МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ»
(ТУСУР)

УТВЕРЖДАЮ
Директор департамента образования
_____ П. Е. Троян
«___» _____ 20__ г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Функциональное программирование и интеллектуальные системы

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки / специальность: **38.03.05 Бизнес-информатика**
Направленность (профиль) / специализация: **ИТ-предпринимательство**
Форма обучения: **заочная (в том числе с применением дистанционных образовательных технологий)**
Факультет: **ФДО, Факультет дистанционного обучения**
Кафедра: **АОИ, Кафедра автоматизации обработки информации**
Курс: **5**
Семестр: **9**
Учебный план набора 2014 года

Распределение рабочего времени

№	Виды учебной деятельности	9 семестр	Всего	Единицы
1	Самостоятельная работа под руководством преподавателя	12	12	часов
2	Контроль самостоятельной работы	2	2	часов
3	Всего контактной работы	14	14	часов
4	Самостоятельная работа	121	121	часов
5	Всего (без экзамена)	135	135	часов
6	Подготовка и сдача экзамена	9	9	часов
7	Общая трудоемкость	144	144	часов
			4.0	З.Е.

Контрольные работы: 9 семестр - 1
Экзамен: 9 семестр

Томск 2018

Согласована на портале № 43666

Рассмотрена и одобрена на заседании кафедры
протокол № 318 от « 3 » 7 2018 г.

ЛИСТ СОГЛАСОВАНИЯ

Рабочая программа дисциплины составлена с учетом требований федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки (специальности) 38.03.05 Бизнес-информатика, утвержденного 11.08.2016 года, рассмотрена и одобрена на заседании кафедры АОИ «___» _____ 20__ года, протокол №_____.

Разработчики:

доцент каф. ТЭО	_____	Ю. В. Морозова
доцент каф. АОИ	_____	Н. Ю. Салмина

Заведующий обеспечивающей каф. АОИ	_____	Ю. П. Ехлаков

Рабочая программа дисциплины согласована с факультетом и выпускающей кафедрой:

Декан ФДО	_____	И. П. Черкашина
Заведующий выпускающей каф. АОИ	_____	Ю. П. Ехлаков

Эксперты:

Доцент кафедры технологий электронного обучения (ТЭО)	_____	Ю. В. Морозова
Доцент кафедры автоматизации обработки информации (АОИ)	_____	А. А. Сидоров

Согласована на портале № 43666

1. Цели и задачи дисциплины

1.1. Цели дисциплины

ознакомление студентов с основами функционального программирования на примере языка Лисп, принципами структурирования знаний в виде фреймов и семантических сетей, а также формирование у студентов профессиональных знаний и практических навыков по разработке и созданию интеллектуальных систем с помощью языка функционального программирования Лисп

1.2. Задачи дисциплины

– получить знания и овладеть понятийным аппаратом: интеллектуальные системы; функциональное программирование; λ-исчисление; функционалы.

2. Место дисциплины в структуре ОПОП

Дисциплина «Функциональное программирование и интеллектуальные системы» (Б1.В.ОД.9) относится к блоку 1 (вариативная часть).

Предшествующими дисциплинами, формирующими начальные знания, являются: Дискретная математика, Программирование.

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

– ПК-18 способность использовать соответствующий математический аппарат и инструментальные средства для обработки, анализа и систематизации информации по теме исследования;

В результате изучения дисциплины обучающийся должен:

– **знать** языки функционального программирования; основные методы и средства эффективной разработки программного продукта; типовые роли в процессе разработки программного обеспечения; методологии разработки программного обеспечения; математические основы лямбда-исчисления.

– **уметь** использовать методы и технологии разработки для генерации исполняемого кода; анализировать поставленные задачи, разрабатывать алгоритмы, представлять знания для решения поставленных задач; разрабатывать модели различных классов систем с применением языка функционального программирования; осуществлять разработку программного обеспечения на языке Лисп.

– **владеть** основными методологиями процессов разработки программного обеспечения; математическим аппаратом, применяемым в функциональном и логическом программировании; языком Лисп для построения моделей искусственного интеллекта.

4. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 4.0 зачетных единицы и представлена в таблице 4.1.

