:	44.03.05 Педагогическое образование (с двумя профилями подготовки)
Направленность (профиль):	"Математика, Информатика"
Программа подготовки:	академическая
Форма обучения:	очная
Квалификация:	бакалавр

Краснодар 2017

Рабочая программа дисциплины «Программирование» составлена в соответствии с Федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 44.03.05 Педагогическое образование (с двумя профилями подготовки)

Программу составили:
Попова Г.И., доцент кафедры информационных образовательных технологий, кандидат педагогических наук _____

Рабочая программа дисциплины «Программирование» утверждена на заседании кафедры информационных образовательных технологий
протокол № 11 от 23 мая 2017 г.
Заведующий кафедрой (разработчика) Грушевский С.П. _____

Рабочая программа обсуждена на заседании кафедры информационных образовательных технологий
протокол № 11 от 23 мая 2017 г.
Заведующий кафедрой (выпускающей) Грушевский С.П. _____

Утверждена на заседании учебно-методической комиссии факультета математики и компьютерных наук
протокол № 3 от 20 июня 2017 г.
Председатель УМК факультета Титов Г.Н. Титов

Рецензенты:

Луценко Е.В., доктор экономических наук, профессор кафедры компьютерных технологий и систем КубГАУ

Кособуцкая Е.В., канд. физ.-мат. наук, доцент кафедры вычислительных технологий факультета компьютерных технологий и прикладной математики КубГУ

1. Цели и задачи изучения дисциплины

1.1 Цель дисциплины

Формирование системы понятий, знаний и умений в области современного программирования, включающего в себя методы проектирования, анализа и создания программных продуктов, основанные на использовании объектно-ориентированной методологии.

1.2 Задачи дисциплины:

– формирование знаний, умений и навыков анализа и проектирования математических и информационных моделей реальных объектов или структур;

– овладение умениями и навыками программирования типовых задач обработки информации (вычисления, сортировка, поиск и т.п.) в одной-двух современных средах программирования;

– овладение умениями и навыками использования библиотек объектов (классов) для решения практических задач.

1.3 Место дисциплины в структуре образовательной программы

Дисциплина «Программирование» для бакалавриата по направлению «Педагогическое образование» относится к вариативной части Блока 1 «Дисциплины» учебного плана.

Дисциплина базируется на знаниях, полученных по стандарту общего среднего образования. Изучение дисциплины «Программирование» является базой для дальнейшего освоения студентами дисциплин «Теоретические основы информатики», «Информационные системы», «Математическая логика и теория алгоритмов», «Численные методы», курсов по выбору профессионального цикла, прохождения педагогической практики.

1.4 Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Изучение данной учебной дисциплины направлено на формирование у обучающихся компетенций ОПК-1, ПК-4).

№ п.п.	Индекс компетенции	Содержание компетенции (или её части)	В результате изучения учебной дисциплины обучающиеся должны		
			знать	**уметь**	**владеть**
1.	ОПК-1	готовностью сознавать социальную значимость своей будущей профессии, обладать мотивацией к осуществлению профессиональной деятельности	основные конструкции языка программирования; о парадигмах программирования (императивной, функциональной, логической); о технологиях программирования (структурной, модульной, объектно-ориентированой);	понимать программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня; анализировать алгоритмы с использованием таблиц	терминологией по дисциплине; навыками алгоритмического мышления, формального описания алгоритмов
2.	ПК-4	способностью использовать возможности образовательной среды для достижения личностных, мета-	основные структуры данных и типовые методы обработки этих структур; основные операторы и инструкции	осознанно использовать языки и методы программирования в учебных программах базовых и	навыками структурного программирования, алгоритмизации, работы в раз-

№ п.п.	Индекс компетенции	Содержание компетенции (или её части)	В результате изучения учебной дисциплины обучающиеся должны		
			знать	**уметь**	**владеть**
		предметных и предметных результатов обучения и обеспечения качества учебно-воспитательного процесса средствами преподаваемых учебных предметов	языка программирования	элективных курсов в различных образовательных учреждениях	личных средах программирования (составление, отладка и тестирование программ; разработка и использование интерфейсных объектов)
	ПК-12	способностью руководить учебно-исследовательской деятельностью обучающихся	содержание работы учителя информатики в средней школе, возможности профессионального самопознания и саморазвития с применением компьютерных технологий	использовать в образовательном процессе разнообразные программные и интернет-ресурсы	способами совершенствования профессиональных знаний и умений путем использования возможностей компьютерных технологий

2. Структура и содержание дисциплины

2.1 Распределение трудоёмкости дисциплины по видам работ

Общая трудоемкость дисциплины за 1 курс составляет 5 зачетных единиц, их распределение по видам работ представлено в таблице.

