

9. Методические указания для обучающихся по освоению дисциплины (модуля)

Можно изучать дополнительно материалы похожих курсов:

- 1) Основы программирования на Python <https://www.coursera.org/learn/python-osnovy-programmirovaniya>
- 2) Язык программирования Python <http://www.intuit.ru/studies/courses/49/49/info>

Приложение

Оценочные материалы по дисциплине (модулю)

по направлению:	Информатика и вычислительная техника
профиль подготовки:	Системный анализ и инжиниринг информационных процессов Сетевое обучение Кафедра «Алгоритмов и технологий программирования»
курс:	2
квалификация:	бакалавр

Семестр, форма промежуточной аттестации: 3 (осенний) – зачет с оценкой

Разработчик: Д.Е. Беляков, ассистент

1. Компетенции, формируемые в процессе изучения дисциплины (модуля)

Код и наименование компетенции	Индикаторы достижения компетенций
ОПК-1 Способен применять естественнонаучные и общетехнические знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности	ОПК-1.1 Демонстрирует естественнонаучные и общетехнические знания, знания методов математического анализа и моделирования, теоретического и экспериментального исследования
	ОПК-1.2 Применяет естественнонаучные и общетехнические знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности
	ОПК-1.3 Использует современные информационные технологии в профессиональной деятельности
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Осуществляет поиск необходимой информации, подвергает ее критическому анализу и обобщению
	УК-1.2 Применяет системный подход для решения поставленных задач

2. Показатели оценивания компетенция

В результате освоения дисциплины Учебная практика (Программирование на языке Python), обучающиеся должны:

- знать:
 - синтаксис языка программирования Python;
 - общепринятые способы решения базовых задач с использованием особенностей языка;
 - основные библиотеки и фреймворки на Python;
 - принцип исполнения программ на Python;
 - типы данных языка Python;
 - управление потоком выполнения в Python;
 - возможности стандартной библиотеки;
 - правила работы с исключениями;
 - внутреннее строение контейнеров стандартной библиотеки и временную сложность операций с ними;
 - принцип работы сборки мусора в Python;
 - кодировки, используемые при хранении текстовых данных (ASCII, Windows-1250/1251, UTF-8, UTF-16).
- уметь:
 - реализовывать библиотеку общего назначения на языке Python по заданным интерфейсам;
 - решать задачи, связанные с обработкой данных, на языке Python.
- владеть:
 - основными библиотеками и инструментами разработчика на языке Python.

3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

Примеры контрольных заданий:

1_3_2. Дан массив целых чисел $A[0..n-1]$. Известно, что на интервале $[0, m]$ значения массива строго возрастают, а на интервале $[m, n-1]$ строго убывают. Найти m за $O(\log m)$.
 $2 \leq n \leq 10000$.

2_4_1. Первые k элементов длинной последовательности.
Дана очень длинная последовательность целых чисел длины n . Требуется вывести в отсортированном виде её первые k элементов. Последовательность может не помещаться в память. Время работы $O(n * \log(k))$. Доп. память $O(k)$. Использовать слияние.

3_3_2. Порядковые статистики. Дано число N и N строк. Каждая строка содержит команду добавления или удаления натуральных чисел, а также запрос на получение k -ой порядковой статистики. Команда добавления числа A задается положительным числом A , команда удаления числа A задается отрицательным числом “- A ”. Запрос на получение k -ой порядковой статистики задается числом k . Требуемая скорость выполнения запроса - $O(\log n)$.

4_4_1. Самая удаленная вершина.
Для каждой вершины определите расстояние до самой удаленной от нее вершины. Время работы должно быть $O(n)$.

Формат входных данных:

В первой строке записано количество вершин $n \leq 10000$. Затем следует $n - 1$ строка, описывающая ребра дерева. Каждое ребро – это два различных целых числа – индексы вершин в диапазоне

$[0, n-1]$. Индекс корня – 0. В каждом ребре родительской вершиной является та, чей номер меньше.

Формат выходных данных:

Выход должен содержать n строк. В i -ой строке выводится расстояние от i -ой вершины до самой удаленной от нее.

4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

1. Среда разработки Python.
2. Сингулярные типы данных.
3. Типы контейнеров.
4. Изменяемые и неизменяемые типы данных.
5. Условные операторы и циклы.
6. Итераторы и генераторы. Отличия реализации.
7. Функциональное программирование. Анонимные функции.
8. Замыкания и декораторы. Декорирование рекурсивных функций.
9. Объектно-ориентированное программирование. Наследование, инкапсуляция, приватность.
10. Специальные методы классов. Менеджеры контекстов.
11. Библиотеки Python, используемые в web-разработке.
12. Библиотеки Python, используемые в анализе данных.

Критерии оценивания

отлично

«10» – всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений;

«9» – систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, правильное обоснование принятых решений;

«8» – глубокие знания учебной программы дисциплины и умение применять их на практике при решении конкретных задач, правильное обоснование принятых решений;

хорошо

«7» – твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;

«6» – знает материал, грамотно излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;

«5» – знает основной материал, грамотно излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач неточности;

удовлетворительно

«4» – фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении

программного материала, но при этом он владеет основными разделами учебной программы, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;

«3» – характер знаний достаточен для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;

неудовлетворительно

«2» – не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных понятий дисциплины и не умеет правильно использовать полученные знания при решении типовых практических задач;

«1» – не знает формулировок основных понятий дисциплины и не умеет использовать полученные знания при решении типовых практических задач.

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Дифференцированный зачет может проводиться по итогам текущей успеваемости и сдачи заданий, лабораторных и других видов работ, предусмотренных программой дисциплины и (или) путем организации специального опроса, проводимого в устной и (или) письменной форме, а также с выдачей заданий для реализации на компьютере.