Таблица 4.1 – Трудоемкость дисциплины

Виды учебной деятельности	Всего часов	Семестры
		9 семестр
Контактная работа (всего)	14	14
Самостоятельная работа под руководством преподавателя (СРП)	12	12
Контроль самостоятельной работы (КСР)	2	2
Самостоятельная работа (всего)	121	121
Подготовка к контрольным работам	10	10
Самостоятельное изучение тем (вопросов) теоретической части курса	111	111
Всего (без экзамена)	135	135
Подготовка и сдача экзамена	9	9

Согласована на портале № 43666

Общая трудоемкость, ч	144	144
Зачетные Единицы	4.0	

5. Содержание дисциплины

5.1. Разделы дисциплины и виды занятий

Разделы дисциплины и виды занятий приведены в таблице 5.1.

Таблица 5.1 – Разделы дисциплины и виды занятий

Названия разделов дисциплины	СРП, ч	КСР, ч	Сам. раб., ч	Всего часов (без экзамена)	Формируемые компетенции
9 семестр					
1 Язык Лисп. Основы программирования	2	2	26	28	ПК-18
2 Рекурсивные функции	2		31	33	ПК-18
3 Технологии программирование на языке Лисп	4		38	42	ПК-18
4 Модели представления знаний	4		26	30	ПК-18
Итого за семестр	12	2	121	135	
Итого	12	2	121	135	

5.2. Содержание разделов дисциплины (самостоятельная работа под руководством преподавателя)

Содержание разделов дисциплин (самостоятельная работа под руководством преподавателя) приведено в таблице 5.2.

Таблица 5.2 – Содержание разделов дисциплин (самостоятельная работа под руководством преподавателя)

Названия разделов	Содержание разделов дисциплины (самостоятельная работа под руководством преподавателя)	Трудоемкость, ч	Формируемые компетенции
9 семестр			
1 Язык Лисп. Основы программирования	Лисп. Элементарные понятия: атомы и списки как основные объекты языка ЛИСП. Программа на языке Лисп. Вычислимые выражения. Базовые функции языка. Лямбда-выражения, и определение новых функций	2	ПК-18
	Итого	2	
2 Рекурсивные функции	Понятие рекурсии. Правила записи рекурсивной функции. Рекурсия с несколькими терминальными или рекурсивными ветвями. Вспомогательные функции над списками	2	ПК-18
	Итого	2	
3 Технологии программирование на языке Лисп	Передача параметров. Глобальные и локальные переменные. Диалоговый режим работы. Функции ввода-вывода. Разрушающие функции. Функционалы. Циклы и	4	ПК-18

Согласована на портале № 43666

	блочные функции. Массивы. Свойства символов. Работа с файлами: определение входных и выходных потоков, чтение символов из файла		
	Итого	4	
4 Модели представления знаний	Фреймы: понятие и состав фрейма, пример построения фреймовой структуры в ЛИСП. Семантические сети: представление знаний с помощью семантических сетей, пример построения семантической сети с помощью языка ЛИСП	4	ПК-18
	Итого	4	
Итого за семестр		12	

5.3. Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (последующими) дисциплинами

Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (последующими) дисциплинами представлены в таблице 5.3.

Таблица 5.3 – Разделы дисциплины и междисциплинарные связи

Наименование дисциплин	№ разделов данной дисциплины, для которых необходимо изучение обеспечивающих и обеспечиваемых дисциплин			
	1	2	3	4
Предшествующие дисциплины				
1 Дискретная математика		+		+
2 Программирование	+		+	

5.4. Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий представлено в таблице 5.4.

Таблица 5.4 – Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Компетенции	Виды занятий			Формы контроля
	СРП	КСР	Сам. раб.	
ПК-18	+	+	+	Контрольная работа, Экзамен, Проверка контрольных работ, Тест

6. Интерактивные методы и формы организации обучения

Не предусмотрено РУП.

7. Лабораторные работы

Не предусмотрено РУП.

8. Контроль самостоятельной работы

Виды контроля самостоятельной работы приведены в таблице 8.1.

Таблица 8.1 – Виды контроля самостоятельной работы

№	Вид контроля самостоятельной работы	Трудоемкость (час.)	Формируемые компетенции
9 семестр			
1	Контрольная работа с автоматизированной про-	2	ПК-18

Согласована на портале № 43666

	веркой		
Итого		2	

9. Самостоятельная работа

Виды самостоятельной работы, трудоемкость и формируемые компетенции представлены в таблице 9.1.