Вид учебной работы	Всего часов	Семестры (часы)	
		1	2
Контактная работа, в том числе:	**106,5**	**56,2**	**50,3**
Аудиторные занятия (всего):	102	54	48
Занятия лекционного типа	34	18	16
Лабораторные работы	68	36	32
Иная контактная работа:	**4,5**	**2,2**	**2,3**
Контроль самостоятельной работы (КСР)	4	2	2
Промежуточная аттестация (ИКР)	0,5	0,2	0,3
Самостоятельная работа, в том числе:	**37,8**	**15,8**	**22**
Курсовая работа	–	–	–
Проработка учебного (теоретического) материала	12	5	7
Выполнение индивидуальных заданий	12	5	7
Подготовка к текущему контролю	13,8	5,8	8
Контроль:	**35,7**	–	**35,7**
Подготовка к экзамену	35,7	**–**	35,7
Общая час.	**180**	**72**	**108**

трудоемкость	**в том числе контактная работа**	**106,5**	**56,2**	**50,3**
	зач. ед.	**5**	**2**	**3**

Общая трудоемкость дисциплины за 2 курс составляет 7 зачетных единиц, их распределение по видам работ представлено в таблице.

Вид учебной работы		Всего часов	Семестры (часы)	
			3	4
Контактная работа, в том числе:		**110,5**	**58,2**	**52,3**
Аудиторные занятия (всего):		102	54	48
Занятия лекционного типа		34	18	16
Лабораторные работы		68	36	32
Иная контактная работа:		**8,5**	**4,2**	**4,3**
Контроль самостоятельной работы (КСР)		8	4	4
Промежуточная аттестация (ИКР)		0,5	0,2	0,3
Самостоятельная работа, в том числе		**105,8**	**49,8**	**56**
Курсовая работа		–	–	–
Проработка учебного (теоретического) материала		35	16	19
Выполнение индивидуальных заданий		35	16	19
Подготовка к текущему контролю		35,8	17,8	18
Контроль:		**35,7**	–	**35,7**
Подготовка к экзамену		35,7	–	35,7
Общая трудоемкость	**час.**	**252**	**108**	**144**
	в том числе контактная работа	**110,5**	**58,2**	**52,3**
	зач. ед.	**7**	**3**	**4**

2.2 Структура дисциплины

Распределение видов учебной работы и их трудоемкости по разделам дисциплины

Разделы дисциплины, изучаемые в **первом** семестре

№ раздела	Наименование разделов	Количество часов			
		Всего	Аудиторная работа		Внеаудиторная работа
			Л	ЛЗ	СРС
1	2	3	4	5	6
1	Типы данных и операторы языка	69,8	18	36	15,8
	Итого:	69,8	18	36	15,8

Примечание: Л – лекции, ПЗ – практические занятия / семинары, ЛР – лабораторные занятия, СРС – самостоятельная работа студента.

Разделы дисциплины, изучаемые во **втором** семестре

№ раздела	Наименование разделов	Количество часов			
		Всего	Аудиторная работа		Внеаудиторная работа
			Л	ЛЗ	СРС
1	2	3	4	5	6

2	Процедуры и функции	23	5	10	8
3	Символьный и строковый типы данных	23	5	10	8
4	Модули	24	6	12	6
	Итого:	**70**	**16**	**32**	**22**

Разделы дисциплины, изучаемые во **третьем** семестре

№ раздела	Наименование разделов	Количество часов			
		Всего	Аудиторная работа		Внеаудиторная работа
			Л	ЛЗ	СРС
1	2	3	4	5	6
5	Файлы	27	4	8	15
6	Указатели и динамическая память	76,8	14	28	34,8
	Итого:	**103,8**	**18**	**36**	**49,8**

Разделы дисциплины, изучаемые в **четвертом** семестре

№ раздела	Наименование разделов	Количество часов			
		Всего	Аудиторная работа		Внеаудиторная работа
			Л	ЛЗ	СРС
1	2	3	4	5	6
7	Объектно-ориентированное программирование	104	16	32	56
	Итого:	**104**	**16**	**32**	**56**

