

Белорусский государственный университет

УТВЕРЖДАЮ

Проректор по учебной работе и
образовательным инновациям


О. И. Чуприс
(И.О.Фамилия)

(подпись)

16.07.2018
(дата утверждения)

Регистрационный № УД- 6452 /уч.

ПРИКЛАДНОЕ ПРОГРАММИРОВАНИЕ

**Учебная программа учреждения высшего образования
по учебной дисциплине для специальностей:**

1-31 80 07 Радиофизика;

1-31 80 08 Физическая электроника;

1-98 80 03 Аппаратное и программно-техническое обеспечение

2018 г.

Учебная программа составлена на основе ОСВО 1-31 80 07-2012, ОСВО 1-31 80 08-2012, ОСВО 1-98 80 03 -2012 и учебных планов УВО №G31-284/уч., УВО №G31-285/уч. от 26.05.2017 г. и УВО № P98-335/уч. от 24.05.2018.

СОСТАВИТЕЛИ:

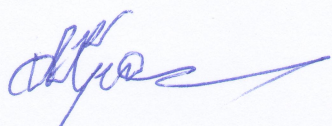
Евгения Владимировна Лисица, старший преподаватель кафедры системного анализа и компьютерного моделирования Белорусского государственного университета

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой системного анализа и компьютерного моделирования Белорусского государственного университета

(протокол № 13 от 29 мая 2018 г.);

Научно-методическим Советом Белорусского государственного университета (протокол № 7 от 13 июля 2018 г.).



ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Учебная программа дисциплины компонента учреждения высшего образования цикла дисциплин специальной подготовки «Прикладное программирование» разработана в соответствии с требованиями ОСВО по специальностям 1-31 80 07 «Радиофизика», 1-31 80 08 «Физическая электроника», 1-98 80 03 «Аппаратное и программно-техническое обеспечение». Программа предназначена для магистрантов дневной формы получения высшего образования.

Цель преподавания дисциплины – изучение теоретических основ обработки данных, включая базовые элементы обработки изображений, интеллектуального анализа данных с использованием языка Python.

Задача дисциплины – научить производить расчеты с применением технологий статистического анализа, обработки изображений, анализа больших данных, интеллектуального анализа данных на языке программирования Python.

Учебная программа составлена с учетом межпредметных связей и программ по дисциплинам «Программирование», «Теория вероятностей и математическая статистика», «Математическое моделирование», «Интеллектуальный анализ данных», «Цифровая обработка сигналов».

Изучение учебной дисциплины «Прикладное программирование» должно обеспечить формирование у студента следующих компетенций:

Академические компетенции:

АК-1. Способность к самостоятельной научно-исследовательской деятельности, готовность генерировать и использовать новые идеи.

АК-2. Методологические знания и исследовательские умения, обеспечивающие решение задач научно-исследовательской, управленческой и инновационной деятельности.

АК-3. Способность самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности знания и умения и углублять свое научное мировоззрение.

Профессиональные компетенции:

ПК-3. Осваивать и внедрять в учебный процесс инновационные образовательные технологии

ПК-7. Работать с научно-технической информацией с использованием современных информационных технологий.

ПК-10. Обосновывать достоверность полученных научных результатов.

ПК-11. Формулировать выводы и рекомендации по применению результатов научно-исследовательской работы.

ПК-13. Составлять отчеты и презентации о научно-исследовательской работе, участвовать в работе научных конференций.

ПК-20. Определять цели инноваций и способы их достижения.

В результате изучения дисциплины студент должен:

знать:

- базовые понятия и принципы анализа данных в прикладных задачах,
- основные алгоритмы анализа изображений, интеллектуального анализа данных, статистической обработки данных,
- задачи прикладного анализа данных;

уметь:

- производить расчеты с применением алгоритмов анализа данных,
- применять методы анализа данных с использованием языка программирования Python для решения прикладных задач;
- творчески и эффективно использовать полученные знания в профессиональной деятельности;

владеть:

- инструментами разработки программных средств Anaconda, Spyder и языка программирования Python.
- технологиями обработки данных с использованием языка программирования Python.

Программа изучаемой дисциплины рассчитана на 152 часа, из них – 38 часов аудиторных. Распределение аудиторных часов по видам занятий: лекций - 14 часов, лабораторных работ – 24 часа. Форма текущей аттестации – экзамен. Зачетные единицы – 4.

