МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«ИРКУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ФГБОУ ВО «ИГУ»
Кафедра алгебраических и информационных систем

«Утверждаю»
Директор ИМЭИ ИГУ
_____Фалалеев М.В.
«30» марта 2016 г.

Рабочая программа дисциплины (модуля)

Б1.Б.19 Информатика и программирование

Направление подготовки **01.03.02 «Прикладная математика и информатика»**

Тип образовательной программы: **прикладной бакалавриат**

Направленность (профиль) **«Математическое и компьютерное моделирование в технике и экономике, методы принятия решений»**

Квалификация (степень) выпускника – БАКАЛАВР

Форма обучения **очная**

Иркутск 2016 г.

Согласовано с УМК ИМЭИ	Рекомендовано кафедрой
Протокол № 1	Протокол № 7
от 11 февраля 2016 г.	от 2 февраля 2016 г.
Председатель _____ Рожина Л.В.	Ио зав. кафедрой
	_____ Манцивода А.В.

Содержание

		стр.
1.	Цели и задачи дисциплины (модуля)	4
2.	Место дисциплины (модуля) в структуре ОПОП.	4
3.	Требования к результатам освоения дисциплины (модуля)	4
4.	Объем дисциплины (модуля) и виды учебной работы	5
5.	Содержание дисциплины (модуля)	5
	5.1 Содержание разделов и тем дисциплины (модуля)	
	5.2 Разделы и темы дисциплин (модулей) и виды занятий	
6.	Перечень семинарских, практических занятий, лабораторных работ, план самостоятельной работы студентов, методические указания по организации самостоятельной работы студентов	9
7.	Примерная тематика курсовых работ (проектов) (при наличии)	10
8.	Учебно-методическое и информационное обеспечение дисциплины (модуля):	11
	а) основная литература;	
	б) дополнительная литература;	
	в) программное обеспечение;	
	г) базы данных, поисково-справочные и информационные системы	
9.	Материально-техническое обеспечение дисциплины (модуля)	11
10.	Образовательные технологии	11
11.	Оценочные средства (ОС)	11

1. Цели и задачи дисциплины:

Цель дисциплины

Целью преподавания дисциплины «Информатика и программирование» является изучение основных понятий информатики и программирования на одном из языков высокого уровня.

Задачи дисциплины

Задачами преподавания дисциплины «Информатика и программирования» являются:

1) ознакомление с основными понятиями и методами информационных технологий;
2) ознакомление с основами языка программирования Python;
3) формирование устойчивого алгоритмического мышления у студентов.

2. Место дисциплины в структуре ОПОП:

Дисциплина входит в цикл общепрофессиональных дисциплин. Для изучения и освоения дисциплины требуются только знания по школьной программе из математики и информатики. Знания полученные при изучении этой дисциплины могут быть использованы в следующих дисциплинах и практиках:

Б1.В.ОД.5	Численные методы
Б1.В.ОД.6	Вычислительные системы и компьютерные сети
Б1.В.ДВ.3.1	Технологии программирования
Б1.В.ДВ.3.2	Языки и системы программирования
Б1.В.ДВ.5.1	Основы информационной безопасности
Б1.В.ДВ.5.2	Программные средства защиты информации
Б1.В.ОД.3	Компьютерное моделирование в технике
Б1.В.ОД.4	Имитационное моделирование в технике
Б1.В.ДВ.4.1	Базы данных и СУБД
Б1.В.ДВ.4.2	Алгоритмы и структуры данных
Б1.В.ДВ.7.1	Компьютерные системы проектирования
Б1.В.ДВ.7.2	Методы оптимального проектирования
Б2.У.1	Учебная практика - По получению первичных профессиональных умений и навыков (ознакомительная)
Б2.У.2	Учебная практика - По получению первичных профессиональных умений и навыков (Вычислительная практика)
Б2.П.1	Производственная практика - По получению профессиональных умений и опыта профессиональной деятельности
Б2.П.2	Производственная практика - Преддипломная

3. Требования к результатам освоения дисциплины:

Процесс изучения дисциплины (модуля) направлен на формирование следующих компетенций:

ОПК-3 (способностью к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям).