2.3 Содержание разделов дисциплины

2.3.1 Занятия лекционного типа

№ раздела	Наименование раздела	Содержание раздела	Форма текущего контроля
1	2	3	4
1.	**Типы данных, операторы языка**	Система программирования PascalABC.NET. Алфавит и словарь языка Паскаль. Структура программы. Данные. Константы и переменные. Раздел описания переменных. Типы данных. Целые типы. Вещественные типы. Выражения. Операции. Арифметические операции. Математические функции. Операторы. Оператор присваивания. Оператор обращения к процедуре. Процедуры ввода-вывода. Строковые константы. Вывод строковых констант. Форматный вывод данных.	Проверка домашнего задания, опрос, контрольная работа

№ раз-дела	Наимено-вание раздела	Содержание раздела	Форма текущего контроля
1	2	3	4
		Логический (булевский) тип. Операции отношения. Логические операции. Условный оператор If. Оператор выбора Case. Составной оператор. Метки. Оператор безусловного перехода Goto. Пустой оператор. Операторы цикла For, Repeat, While. Алгоритмы обработки последовательностей. Вложенные циклы. Типы данных. Тип диапазон. Массивы. Одномерные массивы. Отладка программ в интегрированной среде PascalABC.NET. Двумерные массивы.	
2.	**Процедуры и функции**	Процедуры и функции. Функции пользователя. Процедуры пользователя. Формальные и фактические параметры. Параметры-значения и параметры-переменные. Область действия идентификаторов при использовании процедур и функций. Локальные и глобальные переменные. Рекурсивные процедуры и функции. Опережающее описание.	Проверка домашнего задания, опрос, контрольная работа
3.	**Символьный и строковый типы данных**	Символьный тип данных. Строковый тип данных. Процедуры и функции для обработки строк. Алгоритмы обработки строк. Перечислимые типы. Тип множество.	Проверка домашнего задания, опрос, контрольная работа
4.	**Модули**	Нисходящее программирование. Модули. Подключение модулей к программе. Стандартные модули. Структура модуля: заголовок, раздел интерфейса, раздел исполнения, раздел инициализации. Модули пользователя. Модуль **GraphABC.** Ресурсы модуля **GraphABC**. Построение динамических изображений. Построение графиков функций.	Проверка домашнего задания, опрос, контрольная работа
5.	**Файлы**	Тип запись (Record). Файлы. Типизированные файлы. Стандартные процедуры и функции обработки типизированных файлов. Текстовые файлы. Стандартные процедуры и функции обработки текстовых файлов.	Проверка домашнего задания, опрос, контрольная работа
6.	**Указатели и динамическая память**	Указатели. Динамическая память. Выделение и освобождение динамической памяти. Динамические структуры данных. Связанные динамические данные. Очереди,	Проверка домашнего задания, опрос, контрольная работа

№ раз-дела	Наимено-вание раздела	Содержание раздела	Форма текущего контроля
1	2	3	4
		списки, стеки, деревья.	
7.	**Объектно-ориенти-рованное програм-мирование**	Язык программирования Python. Интегрированная среда Wing IDE 101. Окно кода программы. Структура программы. Текстовый редактор интегрированной среды. Линейные, разветвляющиеся, циклические алгоритмы. Обработка одномерных и двумерных массивов. Отладка программ. Процедуры. Структуры. Объектно-ориентированная парадигма программирования. Объекты. Инкапсуляция, полиморфизм, наследование. Объектно-ориентированное проектирование. Классы и объекты. События и сообщения. Механизм передачи и обработки, сообщений в объектно-ориентированных средах. Математические объекты: рациональные и комплексные числа, вектора, матрицы. Конструирование объектов: строки, стеки, списки, деревья.	Проверка домашнего задания

2.3.2 Занятия семинарского типа

Занятия семинарского типа - не предусмотрены

2.3.3 Лабораторные занятия

№ раздела	Наименование раздела	Наименование лабораторных работ	Форма текущего контроля
1	2	3	4
1.	**Типы данных, операторы языка**	1. Простейшие программы. 2. Линейные алгоритмы. Вычисление выражений. Стандартные функции. 3. Линейные алгоритмы. Построение простейших изображений. 4. Разветвляющиеся алгоритмы.Условный оператор If. Оператор выбора Case. (4 час.). 5. Контрольная работа «Ветвления». 6. Операторы цикла. Операторы цикла For. Задачи с целыми числами. 7. Операторы цикла. Обработка последовательностей. 8. Операторы цикла Repeat, While. Обработка последовательностей, оканчивающихся нулем. Вычисление суммы ряда . Контрольная работа «Циклы» (6 час.) 9. Алгоритмы обработки одномерных массивов (4 час.).	Проверка домашних заданий, опросы, контрольные работы