Таблица 9.1 – Виды самостоятельной работы, трудоемкость и формируемые компетенции

Названия разделов	Виды самостоятельной работы	Трудоемкость, ч	Формируемые компетенции	Формы контроля
9 семестр				
1 Язык Лисп. Основы программирования	Самостоятельное изучение тем (вопросов) теоретической части курса	24	ПК-18	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	2		
	Итого	26		
2 Рекурсивные функции	Самостоятельное изучение тем (вопросов) теоретической части курса	27	ПК-18	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	4		
	Итого	31		
3 Технологии программирование на языке Лисп	Самостоятельное изучение тем (вопросов) теоретической части курса	36	ПК-18	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	2		
	Итого	38		
4 Модели представления знаний	Самостоятельное изучение тем (вопросов) теоретической части курса	24	ПК-18	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	2		
	Итого	26		
	Выполнение контрольной работы	2	ПК-18	Контрольная работа
Итого за семестр		121		
	Подготовка и сдача экзамена	9		Экзамен
Итого		130		

Согласована на портале № 43666

10. Контроль самостоятельной работы (курсовой проект / курсовая работа)

Не предусмотрено РУП.

11. Рейтинговая система для оценки успеваемости обучающихся

Рейтинговая система не используется.

12. Учебно-методическое и информационное обеспечение дисциплины

12.1. Основная литература

1. Салмина Н.Ю. Функциональное программирование и интеллектуальные системы [Электронный ресурс]: учебное пособие. -Томск ФДО, ТУСУР, 2016. Доступ из личного кабинета студента. - Режим доступа: https://study.tusur.ru/study/library/ (дата обращения: 20.08.2018).

12.2. Дополнительная литература

1. Душкин, Р.В. Функциональное программирование на языке Haskell [Электронный ресурс]: учебник / Р.В. Душкин. — Электрон. дан. — Москва ДМК Пресс, 2008. Доступ из личного кабинета студента. - Режим доступа: https://e.lanbook.com/book/1247 (дата обращения: 20.08.2018).

12.3. Учебно-методические пособия

12.3.1. Обязательные учебно-методические пособия

1. Салмина Н.Ю. Функциональное программирование и интеллектуальные системы : электронный курс / Н. Ю. Салмина. – Томск ТУСУР, ФДО, 2016. Доступ из личного кабинета студента.

2. Салмина Н.Ю. Функциональное программирование и интеллектуальные системы [Электронный ресурс]: методические указания по организации самостоятельной работы для студентов заочной формы обучения направления подготовки 38.03.05 Бизнес-информатика, обучающихся с применением дистанционных образовательных технологий / Н.Ю. Салмина, Ю.П. Ехлаков. – Томск ФДО, ТУСУР, 2018. Доступ из личного кабинета студента. - Режим доступа: https://study.tusur.ru/study/library/ (дата обращения: 20.08.2018).

12.3.2. Учебно-методические пособия для лиц с ограниченными возможностями здоровья и инвалидов

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

Для лиц с нарушениями зрения:

- в форме электронного документа;
- в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:

- в форме электронного документа;
- в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:

- в форме электронного документа;
- в печатной форме.

12.4. Профессиональные базы данных и информационные справочные системы

1. ЭБС «Лань»: www.e.lanbook.com (доступ из личного кабинета студента по ссылке http://lanbook.fdo.tusur.ru).

13. Материально-техническое обеспечение дисциплины и требуемое программное обеспечение

13.1. Общие требования к материально-техническому и программному обеспечению дисциплины

13.1.1. Материально-техническое и программное обеспечение дисциплины

Кабинет для самостоятельной работы студентов

учебная аудитория для проведения занятий лабораторного типа, помещение для проведения групповых и индивидуальных консультаций, помещение для проведения текущего контроля и про-

Согласована на портале № 43666

межуточной аттестации, помещение для самостоятельной работы

634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Описание имеющегося оборудования:

- Коммутатор MicroTeak;
- Компьютер PENTIUM D 945 (3 шт.);
- Компьютер GELERON D 331 (2 шт.);
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:

- 7-zip
- Google Chrome
- Kaspersky Endpoint Security 10 для Windows
- Microsoft Windows
- OpenOffice
- SWI-Prolog (с возможностью удаленного доступа)
- XLisp (с возможностью удаленного доступа)

13.1.2. Материально-техническое и программное обеспечение для самостоятельной работы

Для самостоятельной работы используются учебные аудитории (компьютерные классы), расположенные по адресам:

- 634050, Томская область, г. Томск, Ленина проспект, д. 40, 233 ауд.;
- 634045, Томская область, г. Томск, ул. Красноармейская, д. 146, 201 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 47, 126 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Состав оборудования:

- учебная мебель;
- компьютеры класса не ниже ПЭВМ INTEL Celeron D336 2.8ГГц. - 5 шт.;
- компьютеры подключены к сети «Интернет» и обеспечивают доступ в электронную информационно-образовательную среду университета.