В результате изучения дисциплины студент должен:
Знать: основные понятия информатики; методы разработки алгоритмов решения математических и прикладных задач; синтаксис и семантику языка программирования Python.
Уметь: разрабатывать алгоритмы решения математических задач и реализовать их на языке Python.
Владеть: навыками работы на компьютере, оперирования с файлами, использования прикладного и системного программного обеспечения; офисными программами Word и Excel; навыками программирования на языке Python.

4. Объем дисциплины (модуля) и виды учебной работы (разделяется по формам обучения)

Вид учебной работы	Всего часов / зачетных единиц	Семестры			
		1	2		
Аудиторные занятия (всего)	146	74	72		
В том числе:	-	-	-	-	-
Лекции	66	30	36		
Практические занятия (ПЗ)	14	14			
Семинары (С)					
Лабораторные работы (ЛР)	66	30	36		
Самостоятельная работа (всего)	55	27	28		
В том числе:	-	-	-	-	-
Курсовой проект (работа)					
Расчетно-графические работы					
Реферат (при наличии)					
Другие виды самостоятельной работы					
Контроль самостоятельной работы	15	7	8		
Вид промежуточной аттестации (зачет, экзамен)	экзамен (2)	экзамен	экзамен		
Контроль	72	36	36		
Контактная работа (всего)	161	81	80		
Общая трудоемкость часы	288	144	144		
зачетные единицы	8	4	4		

5. Содержание дисциплины

5.1. Содержание разделов и тем дисциплины

1. Информатика

1.1. Основные понятия информатики

Автоматизация обработки информации. Программы. Данные. Файл. Память. Алгоритм. Команда. Программирование. Язык программирования. Парадигмы программирования.

1.2. Файловая система

Понятие файла. Операционные системы и файловые системы. Файловые менеджеры. Свойства файла. Папки. Жёсткие и символические ссылки. Связи. Элементы файловых систем. Носители. Специальные устройства ввода-вывода. Шаблоны имён файлов. Типы файлов. Mime-стандарт. Операции над файлами.

1.3. Системное программное обеспечение

Операционная система. Назначение. Интерфейс. Команды ОС. Переменные

окружения. Перенаправление ввода, вывода. Конвейеры. Пакетная обработка.

1.4. Файловые менеджеры

Проводник. Файловый менеджер Far. Навигация по файловой системе. Операции с файлами в Far. Архивация и работа с архивами. Настройки Far на определённые типы файлов. Клавиатурные макросы в Far.

1.5. Текстовый редактор Word

Средства форматирования текстов в Word. Понятие стиля. Использование и разработка собственных стилей. Шаблоны. Макросы.

1.6. Табличный процессор Excel

Типы данных в Excel. Концепция электронной таблицы. Функции рабочего листа. Разработка таблиц для решения прикладных задач. Макросы.

2. Программирование

2.1. Основные характеристики и особенности языка программирования Python

Интерпретируемые и компилируемые языки программирования. Реализации Python. Интегрированная среда разработки. Дистрибутивы.

2.2. Основы языка Python

Концепция интерпретируемого языка. Среда разработки Spider. Синтаксис языка Python. Концепция типов данных. Вывод и ввод данных. Средства отладки.

2.3. Типы данных языка Python

Простые типы данных. Числа: целые, вещественные и комплексные. Последовательности. Списки, кортежи. Строки.

2.4. Основные операторы языка Python

Проверка условий. Циклы for и while. Операторы continue и break.

2.5. Функции. Рекурсия

Определений функций в Python. Концепция чёрного ящика. Глобальные и локальные переменные. Побочные эффекты функции. Аргументы и параметры. Значения аргументов по умолчанию. Задание имён аргументам. Рекурсия. Взаимозаменяемость рекурсии и итерации. Определение функции как оператор.