№ раздела	Наименование раздела	Наименование лабораторных работ	Форма текущего контроля
1	2	3	4
		10. Алгоритмы сортировки массивов. 11. Контрольная работа «Одномерные массивы». 12. Алгоритмы обработки двумерных массивов (4 час.). 13. Контрольная работа «Двумерные массивы».	
2.	**Процедуры и функции**	Процедуры и функции обработки простых переменных. Процедуры и функции обработки массивов. Контрольная работа «Процедуры и функции».	Проверка домашних заданий, опросы, контрольная работы
3.	**Символьный и строковый типы данных**	Алгоритмы обработки символьных строк. Встроенные процедуры и функции для обработки символьных строк. Контрольная работа «Символьные строки».	Проверка домашних заданий, опросы, контрольная работы
4.	**Модули**	Разработка модулей пользователя. Модуль GraphABC**.** Построение динамических изображений. Построение графиков функций.	Проверка домашних заданий, опросы
5.	**Файлы**	Тип запись (Record). Файлы. Типизированные файлы. Стандартные процедуры и функции обработки типизированных файлов. Текстовые файлы. Стандартные процедуры и функции обработки текстовых файлов.	Проверка домашних заданий, опросы, контрольные работы
6.	**Указатели и динамическая память**	Указатели. Динамическая память. Выделение и освобождение динамической памяти. Динамические структуры данных. Связанные динамические данные. Очереди, списки, стеки, деревья.	Проверка домашних заданий, опросы, контрольные работы
7.	**Объектно-ориентированное программирование**	Язык программирования Python. Линейные, разветвляющиеся, циклические алгоритмы. Обработка одномерных и двумерных массивов. Отладка программ. Процедуры. Структуры. Математические объекты: рациональные и комплексные числа, вектора, матрицы. Конструирование объектов: строки, стеки, списки, деревья.	Проверка домашних заданий, опросы, контрольные работы

2.3.4. Примерная тематика курсовых работ

Курсовые работы не предусмотрены

2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

№	Вид СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	2	3
1.	Проработка лекционного материала	Основная литература, дополнительная литература, периодические издания, ресурсы сети Интернет. Методические указания по выполнению самостоятельной работы, утвержденные кафедрой информационных образовательных технологий, протокол №1 от 31 августа 2017 г.
2.	Чтение и анализ учебной и научной литературы	
3.	Подготовка к лабораторным занятиям	
4.	Подготовка к экзамену	

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ) предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

– в печатной форме увеличенным шрифтом,

– в форме электронного документа,

Для лиц с нарушениями слуха:

– в печатной форме,

– в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

– в печатной форме,

– в форме электронного документа,

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

3. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Семестр	Вид занятия	Используемые интерактивные образовательные технологии	Количество часов
1	Лекции	Интерактивная лекция с мультимедийным оборудованием. Обсуждение сложных и дискуссионных вопросов и проблем.	80
	Лабораторные работы	Компьютерные занятия в режимах взаимодействия «преподаватель – студент», «студент - преподаватель», «студент – студент».	81
Итого:			161

4. Оценочные средства для текущего контроля успеваемости и промежуточной аттестации

4.1 Фонд оценочных средств для проведения текущей аттестации

Вопросы к экзамену, 2 семестр

1. Принципы фон Неймана. Классификация языков программирования.
2. Алгоритм, его свойства, способы записи. Базовые алгоритмические конструкции, виды алгоритмов.
3. Понятие лексики, синтаксиса, семантики языка программирования.
4. Система программирования PascalABC.NET Алфавит и словарь языка Паскаль. Структура программы.