Перечень программного обеспечения:

- Microsoft Windows;
- OpenOffice;
- Kaspersky Endpoint Security 10 для Windows;
- 7-Zip;
- Google Chrome.

13.2. Материально-техническое обеспечение дисциплины для лиц с ограниченными возможностями здоровья и инвалидов

Освоение дисциплины лицами с ограниченными возможностями здоровья и инвалидами осуществляется с использованием средств обучения общего и специального назначения.

При занятиях с обучающимися **с нарушениями слуха** предусмотрено использование звукоусиливающей аппаратуры, мультимедийных средств и других технических средств приема/передачи учебной информации в доступных формах, мобильной системы преподавания для обучающихся с инвалидностью, портативной индукционной системы. Учебная аудитория, в которой занимаются обучающиеся с нарушением слуха, оборудована компьютерной техникой, аудиотехникой, видеотехникой, электронной доской, мультимедийной системой.

При занятиях с обучающимися **с нарушениями зрениями** предусмотрено использование в лекционных и учебных аудиториях возможности просмотра удаленных объектов (например, текста на доске или слайда на экране) при помощи видеоувеличителей для комфортного просмотра.

При занятиях с обучающимися **с нарушениями опорно-двигательного аппарата** используются альтернативные устройства ввода информации и другие технические средства приема/пере-

Согласована на портале № 43666

дачи учебной информации в доступных формах, мобильной системы обучения для людей с инвалидностью.

14. Оценочные материалы и методические рекомендации по организации изучения дисциплины

14.1. Содержание оценочных материалов и методические рекомендации

Для оценки степени сформированности и уровня освоения закрепленных за дисциплиной компетенций используются оценочные материалы в составе:

14.1.1. Тестовые задания

1. Основной формой представления данных в функциональном программировании являются списки. Для получения необходимой информации мы должны уметь обрабатывать списки и выбирать из них требуемые данные.

Дан список (setq x `(a s (d) f g)). Как можно выбрать из указанного списка атом «d», используя суперпозицию функций CAR и CDR?

1) (CADR X)
2) (CDDAAR X)
3) (CDDAR X)
4) (CAADDR X)

2. В двух списках X и Y хранятся данные о возрасте животных в исследуемых популяциях. Необходимо перенести информацию об одном из животных из одного списка в другой. Будем считать, что это возраст этого животного записан во втором элементе списка X. Какое из приведенных выражений переносит информацию о нем во второй список?

1) (LIST (CAR X) (CAR Y))
2) (CONS (CADR X) Y)
3) (APPEND X Y)
4) (CADR (APPEND X Y))

3. Для разветвления вычислений в функциональном языке Лисп используется условное предложение COND. Задан числовой список X из трех элементов. Какое действие выполняет предложение

(COND ((AND(>(CAR X)(CADR X))(>(CAR X)(CADDR X)))(CAR X))
((AND(>(CADR X)(CAR X))(>(CADR X)(CADDR X)))(CADR X))
(T (CADDR X))) ?

1) выбирает третий элемент списка
2) выбирает элементы списка в порядке возрастания
3) выполняет логическое сравнение элементов списка
4) выбирает максимальный элемент в списке.

4. В основе всех функциональных языков лежит лямбда-исчисление в том смысле, что все функциональные программы можно преобразовать в лямбда-выражение. Что выдаст следующее лямбда-выражение, описанное на языке Лисп?

((lambda (x y) (cond ((zerop x) (* y y)) ((< x 0) (+ y y)) (t (+ x y)))) (+ -10 2) (+ 2 10))

1) 24
2) 4
3) Nil
4) 144

5. Механизм рекурсивного вызова является одним из основных принципов функционального программирования. Что выполняет следующая рекурсивная функция, аргументом которой является список?

(defun q (z) (cond ((null z) nil)
(t (append (q (cdr z)) (list (car z))))))

1) переставляет последний элемент списка в начало;

Согласована на портале № 43666

2) меняет первый и последний элемент списка местами;
3) переставляет первый элемент списка в конец списка;
4) переставляет элементы списка в обратном порядке.

6. Любая рекурсивная функция должна иметь терминальные ветви (определяющие правило останова) и рекурсивные ветви. Какое количество терминальных ветвей содержит следующая рекурсивная функция?

