

**ПРИЛОЖЕНИЕ А**  
**ФОНД ОЦЕНОЧНЫХ МАТЕРИАЛОВ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ**  
**ПО ДИСЦИПЛИНЕ «Функциональное и логическое программирование»**

**1. Перечень оценочных средств для компетенций, формируемых в результате освоения дисциплины**

<b>Код контролируемой компетенции</b>	<b>Способ оценивания</b>	<b>Оценочное средство</b>
ПК-5: Владение навыками использования различных технологий разработки программного обеспечения	Зачет	Комплект контролирующих материалов для зачета

**2. Описание показателей и критериев оценивания компетенций, описание шкал оценивания**

Оцениваемые компетенции представлены в разделе «Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций» рабочей программы дисциплины «Функциональное и логическое программирование».

При оценивании сформированности компетенций по дисциплине «Функциональное и логическое программирование» используется 100-балльная шкала.

<b>Критерий</b>	<b>Оценка по 100-балльной шкале</b>	<b>Оценка по традиционной шкале</b>
Студент освоил изучаемый материал, выполняет задания в соответствии с индикаторами достижения компетенций, может допускать отдельные ошибки.	25-100	<i>Зачтено</i>
Студент не освоил основное содержание изученного материала, задания в соответствии с индикаторами достижения компетенций не выполнены или выполнены неверно.	0-24	<i>Не зачтено</i>

**3. Типовые контрольные задания или иные материалы, необходимые для оценки уровня достижения компетенций в соответствии с индикаторами**

*1. Использование хвостовой рекурсии в задачах профессиональной деятельности*

<b>Компетенция</b>	<b>Индикатор достижения компетенции</b>
ПК-5 Владение навыками использования различных технологий разработки программного обеспечения	ПК-5.1 Выбирает необходимую технологию разработки программного обеспечения для решения прикладных задач
	ПК-5.2 Использует современные технологии разработки программного обеспечения для решения прикладных задач

Компетенция	Содержимое компетенции	Индикатор	Содержимое индикатора
ПК-5	Владение навыками использования различных технологий разработки программного обеспечения	ПК-5.1	Выбирает необходимую технологию разработки программного обеспечения для решения прикладных задач
		ПК-5.2	Использует современные технологии разработки программного обеспечения для решения прикладных задач

При обработке естественных языков, после этапа токенизации, слова предложения часто представляются в виде списков как произвольной, так и заданной структуры.

Необходимо реализовать обработку списков известной структуры с использованием механизма частной (хвостовой) рекурсии.

1. Для произвольного списка определить, является ли последний элемент списка списком или атомом.
2. Для произвольного списка построить список той же длины, состоящий из атомов T и NIL в зависимости от того, является ли соответствующий элемент исходного списка списком или атомом соответственно.
3. Определить функцию, разбивающую список на пары:  
 $(A B C D) \rightarrow ((A B) (C D))$
4. Написать функцию, которая, чередуя элементы двух списков, строит новый список:  
 $(A B C) (1 2 3) \rightarrow (A 1 B 2 C 3)$
5. Два множества представлены списками. Найти пересечение множеств.
6. Из данного списка удалить все элементы, которые являются списками длиной более K.

## 2. Использование хвостовой рекурсии в задачах профессиональной деятельности

Компетенция	Индикатор достижения компетенции
ПК-5 Владение навыками использования различных технологий разработки программного обеспечения	ПК-5.1 Выбирает необходимую технологию разработки программного обеспечения для решения прикладных задач
	ПК-5.2 Использует современные технологии разработки программного обеспечения для решения прикладных задач

Компетенция	Содержимое компетенции	Индикатор	Содержимое индикатора
ПК-5	Владение навыками использования различных технологий разработки программного обеспечения	ПК-5.1	Выбирает необходимую технологию разработки программного обеспечения для решения прикладных задач
		ПК-5.2	Использует современные технологии разработки программного обеспечения для решения прикладных задач

При обработке естественных языков, после этапа графематического и морфологического анализа, слова предложения представляются в виде списков токенов произвольной структуры с набором меток .