5. Интегрированная среда программирования PascalABC.NET. Компоненты интегрированной среды. Ввод и редактирование текста программы. Сохранение файлов. Выполнение программы. Завершение работы. Система помощи PascalABC.NET.
6. Данные. Константы и переменные. Раздел описания переменных. Типы данных. Целые типы. Вещественные типы.
7. Выражения. Операции. Арифметические операции. Математические функции.
8. Операторы. Оператор присваивания. Оператор обращения к процедуре. Процедуры ввода-вывода.
9. Строковые константы. Вывод строковых констант. Форматный вывод данных.
10. Логический (булевский) тип. Операции отношения. Логические операции, порядок вычисления.
11. Условный оператор If. Полная и краткая форма оператора. Пример.
12. Функции div и mod. Вычисление суммы цифр произвольного целого числа.
13. Оператор выбора Case. Полная и краткая форма оператора. Пример.
14. Составной оператор. Оператор безусловного перехода Goto. Метки. Пустой оператор. Принципы структурного программирования.
15. Оператор цикла For. Программа проверки, является ли число простым. Замена For на While и возможность замены While на For.
16. Оператор цикла Repeat. Программа вычисления суммы ряда.
17. Операторs BREAK, CONTINUE используемые внутри циклов. Оператор EXIT.
18. Типы данных. Тип диапазон. Перечисляемые типы данных.
19. Одномерные массивы. Типовые алгоритмы для одномерных массивов. Пример.
20. Вложенные циклы. Пример использования.
21. Этапы решения задач на ЭВМ. Ручная прокрутка программ. Отладка программ в Паскале.
22. Сортировка массивов. Метод обмена.
23. Двумерные массивы. Типовые алгоритмы для двумерных массивов.
24. Программа вычисления произведения двух заданных матриц.
25. Программа вычисления вектора сумм строк заданной матрицы.
26. Символьный тип данных CHAR. Таблица ASCII.
27. Строковый тип данных STRING, представление в памяти, операции над строками.
28. Стандартные процедуры и функции для типа данных STRING: Pos, Length, Insert, Delete.
29. Строковый тип данных STRING. Программа подсчета количества слов в предложении.
30. Стандартные процедуры и функции для типа данных STRING: Copy,Val, Str, Upcase.
31. Тип множество. Операции над множествами. Пример.
32. Описание и использование процедур в Паскале. Пример.
33. Описание и использование функций в Паскале. Пример.
34. Формальные и фактические параметры в процедуре, правила соответствия. Параметры-значения и параметры-переменные.
35. Область действия идентификаторов при использовании процедур и функций. Локальные и глобальные переменные.
36. Рекурсивные процедуры и функции. Примеры. Опережающее описание.
37. Модули. Подключение модулей к программе. Стандартные модули.
38. Структура модуля: заголовок, раздел интерфейса, раздел исполнения, раздел инициализации. Создание модуля пользователя. Пример.
39. Модуль **GraphABC**, его ресурсы, пример использования.
40. Построение графика функции в Паскале.
41. Анимация в Паскале, пример.
42. Типизированные файлы. Стандартные процедуры и функции обработки типизированных файлов.

43. Текстовые файлы. Стандартные процедуры и функции обработки текстовых файлов.
44. Тип запись (record). Пример использования.
45. Представление в памяти ЭВМ целых чисел.
46. Представление в памяти ЭВМ вещественных чисел.

Примерные задания к экзамену

1. Разложить заданное натуральное число на простые множители.
2. Нахождение минимума и максимума двух, трех, четырех данных чисел без использования массивов и циклов.
3. Нахождение всех корней заданного квадратного уравнения.
4. Нахождение сумм, произведений элементов данной конечной числовой последовательности (или массива).
5. Использование цикла для решения простых переборных задач (поиск наименьшего простого делителя данного натурального числа, проверка числа на простоту и т.д.).
6. Заполнение элементов одномерного и двумерного массивов по заданным правилам.
7. Операции с элементами массива. Линейный поиск элемента. Вставка и удаление элементов в массиве. Перестановка элементов данного массива в обратном порядке. Суммирование элементов массива.
8. Проверка соответствия элементов массива некоторому условию.
9. Нахождение второго по величине (второго максимального или второго минимального) значения в данном массиве за однократный просмотр массива.
10. Нахождение минимального (максимального) значения в данном массиве и количества элементов, равных ему, за однократный просмотр массива.
11. Операции с элементами массива, отобранных по некоторому условию (например, нахождение минимального четного элемента в массиве, нахождение количества и суммы всех четных элементов в массиве).
12. Сортировка массива.
13. Слияние двух упорядоченных массивов в один без использования сортировки.
14. Обработка отдельных символов данной строки. Подсчет частоты появления символа в строке.
15. Работа с подстроками данной строки с разбиением на слова по пробельным символам. Поиск подстроки внутри данной строки, замена найденной подстроки на другую строку.

Вопросы к зачету, 3 семестр

1. Типизированные файлы. Стандартные процедуры и функции обработки типизированных файлов.
2. Типизированные файлы. Пример обработки.
3. Статическая и динамическая память.
4. Адреса и указатели. Структура указателя. Объявление указателей. Типизированные и нетипизированные указатели.
5. Способы задания значения указателя.
6. Операция разыменования.
7. Выделение и освобождение динамической памяти.
8. Несвязанные динамические переменные.
9. Связанные динамические структуры данных.
10. Однонаправленные списки. Структура узла списка. Типовые процедуры обработки списка.
11. Очереди. Типовые процедуры обработки очереди.
12. Стеки. Типовые процедуры обработки стека.
13. Алгоритм проверки правильности скобочного выражения с использованием стека.
14. Алгоритм вычисления выражения в постфиксной записи с использованием стека.