(defun q (z) (cond ((null z) nil)
((null (cdr z)) 0)
((not (numberp (car z))) nil)
(t (+ (*(car z)(cadr z)) (q (cddr z))))))
1) 1
2) 2
3) 3
4) 4

7. Обработка списков с помощью рекурсии или с применением функционалов позволяет легко рассчитывать некоторые статистические параметры. Какая из описанных функций рассчитывает среднее арифметическое числовой последовательности, заданной в виде списка X?

1) (defun f (x) (cond ((null x) 0)(t (+ (car x)(f (cdr x))))))
2) (defun f (x) (/ (cond ((null x) 0)(t (+ (car x)(f (cdr x))))) (length x)))
3) (defun f (x) (/ (apply `+ x) (length x)))
4) (defun f (x) (mapcar `/ (apply `+ x) (length x)))

8. Функции, которые не формируют новые списки, а изменяют структуру существующих списков, называются разрушающими. Чему будет равен Y в результате следующей последовательности вызова вычислимых выражений с использованием разрушающей функции?

(setq x `(2 3))
(setq y (cons 1 x))
(rplaca x 7)
y - ?
1) (1 2 3)
2) (1 7)
3) 7
4) (1 7 3)

9. Любой функциональный язык содержит функционалы: функции, имеющие аргументы, значением которых являются функции. Имеется список X, элементами которого являются возраста людей исследуемой группы. Какое из перечисленных выражений вычисляет суммарный возраст исследуемой группы с использованием функционала?

1) (APPLY `+ X)
2) (MAPCAR `SUM X)
3) (MAPLIST `+ X)
4) (FUNCALL `+ X)

10. Использование механизма циклов вместо рекурсии позволяет экономить память и строить, зачастую, более эффективные программы. Что будет получено в результате работы следующего вычислимого выражения с использованием цикла?

(let ((x 0)(y nil))(loop (setq x (+ 1 x))(setq y (cons x y))(cond((= x 5)(return y)))))
1) 5
2) nil
3) (5 4 3 2 1)
4) (1 2 3 4 5)

Согласована на портале № 43666

11. Применение функционалов в программе основано на том, что программы и данные в языках функционального программирования представляются одинаково. Определена функция SUM, аргументом которой является список, а результат работы – сумма элементов списка.

Что будет получено в результате работы следующего функционала, где функция SUM рассматривается как аргумент другой функции?

(maplist `sum `(1 2 3 4 5))

1) 15
2) (15 14 12 9 5)
3) (5 9 12 14 15)
4) (1 2 3 4 5)

12. Основная структура данных в языках функционального программирования – списки. По сути, любая функция на языке Лисп является функцией обработки списков. Какой список свойств получится в результате выполнения следующей последовательности вычислимых выражений?

(setf (get `as `v4) `(4))
(setf (get `as `v2) `(2))
(setf (get `as `v3) `(3))
(setf (get `as `v1) `(1))

1) (v4 4 v3 3 v2 2 v1 1)
2) (v1 1 v3 3 v2 2 v4 4)
3) (v4 (4) v2 (2) v3 (3) v1 (1))
4) (v1 (1) v3 (3) v2 (2) v4 (4))

13. Особый вид списков, используемых в языке Лисп – ассоциативные списки, элементами которых являются точечные пары. Какой ассоциативный список получится в результате работы следующей функции: (pairlis `((a) (b) (c)) `(1 2 3) ())?

1) ((c . 3)(b . 2)(a . 1))
2) (c 3 b 2 a 1)
3) ((a . 1)(b . 2)(c . 3))
4) ((a 1)(b 2)(c 3))

14. Списки свойств в языке Лисп позволяют хранить знания в удобной форме. Пусть мы имеем список свойств, в котором хранится информация о студенте Иванов: (группа 445 возраст 19). Какое из перечисленных выражений изменяет список свойств, увеличивая возраст студента на 1 год?

1) (+ 1 (get `Иванов `возраст))
2) (setf (get `Иванов `возраст) (+ 1 (get `Иванов `возраст)))
3) (setq Иванов `(группа 445 возраст 20))
4) (setf `Иванов (+ 1 (get `Иванов `возраст)))

15. Ассоциативные списки в Лиспе позволяют удобно хранить и легко получать требуемую информацию в зависимости от значения ключа. Пусть у нас задан ассоциативный список, в котором храниться информация о Лене в следующем виде: (setq Лена `((возраст . 29)(хобби . рисование)(профессия . учитель))). Какое из следующих выражений позволяет получить информацию о профессии Лены?

1) (car (assoc `профессия Лена))
2) (caddr Лена)
3) (assoc `профессия Лена)
4) (assoc `профессия `Лена)

16. Фрейм является одной из форм представления знаний и является структурой для восприятий пространственных сцен. Как называются элементы, из которых состоит внутренняя структура фрейма?