2.6. Работа со списками

Функции над списками и методы списков. Двумерные списки. Задачи нахождения минимального элемента, сортировки и поиска.

2.7. Множества и словари

Тип данных dict. Запись словарей. Списки ключей и значений. Множество. Преобразование множества в список. Задачи, требующие использования словарей и множеств.

2.8. Файлы. Обработка исключительных ситуаций

Файл как тип данных. Ввод и вывод из файла. Обработка и создание исключительных ситуаций. Оператор assert.

2.9. Модули.

Функции и модули. Подключение модулей. Модули os, sys, math, time.

2.10. Использование графики

Модуль turtle. Модуль tkinter. Организация пользовательского интерфейса. Построение графиков функций.

2.11. Объектно-ориентированное программирование на Python

Концепция объектов. Классы объектов. Объектная декомпозиция. Определение классов в Python. Конструктор и деструктор. Поля и методы объектов. Поля классов. Наследование. Переопределение методов. Множественное наследование. Полиморфизм. Инкапсуляция. Свойства property. Статические методы.

2.12. Коллекции и итераторы

Понятие коллекции. Стандартные коллекции. Создание коллекций. Реализация работы итератора над коллекцией. Итераторы без коллекции.

2.13. Функциональное программирование на Python

Функциональная форма записи. Композиция как основной способ построения функций. Функции высшего порядка. Ламбда-выражения. Карринг. Генераторы.

2.14. Технология разработки и отладки программ

Разработка программ сверху вниз и снизу вверх. Работа с отладчиком и профайлером. Работа в команде.

5.2. Разделы и темы дисциплин (модулей) и виды занятий

№ п/п	Наименование раздела	Наименование темы	Виды занятий в часах					
			Лекц.	Практ. зан.	Семин	Лаб. зан.	СРС	Всего
1.	Информатика	Основные понятия информатики	2	2				4
2.	Информатика	Файловая система	4	2		4		10
3.	Информатика	Системное программное обеспечение	4	4		2		10
4.	Информатика	Файловые менеджеры	4	2		4		10
5.	Информатика	Текстовый редактор Word	2	2		4	10	18
6.	Информатика	Табличный процессор Excel	2	2		6	10	20
7.	Программирование	Основные характеристики и особенности языка программирования Python	1				7	8
8.	Программирование	Основы языка Python	1			2		3
9.	Программирование	Типы данных языка Python	3			4		7
10.	Программирование	Основные операторы языка Python	2			2		4

11.	Программирование	Функции. Рекурсия	2			2		4
12.	Программирование	Работа со списками	6			6		12
13.	Программирование	Множества и словари	6			6		12
14.	Программирование	Файлы. Обработка исключительных ситуаций	2			2		4
15.	Программирование	Модули	2				10	12
16.	Программирование	Использование графики	4			4		8
17.	Программирование	Объектно-ориентированное программирование на Python	8			10		18
18.	Программирование	Коллекции и итераторы	4			4		8
19.	Программирование	Функциональное программирование на Python	5			4	10	19
20.	Программирование	Технология разработки и отладки программ	2				8	10