15. Двоичные деревья, структура узла дерева, правила обхода деревьев.
16. Реализация алгоритма обхода двоичного дерева.
17. Алгоритм сортировки с использованием двоичного дерева.
18. Графы. Кратчайшие пути. Алгоритм Дейкстры.
19. Трансляция программ. Интерпретаторы и компиляторы.
20. Упрощенная модель компилятора. Лексический анализ.
21. Упрощенная модель компилятора. Синтаксический анализ.
22. Упрощенная модель компилятора. Генерация кода.
23. Упрощенная модель компилятора. Оптимизация кода.
24. Упрощенная модель компилятора. Ассемблирование.
25. Дисциплина программирования, структурный подход к программированию.
26. Возникновение объектно-ориентированной парадигмы программирования.
27. Объект, его свойства, методы. Пример.
28. Объектно-ориентированное программирование. Инкапсуляция. Пример.
29. Полиморфизм. Процедуры и функции как реализация методов. Пример.
30. Наследование и иерархия объектов. Примеры.
31. Объектно-ориентированное программирование. Конструктор и деструктор. Создание объекта.
32. Событие и сообщение. Обработка событий от клавиатуры и мыши.
33. Основы визуального программирования. Компоненты в VBA.
34. Применение библиотек и иерархий объектов при программировании.

4.2 Фонд оценочных средств для проведения промежуточной аттестации

Вопросы к экзамену, 4 семестр

1. Состав языка PYTHON. Алфавит, идентификаторы, ключевые слова, знаки операций, константы, комментарии.
2. Основные типы данных PYTHON. Структура программы.
3. Переменные, операции и выражения в языке PYTHON.
4. Модули в программе на языке PYTHON, три варианта использования команды подключения модуля import.
5. Ввод и вывод данных в языке PYTHON. Функции input, int, map, split. print (с аргументами sep, end, format).
6. Операторы ветвления в языке PYTHON. Каскадное ветвление (elif). Примеры.
7. Операторы цикла for в языке PYTHON. Пример.
8. Операторы цикла while в языке PYTHON. Пример: определить количество цифр в десятичной записи целого положительного числа.
9. Процедуры и функции в языке PYTHON. Примеры.
10. Одномерные массивы в языке PYTHON (списки, тип данных list). Способы ввода и вывода массива. Стандартные алгоритмы обработки массивов.
11. Алгоритмы обработки массивов с использованием встроенных функций Python. Алгоритм поиска заданного элемента: требуется найти в массиве элемент, равный значению переменной X, или сообщить, что его там нет.
12. Алгоритмы обработки массивов с использованием встроенных функций Python. Алгоритм поиска максимального элемента: требуется найти значение и номер максимального элемента.
13. Алгоритмы обработки массивов с использованием встроенных функций Python. Реверс массива.
14. Алгоритмы обработки массивов с использованием встроенных функций Python. Циклический сдвиг элементов массива на один элемент влево.
15. Алгоритмы обработки массивов с использованием встроенных функций Python. Отобрать все четные элементы массива A в новый массив B.

16. Алгоритмы обработки массивов с использованием встроенных функций Python. Сортировка массива, способы задания ключей сортировки, использование лямбда-функции.
17. Алгоритмы обработки матриц с использованием встроенных функций Python. Ввод, вывод, суммирование матрицы.
18. Концептуальное отличие процедурного программирования от объектно-ориентированного программирования. Преимущества и недостатки решения задачи с помощью ООП.
19. Этапы объектно-ориентированного анализа.
20. Понятия класса и объекта в ООП, различие между ними. Свойства объекта (поля), методы, события.
21. Объявление переменной для работы с объектом некоторого класса. Создание экземпляра класса. Конструктор.
22. Основные принципы ООП. Инкапсуляция. Пример.
23. Основные принципы ООП. Наследование и иерархия классов. Пример.
24. Основные принципы ООП. Полиморфизм. Функции как реализация методов. Пример.
25. Абстрактные классы. Пример.
26. Интерфейс объекта. Скрытие внутреннего устройства объекта. Открытые (public) и закрытые (protected).
27. Доступ к полям объекта с помощью методов. Пример.
28. Интегрированная среда программирования Wing IDE 101.
29. Среда, управляемая событиями. Событие и сообщение. Передача сообщений между объектами. Пример.
30. Математические объекты: рациональные числа. Описание их свойств и методов в PYTHON.
31. Математические объекты: комплексные числа. Описание их свойств и методов в PYTHON.
32. Математические объекты: вектора. Описание их свойств и методов в PYTHON.
33. Математические объекты: матрицы. Описание их свойств и методов в PYTHON.