Согласована на портале № 43666

1) атомы
2) списки
3) атрибуты
4) слоты

17. Обычно, при описании иерархической структуры знаний используются фреймы. Каким образом задается положение фрейма в иерархической структуре?
1) фрейм содержит указатель на список потомков
2) фреймовая структура содержит матрицу связности фреймов
3) фрейм содержит слоты - указатели на родительский фрейм и список дочерних фреймов
4) фреймовая структура представляет собой многоуровневые вложенные списки: каждый вложенный фрейм является потомком того фрейма, куда он включен.

18. Семантическая сеть представляет собой информационную модель предметной области и имеет вид ориентированного графа. Чему соответствуют вершины и дуги этого графа?
1) вершины – объекты, дуги – свойства этих объектов
2) вершины – понятия предметной области, дуги – отношения между ними
3) вершины – объекты, дуги – события, происходящие с объектами
4) вершины – объекты, дуги – отношения между объектами

19. Семантические сети и фреймы являются моделями представления знаний и работают как с числовой, так и с символьной информацией. Функциональные языки предоставляют удобный инструментарий представления и обработки знаний. Какие объекты языка Лисп наиболее удобны для представления фреймов и семантических сетей?
1) списки свойств
2) символьные и числовые атомы
3) ассоциативные списки
4) S-выражения

20. Вывод в семантической сети определяется с помощью использующих ее процедур. Наиболее типичный способ вывода основан на сопоставлении частей сетевой структуры. Какой механизм функционального программирования является основой такого вывода?
1) суперпозиция функций
2) рекурсия
3) использование функционалов
4) управляемость данными

14.1.2. Экзаменационные тесты

Приведены примеры типовых заданий из банка экзаменационных тестов, составленных по пройденным разделам дисциплины.

1. Какое понятие лежит в основе функционального программирования?
1. функционал;
2. функция;
3. программа;
4. рекурсия;

2. Какие из перечисленных методов являются основными в функциональном программировании?
1. суперпозиция функций;
2. модульность;
3. последовательность действий;
4. функциональная взаимозаменяемость;
5. рекурсия.

3. Какие из перечисленных требований относятся к строго функциональному языку?

Согласована на портале № 43666

1. отсутствуют циклы;
2. отсутствует оператор присваивания;
3. любая функция должна однозначно определять результат по любому набору аргументов;
4. в языке присутствуют функции и процедуры;
5. в языке присутствуют функционалы.

4 Что будет выдано программой в результате следующей последовательности вызова вычислимых выражений? (В случае ошибки ответ запишите как Error)

```
> (defun f (x y) (setq x 3) (+ x y))
F
> (f 5 7)
10
> X
?
```

5. Что будет выдано программой в результате следующей последовательности вызова вычислимых выражений? (В случае ошибки ответ запишите как Error)

```
> (setq x 123)
Q
> (defun f (y) (setq x 10) (+ x y))
F
> (f 1)
11
> x
?
```

6. Что будет выдано программой в результате следующей последовательности вызова вычислимых выражений? (В случае ошибки ответ запишите как Error)

```
> (defun f (y) (setq x 10) (+ x y))
F
> (* (f 3) x)
?
```

7. Что будет выдано программой в результате следующей последовательности вызова вычислимых выражений? (В случае ошибки ответ запишите как Error)

```
> (setq x 12)
12
> (defun q (z) (setq x 7) (- z x))
Q
> (q 5)
-2
> x
?
```

8 Что выполняет следующая функция, аргументом которой является список?

```
(defun f (l) (append (cons (car l) nil) (cdr l) ))
```

1) возвращает список из первых двух элементов исходного списка и атома NIL;
2) возвращает список из первых двух элементов исходного списка;
3) возвращает атом NIL;
4) возвращает исходный список.

9. Что выполняет следующая функция, аргументом которой является список?

```
(defun f (l) (+ (cadr l) (caddr l)))
```

1) считает сумму из первых двух элементов исходного списка;
2) считает сумму из первого и третьего элементов списка;
3) считает сумму из второго и третьего элементов списка;

Согласована на портале № 43666

4) считает сумму элементов списка.

10. Что выполняет следующая функция, аргументом которой являются три числа?
(defun f (a b c) (cond ((> a 0) (+ a b c)) ((zerop a) (* a b c))))
1) возвращает сумму и произведение заданных чисел;
2) если первое число положительно – считает сумму чисел, иначе - произведение;
3) если первое число положительно – считает сумму чисел, если равно нулю – произведение, иначе – возвращает NIL;
4) если первое число положительно – считает сумму чисел, если отрицательно – произведение, иначе – возвращает NIL.

11. В каком случае данный предикат, аргумент которого является списком, вернет true?
(defun f (l) (cond ((atom (car l)) nil) ((numberp (car l)) nil) (t t)))
1) если первый элемент списка список;
2) если первый элемент списка атомом;
3) если первый элемент списка не число;
4) если первый элемент списка не атом.

12. Что является значением функции (PRINT X)?
1) значение выражения X;
2) печать выражения X;
3) печать символа «X»;
4) переход на новую строку и печать X.

13. Что считается окончанием ввода для функции READ?
1) переход на новую строку;
2) пробел или закрывающаяся скобка;
3) пробел или открывающаяся скобка;
4) Enter.

14. Что является побочным эффектом функции (PRINT X)?
1) значение выражения X;
2) двойной вывод значения X;
3) печать символа «X»;
4) печать значения X.

15. Всегда ли при печати выводится две дублирующие строки?
1) нет, всегда только одна;
2) нет, только если печать происходит на высшем уровне;
3) да, всегда две дублирующие;
4) нет, только при печати списков.

16. Что делает функция (PRINC X)?
1) печатает значение выражения X с переводом на другую строку;
2) печатает значение выражения X без перевода на другую строку;
3) печатает символ «X» с переводом на другую строку;
4) печатает символ «X» без перевода на другую строку.

17. Что делает разрушающая функция (RPLACA X E)?
1) заменяет левую ссылку в первом звене списка X ссылкой на выражение E;
2) заменяет правую ссылку в первом звене списка X ссылкой на выражение E;
3) заменяет ссылку в последнем звене списка X ссылкой на E;
4) заменяет в списке X все пустые списки на E.

Согласована на портале № 43666

18. Что делает разрушающая функция (RPLACD X E)?
1) заменяет левую ссылку в первом звене списка X ссылкой на выражение E;
2) заменяет правую ссылку в первом звене списка X ссылкой на выражение E;
3) заменяет ссылку в последнем звене списка X ссылкой на E;
4) заменяет в списке X все пустые списки на E.

19. Что делает разрушающая функция (NCONC X E)?
1) заменяет левую ссылку в первом звене списка X ссылкой на выражение E;
2) заменяет правую ссылку в первом звене списка X ссылкой на выражение E;
3) заменяет ссылку в последнем звене списка X ссылкой на E;
4) заменяет в списке X все пустые списки на E.

20. Какая функция используются только в PROG-выражениях?
1) LET;
2) GO;
3) COND;
4) RETURN;
5) DO.

14.1.3. Темы контрольных работ

Функциональное программирование и интеллектуальные системы

1. Как называется часть списка без первого элемента?
1. голова;
2. хвост;
3. ячейка;
4. функционал;
5. атом.

2. Какое обозначение имеет пустой список?
1. TRUE;
2. T;
3. NIL;
4. FALSE;
5. (NIL).

3 В рекурсивном определении важен порядок следования условий. Какая ветвь должна быть первой в COND?
1. проверка на NULL;
2. рекурсивная;
3. терминальная;
4. функциональная;
5. проверка на истину.

4. Какие задачи мы должны решить при планировании рекурсивной ветви?
1. когда функция может вернуть значение без рекурсивного вызова;
2. в каком порядке поставить терминальную и рекурсивную ветви;
3. определить соотношение между терминальной и рекурсивной ветвями;
4. как упрощать аргумент, приближая его к конечному значению;
5. как построить форму, называемую рекурсивным отношением.

5 Что выполняет следующая рекурсивная функция, аргументом которой является список?
(defun q (z) (cond ((null (cdr z)) 1) (t (* (cadr z) (q (cddr z))))))
1) перемножает элементы списка;
2) перемножает элементы списка, стоящие на нечетных позициях;

Согласована на портале № 43666

3) перемножает элементы списка, стоящие на четных позициях;
4) перемножает элементы списка, не равные нулю.

6 Что выполняет следующая рекурсивная функция, аргументами которой являются два списка?
(defun q (a b) (cond ((null a) nil) (t (cons (+ (car a)(car b)) (q (cdr a)(cdr b))))))
1) формирует список из поэлементных сумм исходных списков;
2) создает список из двух сумм элементов – 1-го и 2-го списка;
3) суммирует все элементы в обоих списках;
4) подсчитывает количество элементов в кратчайшем списке.

7 Что является результатом работы функции (MAKE-ARRAY N)?
1) создание массива ячеек размерности N
2) создание списка из атомов NIL размерностью N
3) создание массива с именем N
4) объявление массива размерностью N

8 Определите порядок приведенных ниже выражений, в результате которого должен сформироваться следующий список свойств: (a 1 s (2 3) d qwe f 5)
1) (setf (get `sp `s) `(2 3))
2) (setf (get `sp `f) 5)
3) (setf (get `sp `a) 1)
4) (setf (get `sp `d) qwe)

9 Определите порядок приведенных ниже выражений, в результате которого должен сформироваться следующий список свойств: (k1 45 k2 asd k3 (z x c) k4 1)
1) (setf (get `sp `k4) 1)
2) (setf (get `sp `k3) `(z x c))
3) (setf (get `sp `k1) 45)
4) (setf (get `sp `k2) asd)

10 Создан массив ячеек A. Его значение равно: #(a s d f)
Чему будет равно значение переменной X после выполнения следующего вычислимого выражения?
(setq X (cons (aref A 3) `(1 2)))

14.1.4. Методические рекомендации

Учебный материал излагается в форме, предполагающей самостоятельное мышление студентов, самообразование. При этом самостоятельная работа студентов играет решающую роль в ходе всего учебного процесса.

Начать изучение дисциплины необходимо со знакомства с рабочей программой, списком учебно-методического и программного обеспечения. Самостоятельная работа студента включает работу с учебными материалами, выполнение контрольных мероприятий, предусмотренных учебным планом.

В процессе изучения дисциплины для лучшего освоения материала необходимо регулярно обращаться к рекомендуемой литературе и источникам, указанным в учебных материалах; пользоваться через кабинет студента на сайте Университета образовательными ресурсами электронно-библиотечной системы, а также общедоступными интернет-порталами, содержащими научно-популярные и специализированные материалы, посвященные различным аспектам учебной дисциплины.

При самостоятельном изучении тем следуйте рекомендациям:

- чтение или просмотр материала необходимо осуществлять медленно, выделяя основные идеи; на основании изученного составить тезисы. Освоив материал, попытаться соотнести теорию с примерами из практики;

- если в тексте встречаются термины, следует выяснить их значение для понимания даль-

Согласована на портале № 43666

нейшего материала;

- необходимо осмысливать прочитанное и изученное, отвечать на предложенные вопросы.

Студенты могут получать индивидуальные консультации с использованием средств телекоммуникации.

По дисциплине могут проводиться дополнительные занятия в форме вебинаров. Расписание вебинаров публикуется в кабинете студента на сайте Университета. Запись вебинара публикуется в электронном курсе по дисциплине.

14.2. Требования к оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусмотрены дополнительные оценочные материалы, перечень которых указан в таблице 14.

Таблица 14 – Дополнительные материалы оценивания для лиц с ограниченными возможностями здоровья и инвалидов

Категории обучающихся	Виды дополнительных оценочных материалов	Формы контроля и оценки результатов обучения
С нарушениями слуха	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы	Преимущественно письменная проверка
С нарушениями зрения	Собеседование по вопросам к зачету, опрос по терминам	Преимущественно устная проверка (индивидуально)
С нарушениями опорно-двигательного аппарата	Решение дистанционных тестов, контрольные работы, письменные самостоятельные работы, вопросы к зачету	Преимущественно дистанционными методами
С ограничениями по общемедицинским показаниям	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы, устные ответы	Преимущественно проверка методами исходя из состояния обучающегося на момент проверки

14.3. Методические рекомендации по оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусматривается доступная форма предоставления заданий оценочных средств, а именно:

- в печатной форме;
- в печатной форме с увеличенным шрифтом;
- в форме электронного документа;
- методом чтения ассистентом задания вслух;
- предоставление задания с использованием сурдоперевода.

Лицам с ограниченными возможностями здоровья и инвалидам увеличивается время на подготовку ответов на контрольные вопросы. Для таких обучающихся предусматривается доступная форма предоставления ответов на задания, а именно:

- письменно на бумаге;
- набор ответов на компьютере;
- набор ответов с использованием услуг ассистента;
- представление ответов устно.

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

Для лиц с нарушениями зрения:

- в форме электронного документа;
- в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:

Согласована на портале № 43666

– в форме электронного документа;
– в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:

– в форме электронного документа;
– в печатной форме.

При необходимости для лиц с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения может проводиться в несколько этапов.

Согласована на портале № 43666