6. Перечень семинарских, практических занятий и лабораторных работ

№ п/п	№ раздела и темы дисциплины (модуля)	Наименование семинаров, практических и лабораторных работ	Трудоемкость (час.)	Оценочные средства	Формируемые компетенции
1	**2**	**3**	**4**	**5**	**6**
1.	1.1	Основные понятия информатики	2	Текущ. контроль	ОПК-3
2.	1.2	Файловая система	6	Текущ. контроль	ОПК-3
3.	1.3	Системное программное обеспечение	6	Текущ. контроль	ОПК-3
4.	1.4	Файловые менеджеры	6	Текущ. контроль	ОПК-3
5.	1.5	Текстовый редактор Word	6	Текущ. контроль	ОПК-3
6	1.5	Табличный процессор Excel	8	Текущ. контроль	ОПК-3
7.	2.2	Основные характеристики и особенности языка программирования Python	2	Текущ. контроль	ОПК-3
8.	2.3	Типы данных языка Python	4	Текущ. контроль	ОПК-3
9.	2.4	Основные операторы языка Python	2	Текущ. контроль	ОПК-3
10.	2.5	Функции. Рекурсия	2	Текущ. контроль	ОПК-3
11.	2.6	Работа со списками	6	Текущ. контроль	ОПК-3
12.	2.7	Множества и словари	6	Текущ. контроль	ОПК-3
13.	2.8	Файлы. Обработка исключительных ситуаций	2	Текущ. контроль	ОПК-3
14.	2.10	Использование графики	4	Текущ. контроль	ОПК-3
15.	2.11	Объектно-ориентированное программирование на Python	10	Текущ. контроль	ОПК-3
16.	2.12	Коллекции и итераторы	4	Текущ. контроль	ОПК-3
17.	2.13	Функциональное программирование на Python	6	Текущ. контроль	ОПК-3
18.	2.14	Технология разработки и отладки программ	2	Текущ. контроль	ОПК-3

6.1. План самостоятельной работы студентов

№ нед.	Тема	Вид самостоятельной работы	Задание	Рекомендуемая литература	Количество часов
1	Текстовый редактор Word	контрольное задание		ОЛ-2	10
2	Табличный процессор Excel	контрольное задание		ОЛ-2	10
3	Основные характеристики и особенности языка программирования Python	контрольное задание		ОЛ-1	7
4	Модули	контрольное задание		ОЛ-1	10
5	Функциональное программирование на Python	контрольное задание		ОЛ-1	10
6	Технология разработки и отладки программ	контрольное задание		ОЛ-1	8

6.2. Методические указания по организации самостоятельной работы студентов

7. Примерная тематика курсовых работ (проектов) (при наличии)

Не предусмотрены.

8. Учебно-методическое и информационное обеспечение дисциплины (модуля):

а) основная литература

1. Буйначев, С.К. Основы программирования на языке Python : учеб. пособие / Н.Ю. Боклаг, С.К. Буйначев .— Екатеринбург : Издательство Уральского университета, 2014 .— 94 с. — ISBN 978-5-7996-1198-9— Режим доступа: https://rucont.ru/efd/292806 — Неогранич. доступ.

2. Информатика [Текст] : учеб. пособие для студ. вузов, обуч. по спец. гр. "Экономика и управление" и напр. "Информатика и вычислит. техника" / О. П. Новожилов; Московский гос. индустриальный ун-т. - 2-е изд., испр. и доп. - М. : Юрайт, 2012. - 564 с. : ил. ; 21 см. - (Бакалавр). - Библиогр.: с. 562-564. – ISBN 978-5-9916-1756-7 : 550.60 р. (50 экз.)

б) дополнительная литература

2. Основы программирования [Текст] : [Учеб. пособие] / В. В. Борисенко ; Моск. гос. ун-т им. М.В. Ломоносова. - М. : Интернет-Ун-т информ. технологий, 2005. - 314 с. : ил. ; 22 см. - (Основы информатики и математики). - Библиогр.: с. 309-310. - Предм. указ.: с. 311-314. – ISBN 5-9556-0039-6 : 240.00 р., 246.82 р., 175.00 р. (13 экз.)

в) программное обеспечение

Microsoft Windows, Microsoft Word (Office 365), Microsoft Excel (Office 365), Far Manager 3.0, Anaconda.

г) базы данных, информационно-справочные и поисковые системы

9. Материально-техническое обеспечение дисциплины (модуля):

Компьютерный класс, оборудованный персональными компьютерами.

10. Образовательные технологии:

- лекционно-семинарская форма обучения;
- активная форма обучения;
- разноуровневое обучение.

Разработчик: к.ф.-м.н., доцент В.А. Петухин

Приложение

11. Оценочные средства (ОС):

11.1. Оценочные средства для входного контроля

Оценочные средства для входного контроля не предусмотрены.

11.2. Оценочные средства текущего контроля

Оценочные средства разработаны в рамках балльно-рейтинговой системы. В течение семестра студенты выполняют задания, целью которых является как закрепление теоретических знаний, так и освоение конкретных методов кодирования и декодирования.

Перечень заданий, в ходе выполнения которых студенты получают баллы:

Файловая система:

1. Восстановление расширений файлов.

Системное программное обеспечение:

2. Создание жёстких ссылок и связей с папками и файлами.
3. Сравнение содержимого папок.

Файловые менеджеры:

4. Настройка файлового менеджера Far на типы файлов.
5. Клавиатурные макросы в Far Manager.

Текстовый редактор Microsoft Word:

6. Разработка собственных стилей.

Табличный процессор Microsoft Excel:

7. Решение задач с использованием инструментария таблиц.

Язык программирования Python:

8. Простые типы данных языка Python.

9. Проверка условий.
10. Цикл.
11. Кортежи, списки.
12. Функции.
13. Обработка списков.
14. Двумерные списки.
15. Словари.
16. Множества.
17. Ввод-вывод из файлов.
18. Модули.
19. Графика. Построение графиков функций.
20. Объектно-ориентированное программирование. Создание классов.
21. Объектно-ориентированное программирование. Наследование.
22. Объектно-ориентированное программирование. Инкапсуляция.
23. Контейнеры и итераторы.
24. Рекурсия.
25. Функциональное программирование. Функции высших порядков.
26. Лямбда-функции. Частичное применение.

Технология разработки и отладки программ:

27. Отладка программ.

Полный фонд оценочных средств содержится по адресам:

http://domic.isu.ru/instructor/course/informat2_pmi/

http://domic.isu.ru/instructor/course/2informat2_pmi/

Примеры заданий:

Задания по теме «Файловая система»:

1. В вашем распоряжении файлы, у которых утрачены расширения. Вам надо восстановить расширения по содержимому файлов. Файлы с потерянными расширениями находятся в zip-архиве. Извлеките файлы из архива, восстановите расширения, заархивируйте заново и пошлите результат на проверку.

Задания по теме «Текстовый редактор Microsoft Word. Разработка собственных стилей*»:*

2. Разработайте стили для Word для различных элементов текста:

   - заголовков глав, разделов, подразделов,
   - примечаний, отделяемых от текста горизонтальными линиями,
   - фрагментов-предостережений, заключённых в прямоугольную рамку,
   - фрагментов кода, для которых обязательно используется равноширинный текст,
   - обычного текста.

   В стили включите как задание оформления шрифтов (размер, гарнитура, цвет и т.п.), так и свойств абзаца (отступы и т.п.), там, где это имеет смысл и стиль следующего абзаца. В качестве примера можно взять оформление книги "A Byte of Python".
   Разработанные стили включите в шаблон. Задайте в шаблоне также колонтитулы:

верхний должен содержать название всего текста, нижний для чётных страниц - номер страницы и название главы, для нечётных - номер страницы и название раздела.
На основе Вашего шаблона создайте файл с примером текста (содержащий все перечисленные элементы).

Задания по теме «Табличный процессор Microsoft Excel»:

3. Создайте таблицу для учёта посещаемости и итогов промежуточной аттестации студенческой группы. Студентов разместите по строкам, по столбцам - занятия. Запись значения "н" в соответствующую ячейку помечает студента как отсутствующего. В первых слобцах таблицы (после столбца со списком студентов) поместите данные промежуточной аттестации. В таблице для каждого студента должен подсчитываться средний балл по всем предметам, количество пропусков, количество предметов, по которым он не аттестован.
   Предусмотрите возможности (на Ваш выбор) отслеживать студентов, у которых:
   a. больше 20 пропусков,
   b. по крайней мере по двум предметам нет аттестации.
   Отсортируйте студентов по среднему баллу аттестации.

Задания по теме «Кортежи, списки»:

4. Дано целое число N (> 0). Сформировать и вывести целочисленный список размера N, содержащий степени двойки от первой до N-й: 2, 4, 8, 16, ...
5. Дан целочисленный список размера N (размерность списка вводится с клавиатуры, элементы списка заполняются случайными целыми числами). Вывести его элементы в обратном порядке.
6. Дан список размера N и целые числа K и L (0 <= K <= L < N). Найти сумму элементов списка с номерами от K до L включительно.
7. Даны списки A и B одинакового размера N (элементы списков заполняются случайными целыми числами). Поменять местами их содержимое и вывести вначале элементы преобразованного списка A, а затем - элементы преобразованного списка B.
8. Дан список A размера N (элементы списка заполняются случайными целыми числами). Поменять местами его минимальный и максимальный элементы.

Задания по теме «Двумерные списки»:

9. Дан список размерности M на N. Найти первый максимальный элемент двумерного списка и его индексы.
10. Определить является ли двумерный список a[n][n] симметричным относительно своей главной диагонали.

Задания по теме «Словари»:

11. Шифр Цезаря получается путем сдвига алфавита на n позиций. Это число является ключом. Например, для n=3:
    ```
    АБВГДЕЖЗ...ЭЮЯ
    ГДЕЖЗИЙК...АБВ
    ```
    a. Написать функцию, которая возвращает словарь для каждого символа.

    ```
    def cypher(n)
    dict3 = {
           'А': 'Г',
           'Б': 'Д',
       ...
    }
    ```

b. Написать функцию, которая зашифровывает сообщение:
   `def encrypt(text)`

Задания по теме «Объектно-ориентированное программирование. Создание классов»:

12. Определить класс "Вектор размерности n". Реализовать методы сложения, вычитания, скалярного произведения, модуля вектора.
    a. Создать список из m объектов.
    b. Каждую из пар векторов передать в метод, возвращающие их скалярное произведение.
    c. Найти и вывести два наиболее близких по значению вектора из списка (два вектора, модель разности которых минимален).
    d. Вычислить и вывести углы между двумя заданными векторами.
13. Определить класс "Дробь" в виде пары (m,n). Реализовать методы для сложения, вычитания, умножения и деления дробей.
    a. Создать список из m дробей.
    b. Ввести/вывести значения для списка дробей.
    c. От всех элементов списка с нечетными индексами отнять предыдущий элемент списка.

11.3. Оценочные средства для промежуточной аттестации (в форме экзамена)

Выполняя 13 заданий, оцениваемых средствами текущего контроля в первом семестре, студенты могут получить 130 баллов (по 10 баллов за задание), во втором семестре – 140 баллов (за 14 заданий). Ещё 50 баллов можно набрать за экзамен. Количество баллов от 140 до 180 в первом семестре (во втором от 150 до 190) соответствует оценке «5», от 100 до 140 (110-150 во втором семестре) – «4», от 70 до 100 (80-110 во втором семестре) – «3».

Примерный перечень вопросов и заданий к экзамену

1. Основные понятия информатики
2. Файловая система
3. Системное программное обеспечение
4. Файловые менеджеры
5. Текстовый редактор Word
6. Табличный процессор Excel
7. Основные характеристики и особенности языка программирования Python
8. Основы языка Python
9. Типы данных языка Python
10. Основные операторы языка Python
11. Функции. Рекурсия
12. Работа со списками
13. Множества и словари
14. Файлы. Обработка исключительных ситуаций
15. Модули
16. Использование графики
17. Объектно-ориентированное программирование на Python
18. Коллекции и итераторы
19. Функциональное программирование на Python
20. Технология разработки и отладки программ