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

– при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;

– при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических– при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине (модулю) предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

– в печатной форме увеличенным шрифтом,

– в форме электронного документа.

Для лиц с нарушениями слуха:

– в печатной форме,

– в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

– в печатной форме,

– в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

5. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

5.1 Основная литература:

1. Окулов, С.М. Основы программирования [Электронный ресурс] : учебное пособие / С.М. Окулов. — Электрон. дан. — Москва : Издательство "Лаборатория знаний", 2015. — 339 с. — Режим доступа: https://e.lanbook.com/book/66119.
2. Буйначев С. К., Боклаг Н. Ю. Основы программирования на языке Python: учебное пособие. Издательство Уральского университета, 2014. 92 с. [Электронный ресурс, ЭБС «Университетская библиотека ONLINE»], URL: http://biblioclub.ru/index.php?page=book_red&id=275962
3. Долинер Л. И. Основы программирования в среде PascalABC.NET: учебное пособие. Екатеринбург. Изд-во Урал. ун-та. 2014. – 129 с. [Электронный ресурс, ЭБС «Университетская библиотека ONLINE»], URL: http://biblioclub.ru/index.php?page=book_red&id=275988.

5.2 Дополнительная литература:

4. Огнева, М. В. Программирование на языке С++: практический курс : учебное пособие для бакалавриата и специалитета / М. : Издательство Юрайт, 2018. – 335 с. [Электронный ресурс]. URL: https://biblio-online.ru/book/7670D7EC-AC37-4675-8EAE-DD671BC6D0E4/programmirovanie-na-yazyke-s-prakticheskiy-kurs.
5. Алексеев Е.Р, Чеснокова О., Кучер Т. Программирование на Free Pascal и Lazarus / - 2-е изд., исправ. - Москва: Национальный Открытый Университет «ИНТУИТ», 2016. - 552 с.: ил. [Электронный ресурс]. URL: https://biblioclub.ru/index.php?page=book_red&id=429189&sr=1.
6. Абрамян М.Э. Практикум по программированию на языке Паскаль: массивы, строки, файлы, рекурсия, линейные динамические структуры, бинарные деревья : учебное пособие / Изд. 7-е, перераб. и доп. - Ростов-на-Дону: Издательство Южного федерального университета, 2010. - 277 с. [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=240952.
7. Окулов, С.М. Программирование в алгоритмах [Электронный ресурс] / С.М. Окулов. — Электрон. дан. — Москва : Издательство "Лаборатория знаний", 2017. — 386 с. — Режим доступа: https://e.lanbook.com/book/94140 .
8. Северенс Ч. Введение в программирование на Python [Электронный ресурс] / Ч. Северенс. - 2-е изд., испр. - М.: Национальный Открытый Университет «ИНТУИТ», 2016. - 231 с. URL: https://biblioclub.ru/index.php?page=book&id=429184.
9. Зыков, С. В. Программирование. Объектно-ориентированный подход : учебник и практикум для академического бакалавриата. [Электронный ресурс] М.: Издательство Юрайт, 2018. — 155 с. URL: https://biblio-online.ru/book/programmirovanie-obektno-orientirovannyy-podhod-414203.
10. Гниденко И. Г., Павлов Ф. Ф., Федоров Д. Ю. Технологии и методы программирования: учебное пособие для прикладного бакалавриата / М. : Издательство Юрайт, 2018. – 235 с. – [Электронный ресурс]. - URL: https://biblio-online.ru/book/tehnologii-i-metody-programmirovaniya-413762.

5.3 Периодические издания

1. Журнал «Информатика и образование».
2. Журнал «Информатика в школе».