Необходимо реализовать обработку списков произвольной структуры с использованием механизма общей рекурсии.

1. Написать функцию, которая из заданного списка строит одноуровневый список:  
 $(A (B (C 1) D)) \rightarrow (A B C 1 D)$
2. Даны два списка одинаковой структуры. Построить список такой же структуры, состоящий из пар элементов.  
 $((A B(V)C)D), ((1 C (2)3)4) \rightarrow (((A 1) (B C))(V 2))(C 3))(D 4)$
3. Определить, сколько атомов находится на заданном уровне вложенности.  
 Элемент может быть как атомом, так и списком.  
 $(A B (B A (C)) A C (A)), 2 \rightarrow 3$

### *3.Использование функционального подхода для обработки списков в задачах профессиональной деятельности*

Компетенция	Индикатор достижения компетенции
ПК-5 Владение навыками использования различных технологий разработки программного обеспечения	ПК-5.1 Выбирает необходимую технологию разработки программного обеспечения для решения прикладных задач
	ПК-5.2 Использует современные технологии разработки программного обеспечения для решения прикладных задач

Компетенция	Содержимое компетенции	Индикатор	Содержимое индикатора
ПК-5	Владение навыками использования различных технологий разработки программного обеспечения	ПК-5.1	Выбирает необходимую технологию разработки программного обеспечения для решения прикладных задач
		ПК-5.2	Использует современные технологии разработки программного обеспечения для решения прикладных задач

При анализе данных, некоторые объекты могут быть представлены в виде списков и множеств, например имена признаков объекта и т.п.

Выполнить обработку списков на языке Python через любые функции высших порядков: **map**, **reduce**, **filter**, **any** и т.п., а также **list comprehension**.

1. Дан список из произвольных элементов. Найти наиболее часто встречающийся элемент.
2. Выполнить преобразование списка.  
(a a a a b c c a a d e e e e) => ((4 a) (1 b) (2 c) (2 a) (1 d) (4 e))
3. Преобразовать двухуровневый список в одноуровневый.
4. Два множества представлены списками. Найти объединение множеств.

#### *4.Использование функционального подхода для обработки задач общего вида в профессиональной деятельности*

Компетенция	Индикатор достижения компетенции
ПК-5 Владение навыками использования различных технологий разработки программного обеспечения	ПК-5.1 Выбирает необходимую технологию разработки программного обеспечения для решения прикладных задач
	ПК-5.2 Использует современные технологии разработки программного обеспечения для решения прикладных задач

Компетенция	Содержимое компетенции	Индикатор	Содержимое индикатора
ПК-5	Владение навыками использования различных технологий разработки программного обеспечения	ПК-5.1	Выбирает необходимую технологию разработки программного обеспечения для решения прикладных задач
		ПК-5.2	Использует современные технологии разработки программного обеспечения для решения прикладных задач

При прохождении собеседования при приеме на работу в корпорации Yandex, соискателям предлагают решить ряд нестандартных задач. Функциональная парадигма часто помогает в эффективном решении таких задач. Вам предлагается привести решение задачи на ЯП Python с использованием функциональной парадигмы (списковые включения, функции высших порядков и т.п.)

1. Допустим, требуется зашифровать сообщения, состоящие из определенных символов. Например, исходный текст состоит из цифры от 0 до 9 (представленных в программе в виде символов) и других знаков и имеет длину в 15 символов. В шифровке на место цифр подставляются соответствующие им знаки, на место всех остальных символов дополнительный 11-й знак.
2. Дан массив из двадцати элементов. Затем находит самую длинную последовательность из нулей и выводит на экран ее длину и номер ее начала в массиве.
3. Двоичное число, введенное по запросу с клавиатуры, в случае правильной записи преобразовать в шестнадцатеричное и десятичное число и результат вывести на экран.
4. Написать программу определения количества шестизначных "счастливых" трамвайных билетов, у которых сумма первых трех цифр совпадает с суммой трех последних.

**4. Файл и/или БТЗ с полным комплектом оценочных материалов прилагается.**