Дисциплина «Прикладное программирование» изучается студентами магистратуры дневной формы получения высшего образования на 1-м курсе в 1-ом семестре.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. Основы Python

Тема 1.1. Основы языка Python. Установка Python. Интерпретаторы Python. Возможности IDLE. Структура программы. Комментарии. Доступ к документации. Переменные. Операторы. Условные операторы и циклы. Числа. Строки и двоичные данные. Регулярные выражения. Списки. Кортежи. Множества. Диапазоны. Словари. Работа с датой и временем.

Тема 1.2. Модульное и объектно-ориентированное программирование. Пользовательские функции. Модули и пакеты. Объектно-ориентированное программирование. Определение класса. Наследование. Множественное наследование. Специальные методы. Перегрузка операторов. Статистические методы и методы класса. Абстрактные методы. Ограничение доступа к идентификаторам. Свойства класса. Детекторы классов. Обработка исключений. Итераторы. Контейнеры. Перечисления. Работа с файлами и каталогами.

Тема 1.3. Библиотеки NumPy и SciPy. Импорт модулей. Тип данных массив. Функции для создания массивов. Основные операции при работе с массивами. Базовые функции библиотеки NumPy. Индексы и срезы. Статистические функции. Файловые функции. Основные пакеты модуля SciPy. Константы. Модуль `scipy.linalg`. Интерполирование. Решение уравнений средствами SciPy. Оптимизация `scipy.optimize`. Интегрирование средствами `scipy.integrate`. Решение ОДУ в `scipy.integrate`.

Раздел 2. Визуализация и машинное обучение.

Тема 2.1. Визуализация с помощью библиотеки Matplotlib. Построение графиков из сценария. Сохранение рисунков в файл. Интерфейс в стиле MATLAB. Объектно-ориентированный интерфейс. Простые линейные графики. Настройка графика: цвет и тип линии. Метки на графиках. Простые диаграммы рассеяния. Построение диаграмм рассеяния с помощью функции `plt.scatter`. Визуализация погрешностей. Графики плотности и контурные графики. Гистограммы, разбиения по интервалам и плотность. Гистограммы нескольких распределений. Двумерные гистограммы и разбиения по интервалам. Пользовательские настройки легенд на графиках. Задание легенды для точек различного размера. `plt.subplot`: простые сетки субграфиков. Стрелки и поясняющие надписи. Краткая сводка локаторов. Краткая сводка форматоров.

Тема 2.2. Python. Машинное обучение. Категории машинного обучения. Обучение с учителем. Классификация. Регрессия. Обучение без учителя. Кластеризация. Понижение размерности. Представление данных в Scikit-Learn. Данные как таблица. Основы API статистического оценивания. Простая линейная регрессия. Выбор класса модели. Выбор гиперпараметров модели. Прикладная задача: анализ рукописных цифр. Обучение без учителя: понижение размерности. Перекрестная проверка. Выбор оптимальной моде-

ли. Компромисс между систематической ошибкой и дисперсией. Оценки эффективности. Кривые проверки в библиотеке Scikit-Learn.

Раздел 3. Обработка изображений и графический интерфейс.

Тема 3.1. Обработка изображений в Python. Библиотеки по обработке изображений. Scikit-image. OpenCV-Python. PILLOW. Mahotas. PythonMagick. Pycairo. Загрузка изображения. Сохранение изображения. Создание нового изображения. Получение информации об изображении. Копирование изображения. Поворот изображения. Рисование линий и фигур. Случайное блуждание. Метод активных контуров. Морфологические контуры. Методы на основе теории графов. Метод k-средних для сегментации. Метод быстрого сдвига. Водораздельный метод. Морфологические контуры. Как происходит линейная фильтрация. Простейшие фильтры. Фильтр Гаусса. Фильтр Гаусса для моделирования PSF. Медианный фильтр. Детекторы границ. Градиент изображения. Дифференцирование и свертка. Фильтр Робертса. Фильтр Превитта. Фильтры Франджи. Нерезкое маскирование.

Тема 3.2. PyQt5 Разработка приложений. Установка PyQt5. Структура PyQt-программы. ООП-стиль создания. Модули PyQt 5. Типы данных в PyQt. Управление основными циклом приложения. Многопоточные приложения. Управление окном приложения. Создание и отображение окна. Всплывающие подсказки. Программное закрытие окна. Использование таблиц стилей CSS для оформления окон. Обработка сигналов и событий. Назначение обработчиков сигналов. Генерация сигналов. Передача данных в обработчик. События окна. События клавиатуры. События мыши. Технология drag&drop.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Формы контроля знаний
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Иное		
1.	Основы Python (10 ч.):							
1.1.	Основы языка Python	2						
1.2.	Модульное и объектно-ориентрованное программирование	2						
1.3.	Библиотеки NumPy и SciPy	2						
1.4	Работа с массивами в NumPy				4			Отчет по лабораторной работе
2.	Визуализация и машинное обучение (12 ч):							
2.1.	Визуализация с помощью библиотеки Matplotlib	2						
2.2.	Использование библиотеки Matplotlib для построения графиков				4			Отчет по лабораторной работе
2.3.	Python. Машинное обучение.	2						
2.4.	Решение статистических задач средствами Python.				4			
2.5.	Машинное обучение на Python. библиотека Scikit-learn				4			Отчет по лабораторной работе
3.	Обработка изображений и графический интерфейс (8 ч):							
3.1.	Обработка изображений в Python	2						
3.2.	Общие принципы работы с изображениями в Skimage				4			Отчет по лабораторной работе
3.3.	PyQT5 Разработка приложений	2			4			Отчет по лабораторной работе
Итого:		14			24			