6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. Современное программирование на языке Паскаль. URL http://pascalabc.net/.
2. Веб-среда разработки ProgrammingABC.NET WDE (www.pascalabc.net/WDE).
3. www.freepascal.org.
4. http://www.pascal-central.com/.
5. К. Поляков. Язык Python. URL: // http://kpolyakov.spb.ru/school/probook/python.htm.
6. Поляков К.Ю. Газета «Информатика». Язык Python глазами учителя. – М.: Первое сентября, 2014 – № 9 – С. 4–16 // URL:http://kpolyakov.spb.ru/download/inf-2014-09a.pdf.
7. К. Поляков. Язык Python: избранные алгоритмы. Часть 1 // Информатика, № 9, 2014, с. 18-26. // URL: http://kpolyakov.spb.ru/download/inf-2014-09b.pdf/
8. К. Поляков. Объектно-ориентированное программирование // URL: http://kpolyakov.spb.ru/school/c.htm.
9. Кириенко Д.П. Программирование на Python [Электронный ресурс] URL: http://server.179.ru/wiki/?page=DenisKirienko/Python.
10. Кириенко Д.П. Программирование на Python [Электронный ресурс] URL: http://informatics.mccme.ru/course/view.php?id=156
11. Сайт дистанционной подготовки по информатике http://informatics.mccme.ru/moodle/.
12. Язык программирования Python. Учебное пособие http://www.intuit.ru/studies/courses/49/49/info.
13. Столяров А.В. Введение в язык C++. http://www.stolyarov.info/books/pdf/cppintro3.pdf.

7. Методические указания для обучающихся по освоению дисциплины

На лекциях преподаватель рассматривает вопросы программы курса, составленной в соответствии с государственным образовательным стандартом. Из-за недостаточного количества аудиторных часов некоторые темы не удается осветить в полном объеме, поэтому преподаватель некоторые вопросы выносит на самостоятельную работу студентов, рекомендуя ту или иную литературу.

Для освоения материала и систематизации знаний по дисциплине необходимо постоянно разбирать материалы лекций по конспектам и учебным пособиям и в случае необходимости обращаться к преподавателю за консультацией. Полный список источников по дисциплине приведен в п. 5, 6.

После изучения некоторых разделов курса «Программирование» на лабораторных занятиях проводятся контрольные аудиторные работы. Для успешного их написания,готовиться к контрольным работам нужно по материалам лекций, презентаций и рекомендованных источников.

В освоении дисциплины инвалидами и лицами с ограниченными возможностями здоровья большое значение имеет индивидуальная учебная работа (консультации) – дополнительное разъяснение учебного материала.

Индивидуальные консультации по предмету являются важным фактором, способствующим индивидуализации обучения и установлению воспитательного контакта между преподавателем и обучающимся инвалидом или лицом с ограниченными возможностями здоровья.

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю) (при необходимости)

8.1 Перечень информационных технологий

Мультимедийные лекции; демонстрационные примеры программ; использование компьютера при выдаче заданий и проверке их решений, выполнении лабораторных работ; компьютерное тестирование по итогам изучения разделов дисциплины.

8.2 Перечень необходимого программного обеспечения

1. Операционная система MS Windows.
2. Интегрированное офисное приложение MS Office.
3. Программное обеспечение для организации управляемого коллективного и безопасного доступа в Интернет.
4. Среда программирования PascalABC.Net.
5. Среда программирования на языке Python – Wing IDE 101.

8.3 Перечень информационных справочных систем:

Электронная библиотечная система eLIBRARY.RU (http://www.elibrary.ru)/

Электронная библиотечная система "Университетская библиотека онлайн" (https://biblioclub.ru/)

Электронная библиотечная система издательства "Лань" https://e.lanbook.com

Электронная библиотечная система "Юрайт" http://www.biblio-online.ru/

9. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

№	Вид работ	Материально-техническое обеспечение дисциплины (модуля) и оснащенность
1	Лекционные занятия	Лекционная аудитория, специально оборудованная мультимедийными демонстрационными комплексами, учебной мебелью
2	Семинарские занятия	Специальное помещение, оснащенное учебной мебелью, презентационной техникой (проектор, экран, ноутбук) и соответствующим программным обеспечением (ПО).
3	Лабораторные занятия	Помещение для проведения лабораторных занятий оснащенное учебной мебелью, персональными компьютерами с доступом к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду организации
4	Групповые (индивидуальные) консультации	Помещение для проведения групповых (индивидуальных) консультаций, учебной мебелью, оснащенное презентационной техникой (проектор, экран, ноутбук) и соответствующим программным обеспечением
5	Текущий контроль, промежуточная аттестация	Помещение для проведения текущей и промежуточной аттестации, оснащенное учебной мебелью, презентационной техникой (проектор, экран, ноутбук) и соответствующим программным обеспечением
6	Самостоятельная работа	Кабинет для самостоятельной работы, оснащенный компьютерной техникой с возможностью подключения к сети «Интернет», программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета