

Бюджетное учреждение высшего образования
Ханты-Мансийского автономного округа-Югры
"Сургутский государственный университет"

УТВЕРЖДАЮ
Проректор по УМР

_____ Е.В. Коновалова

16 июня 2022 г., протокол УС №6

МОДУЛЬ ПРОГРАММНО-АЛГОРИТМИЧЕСКОГО ОБЕСПЕЧЕНИЯ

Функциональное программирование рабочая программа дисциплины (модуля)

Закреплена за кафедрой **Автоматики и компьютерных систем**

Учебный план bz090304-ПОКС-22-2.plx
09.03.04 ПРОГРАММНАЯ ИНЖЕНЕРИЯ
Направленность (профиль): Программное обеспечение компьютерных систем

Квалификация **Бакалавр**

Форма обучения **заочная**

Общая трудоемкость **3 ЗЕТ**

Часов по учебному плану 108

в том числе:

аудиторные занятия 16

самостоятельная работа 83

часов на контроль 9

Виды контроля на курсах:
экзамены 2

Распределение часов дисциплины по курсам

Курс	1		2		Итого	
	уп	рп	уп	рп		
Лекции	2	2	4	4	6	6
Лабораторные	4	4	6	6	10	10
Итого ауд.	6	6	10	10	16	16
Контактная работа	6	6	10	10	16	16
Сам. работа	30	30	53	53	83	83
Часы на контроль			9	9	9	9
Итого	36	36	72	72	108	108

Программу составил(и):

к.т.н., доцент, Даниленко И. Н.

Рабочая программа дисциплины

Функциональное программирование

разработана в соответствии с ФГОС:

Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.04 Программная инженерия (приказ Минобрнауки России от 19.09.2017 г. № 920)

составлена на основании учебного плана:

09.03.04 ПРОГРАММНАЯ ИНЖЕНЕРИЯ

Направленность (профиль): Программное обеспечение компьютерных систем

утвержденного учебно-методическим советом вуза от 16.06.2022 протокол № 6.

Рабочая программа одобрена на заседании кафедры

Автоматики и компьютерных систем

Зав. кафедрой к.т.н., доцент кафедры автоматике и компьютерных систем Запевалов А. В.

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ	
1.1	Цели дисциплины:
1.2	- создать у обучающегося представление о парадигме функционального программирования, а также влиянии этой парадигмы на развитие современных языков высокого уровня, технологий и средствах разработки программного обеспечения;
1.3	- сформировать знания о синтаксисе и семантике конструкций языка программирования Common Lisp, назначении и составе средств и библиотек языка;
1.4	- сформировать навыки использования средств функционального программирования при решении прикладных вычислительных задач.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП	
Цикл (раздел) ООП:	Б1.В.03
2.1	Требования к предварительной подготовке обучающегося:
2.1.1	Алгебра и геометрия
2.1.2	Математический анализ
2.1.3	Дискретная математика
2.1.4	Программирование и основы алгоритмизации
2.1.5	Иностранный язык
2.2	Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:
2.2.1	Технология разработки программного обеспечения
2.2.2	Программирование мобильных устройств
2.2.3	Параллельное программирование
2.2.4	Программирование на языке Python

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)
ПК-4.2: Использует современные технологии разработки ПО
ОПК-6.1: Разрабатывает алгоритмическое и программное обеспечение на основных языках программирования высокого уровня

В результате освоения дисциплины обучающийся должен

3.1	Знать:
3.1.1	особенности парадигмы функционального программирования;
3.1.2	основы синтаксиса и семантики языка программирования высокого уровня Common Lisp;
3.2	Уметь:
3.2.1	применять ключевые идеи функционального программирования при разработке алгоритмов и программ;
3.2.2	использовать функции как объекты первого рода;
3.2.3	использовать встроенные и разрабатывать пользовательские функции высокого порядка;
3.3	Владеть:
3.3.1	принципами парадигмы функционального программирования при проектировании, конструировании и тестировании программных продуктов на языках программирования высокого уровня;
3.3.2	базовыми навыками использования языка Common Lisp при разработке программного обеспечения;

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)						
Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Компетенции	Литература	Примечание
	Раздел 1. Функциональное программирование					
1.1	Декларативное программирование. Лямбда-исчисление. /Лек/	2	2	ПК-4.2 ОПК-6.1	Л1.1 Л1.2Л3.1 Э1 Э2 Э3 Э4 Э5	
1.2	Основы языка Common Lisp: типы данных, лямбда-функции, формы, универсальные и базовые функции. /Лек/	2	1	ПК-4.2 ОПК-6.1	Л1.1 Л1.2Л3.1 Э1 Э2 Э3 Э4 Э5	

1.3	Организация вычислений в Common Lisp. Условные вычисления. Рекурсия. Хвостовая рекурсия. /Лек/	2	1	ПК-4.2 ОПК-6.1	Л1.1 Л1.2Л3.1 Э1 Э2 Э3 Э4 Э5	
1.4	Функции высокого порядка. Встроенные функционалы. Функции как объекты первого рода. /Лек/	2	2	ПК-4.2 ОПК-6.1	Л1.1 Л1.2Л3.1 Э1 Э2 Э3 Э4 Э5	
1.5	Знакомство со средой LispWorks /Лаб/	1	2	ПК-4.2 ОПК-6.1	Л1.1 Л1.2 Л1.3Л3.1 Э1 Э3 Э4 Э5	
1.6	Организация вычислений в Common Lisp /Лаб/	1	2	ПК-4.2 ОПК-6.1	Л1.1 Л1.2 Л1.3Л3.1 Э1 Э3 Э4 Э5	
1.7	Условные вычисления Common Lisp /Лаб/	2	1	ПК-4.2 ОПК-6.1	Л1.1 Л1.2 Л1.3Л3.1 Э1 Э3 Э4 Э5	
1.8	Повторные вычисления в Common Lisp /Лаб/	2	1	ПК-4.2 ОПК-6.1	Л1.1 Л1.2 Л1.3Л3.1 Э1 Э3 Э4 Э5	
1.9	Рекурсивные вычисления в Common Lisp /Лаб/	2	2	ПК-4.2 ОПК-6.1	Л1.1 Л1.2 Л1.3Л3.1 Э1 Э3 Э4 Э5	
1.10	Функционалы /Лаб/	2	2	ПК-4.2 ОПК-6.1	Л1.1 Л1.2 Л1.3Л3.1 Э1 Э3 Э4 Э5	
1.11	Обзор литературы по темам "Декларативное программирование", "Лямбда-исчисление" /Ср/	2	30	ПК-4.2 ОПК-6.1	Л1.1 Л1.2Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5	
1.12	Подготовка по теме "Основы языка Common Lisp: типы данных, лямбда-функции, формы, универсальные и базовые функции" и к выполнению лабораторных работ "Знакомство со средой LispWorks", "Организация вычислений в Common Lisp" /Ср/	2	10	ПК-4.2 ОПК-6.1	Л1.1 Л1.2Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5	
1.13	Подготовка по темам "Организация вычислений в Common Lisp. Условные вычисления. Рекурсия. Хвостовая рекурсия." и к выполнению лабораторных работ "Повторные вычисления в Common Lisp", "Рекурсивные вычисления в Common Lisp" /Ср/	2	23	ПК-4.2 ОПК-6.1	Л1.1 Л1.2Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5	
1.14	Подготовка по темам "Функции высокого порядка. Встроенные функционалы. Функции как объекты первого рода." и к выполнению лабораторной работы "Функционалы" /Ср/	2	20	ПК-4.2 ОПК-6.1	Л1.1 Л1.2Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5	
1.15	/Экзамен/	2	9	ПК-4.2 ОПК-6.1	Л1.1 Л1.2 Л1.3Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5	Экзамен

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

5.1. Контрольные вопросы и задания

Представлено отдельным документом

5.2. Темы письменных работ

Представлено отдельным документом

5.3. Фонд оценочных средств

Представлено отдельным документом

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)**6.1. Рекомендуемая литература****6.1.1. Основная литература**

	Авторы, составители	Заглавие	Издательство, год	Колич-во
Л1.1	Городня Л. В.	Основы функционального программирования: курс лекций	М.: Интернет-Университет Информационных Технологий, 2004	13
Л1.2	Рогозин О. В.	Функциональное и рекурсивно-логическое программирование: Учебное пособие	Москва: Евразийский открытый институт, 2009, электронный ресурс	1
Л1.3	Галкина М.Ю.	Функциональное и логическое программирование: практикум	Новосибирск: Сибирский государственный университет телекоммуникаций и информатики, 2008, электронный ресурс	1

6.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год	Колич-во
Л2.1	Филд А., Харрисон П.	Функциональное программирование	М.: Мир, 1993	8

6.1.3. Методические разработки

	Авторы, составители	Заглавие	Издательство, год	Колич-во
Л3.1	Даниленко И. Н.	Основы функционального и логического программирования: учебное пособие	Сургут: Издательский центр СурГУ, 2009	50

6.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	Березин, Н. Введение в программирование на Лиспе [Электронный ресурс] / Н. Березин, Л. Городня. — Электрон. дан. – Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), 2007. — Режим доступа: http://www.intuit.ru/studies/courses/1026/158/info			
Э2	Сошников, Д. Функциональное программирование [Электронный ресурс] / Д. Сошников. — Электрон. дан. – Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), 2010. — Режим доступа: http://www.intuit.ru/studies/courses/471/327/info			
Э3	«Хабрахабр» [Электронный ресурс]. – 201-. – Режим доступа: http://habrahabr.ru/ , свободный. – Загл. с экрана.			
Э4	Graham, P. On Lisp. [Электронный ресурс] / P. Graham. – Prentice Hall, 1993. – 432 p. - Режим доступа: http://www.paulgraham.com/onlisptext.html			
Э5	Steel Bank Common Lisp [Электронный ресурс]. – 201-. – Режим доступа: http://www.sbcl.org/ , свободный			

6.3.1 Перечень программного обеспечения

6.3.1.1	Операционная система Microsoft Windows
6.3.1.2	Adobe Acrobat Reader
6.3.1.3	Microsoft Office
6.3.1.4	веб-браузер
6.3.1.5	LispWorks® Personal Edition, LispIDE

6.3.2 Перечень информационных справочных систем

6.3.2.1	Национальная электронная библиотека (НЭБ) https://rusneb.ru/
6.3.2.2	LispWorks® Personal Edition, LispIDE и др.
6.3.2.3	СПС «КонсультантПлюс» - www.consultant.ru/
6.3.2.4	СПС «Гарант» - www.garant.ru/

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1	Помещение для проведения лабораторных занятий (компьютерный класс)
7.2	Компьютеры
7.3	Доступ к сети Интернет и ресурсам локальной сети СурГУ
7.4	Мультимедийное оборудование (экран, проектор)

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
для промежуточной аттестации по дисциплине

Функциональное программирование

Код, направление подготовки	09.03.04 Программная инженерия
Направленность (профиль)	Программное обеспечение компьютерных систем
Форма обучения	Заочная
Кафедра разработчик	Автоматики и компьютерных систем
Выпускающая кафедра	Автоматики и компьютерных систем

Типовые задания для контрольной работы

Тема 1. Декларативное программирование. Лямбда-исчисление.

Цель сформировать представление о декларативной парадигме программирования и в частности, функциональном программировании, как одной из современных методологий разработке программного обеспечения. Лямбда-исчисление как математическая основа функционального программирования. Нормальный и аппликативный порядок вычислений.

Примеры вопросов контрольной работы:

- в чем принципиальное отличие императивной парадигмы от декларативной?
- каким образом осуществляются повторные вычисления в функциональных программах при отсутствии состояния программы и «операции присваивания»?
- какого назначения анонимных функций?
- выполнить редукцию терма $(\lambda xyz.xyz)(\lambda x.xx)(\lambda x.x)x$

Тема 2. Основы языка Common Lisp: типы данных, лямбда-функции, формы, универсальные и базовые функции.

Примеры вопросов контрольной работы:

- Построить функцию, реализующую математическое выражение $\sqrt{a^2 + b^2 + c^2}$.
- Построить функцию, реализующую математическое выражение $\frac{a^2 + b^2}{c^2}$.
- Построить функцию, реализующую математическое выражение $\frac{\sin a + \sin b + \sin c}{\cos abc}$.

Тема 3. Организация вычислений в Common Lisp. Условные вычисления. Рекурсия. Хвостовая рекурсия.

Примеры вопросов контрольной работы:

- Построить функцию, реализующую математическое выражение так, чтобы повторные вычисления не осуществлялись $\frac{\sin ab}{\cos^2 ab}$

– Определена функция

```
(defun f(x) (if (null (cdr x)) nil (cons (f (cdr x)) (cons (car x) (cdr x)))))
```

Определить, какое значение вернет вызов:

```
(f '(a b c))
```

– Реализовать рекурсивную функцию (не используя специальные функции и функционалы), получающую список (четной длины) чисел и возвращающую список пар исходных элементов

$(1\ 2\ 3\ 4) \Rightarrow ((1\ 2)\ (3\ 4))$

– Построить функцию, реализующую математическое выражение так, чтобы повторные вычисления не осуществлялись $\frac{abc}{abc + \ln ab}$

– Определена функция

```
(defun f(x) (if (null x) () (cons (car x) (f (cdr x)))))
```

Определить, какое значение вернет вызов:

```
(f '(a b c))
```

– Реализовать рекурсивную функцию (не используя специальные функции и функционалы), получающую список (четной длины) чисел и возвращающую список, содержащий элементы, которые находились на четных позициях исходного списка

$(1\ 2\ 3\ 4) \Rightarrow (2\ 4)$

Тема 4. Функции высокого порядка. Встроенные функционалы. Функции как объекты первого рода.

Примеры вопросов контрольной работы:

– Построить функционал, получающий две функции одного аргумента и возвращающий функцию одного аргумента, представляющую композицию функций (f1 (f2)).

– Построить функционал, получающий две функции одного аргумента и возвращающий функцию одного аргумента, представляющую сумму функций (f1 + f2).

– Построить функционал, получающий две функции одного аргумента и возвращающий функцию одного аргумента, представляющую произведение функций (f1 * f2).

Типовые задания для экзамена

1. Декларативное и императивное программирование. Языки функционального программирования.
2. Лямбда-исчисление Черча.
3. Основные элементы семантики и синтаксиса языка Lisp. Области применения функционального программирования.
4. Структурные единицы Lisp. Понятия атомов, списков и s-выражений.
5. Базовые функции языка Lisp. Основные функции обработки списков.
6. Точечная пара. Предикаты. Дополнительные функции обработки списков.
7. Арифметические функции. Логические функции.
8. Понятие символа. Функции связывания символа и его значения.
9. Анонимные функции, λ -выражения и λ -вызовы. Именованные функции (функции пользователя). Ключевые слова. Функциональная блокировка.
10. Организация ввода-вывода средствами языка Lisp. Функции ввода-вывода. Форматированный ввод-вывод.

11. Управляющие конструкции: последовательные вычисления. Разветвление вычислений.
12. Управляющие конструкции: условные предложения. Циклические вычисления.
13. Функционалы. Отображающие и применяющие функционалы.
14. Функции высоких порядков.
15. Энергичные и ленивые вычисления.
16. Определение рекурсии. Трассировка. CAR- и CDR-рекурсия.
17. Множественные терминальные ветви. Множественные рекурсивные ветви.
18. Представление списков через списочные ячейки и через точечные пары. Сборка мусора.