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Основная литература

1. Прохоренок, Н. А. Python 3 и PyQt / Н. А. Прохоренок – Санкт-Петербург: «БВХ-Петербург», 2012. – 256 с.
2. Вандер Плас Дж. Python для сложных задач: наука о данных и машинное обучение / Плас Дж. Вандер – СПб.: Питер, 2016. – 576 с.
3. Уэсли Дж. Чан. Python. Создание приложений/ Дж. Чан Уэсли. – Москва: Вильямс, 2016. – 300 с.
4. Лутц М. Python. Карманный справочник/ Марк Лутц. – Москва: Вильямс, 2016. – 320 с.
5. Fabio N. Python Data Analytics/ Nelli Fabio. – Apress, 2015. – 364 с.
6. Solem J. E. Programming Computer Vision with Python: Tools and algorithms for analyzing images Jan Erik Solem. – O'Reilly Media, 2012. – 264 с.

Дополнительная литература

1. Дюк, В. Data Mining: учебный курс / В. Дюк, А. Самойленко – СПб., 2001. 368 с
2. Кобзарь, А. И. Прикладная математическая статистика: для инженеров и научных работников / А. И. Кобзарь. – М., 2006. – 816 с
3. Мандель, И.Д. Кластерный анализ / И.Д. Мандель – М., 1988. – 176 с.
4. Сегаран, Т. Программируем коллективный разум : пер. с англ. / Т. Сегаран. – СПб., 2008. – 368с.

Перечень тем лабораторных работ:

1. Работа с массивами в NumPy (4 часа).
2. Использование библиотеки Matplotlib для построения графиков (4 часа).
3. Решение статистических задач средствами Python (4 часа).
4. Машинное обучение на Python. библиотека Scikit-learn (4 часа).
5. Общие принципы работы с изображениями в Skimage (4 часа)
6. PyQt5 Разработка приложений (4 часа)

Методические рекомендации по организации управляемой самостоятельной работы студентов

Для организации управляемой самостоятельной работы студентов по учебной дисциплине рекомендуется использовать современные информационные технологии: размещение в сетевом доступе комплекса учебных и учебно-методических материалов (программа курса, электронный вариант учебного пособия, методические указания, задания и информационные ресурсы для выполнения лабораторных работ, список рекомендуемой литературы и др.).

Перечень используемых средств диагностики результатов учебной деятельности

Для оценки уровня знаний студентов используется следующий диагностический инструментарий:

- защита индивидуальных заданий при выполнении лабораторных работ;
- отчеты по лабораторным работам;

Методика формирования итоговой оценки

Итоговая оценка формируется на основе:

1. Правил проведения аттестации студентов (Постановление Министерства образования Республики Беларусь № 53 от 29.05.2012 г.);
2. Положения о рейтинговой системе оценки знаний по дисциплине в БГУ (Приказ ректора БГУ от 18.08.2015 г. № 382-ОД);
3. Критериев оценки знаний студентов (письмо Министерства образования от 22.12.2003 г.).

Итоговая оценка по курсу выводится по рейтинговой системе и определяется как сумма текущей средней оценки по лабораторным работам, взятой с коэффициентом 0,7 и экзаменационной оценки, взятой с коэффициентом 0,3. Отчеты по лабораторным работам и тестам оцениваются по 10-бальной системе. Оценка текущей успеваемости определяется как средняя по оценкам лабораторных работ.

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ УВО
на ____ / ____ учебный год

№ п/п	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры
_____ (протокол № ____ от _____ 20__ г.)
(название кафедры)

Заведующий кафедрой

(ученая степень, ученое звание)

(подпись)

(И.О.Фамилия)

УТВЕРЖДАЮ
Декан факультета

(ученая степень, ученое звание)

(подпись)

(И.О.Фамилия)