

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Б1.В.11
(индекс дисциплины)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Функциональное программирование

(наименование дисциплины)

по направлению подготовки

09.03.03 Прикладная информатика

направленность (профиль)

Разработка программного обеспечения

Форма обучения: заочная

Год набора: 2022

Общая трудоемкость: ЗЕ 5

Распределение часов дисциплины по семестрам

Семестр	6	Итого
Форма контроля	Зачет с оценкой	
Вид занятий		
Лекции	4	4
Лабораторные		
Практические	6	6
Руководство: курсовые работы (проекты) / РГР		
Промежуточная аттестация	0,25	0,25
Контактная работа	10,25	10,25
Самостоятельная работа	166	166
Контроль	3,75	3,75
Итого	180	180

Рабочую программу составил(и):
доцент кафедры «Прикладная математика и информатика», доцент, к.п.н. Оськина О.В.
(должность, ученое звание, степень, Фамилия И.О.)

(должность, ученое звание, степень, Фамилия И.О.)

Рецензирование рабочей программы дисциплины:



Отсутствует



Рецензент

(должность, ученое звание, степень, Фамилия И.О.)

Рабочая программа дисциплины составлена на основании ФГОС ВО и учебного плана
направления подготовки

09.03.03 Прикладная информатика

Срок действия рабочей программы дисциплины до «31» августа 2027 г.

УТВЕРЖДЕНО

На заседании кафедры «Прикладная математика и информатика»

(протокол заседания № 2 от «15» сентября 2021 г.).

1. Цель освоения дисциплины

Цель освоения дисциплины – формирование знаний и умений в области методов представления данных в памяти компьютера, основных алгоритмов, оперирующих с ними, а также овладение первичными навыками разработки, отладки и тестирования программ.

2. Место дисциплины в структуре ОПОП ВО

Дисциплины и практики, на освоении которых базируется данная дисциплина:
Ведение в профессию

Дисциплины и практики, для которых освоение данной дисциплины необходимо как предшествующее:

Объектно-ориентированное программирование

6. Планируемые результаты обучения

Формируемые и контролируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Планируемые результаты обучения
Способен разрабатывать и адаптировать прикладное программное обеспечение (ПК-3)	ПК-3.1 Знает современные технологии разработки и адаптации прикладного программного обеспечения, их достоинства и недостатки	Знать: методы разработки, внедрения и адаптации прикладного программного обеспечения
	ПК-3.2 Умеет разрабатывать, адаптировать компоненты прикладного программного обеспечения	Уметь: разрабатывать, внедрять и адаптировать прикладное программное обеспечение
	ПК-3.3 Владеет навыками разработки прикладного программного обеспечения на современных языках программирования, методами адаптации прикладного программного обеспечения	Владеть: навыками разработки, внедрения и адаптации прикладного программного обеспечения в различных сферах профессиональной деятельности
Способен выполнять разработку и отладку программного кода (ПК-6)	ПК-6.1 Знает методы и приемы отладки программного кода, типы и форматы сообщений об ошибках, предупреждениях, виды современных компиляторов, отладчиков программного кода.	Знать: основные этапы кодирования, основы тестирования
	ПК-6. Умеет выявлять ошибки в программном коде, применять методы и средства проверки работоспособности программного кода, интерпретировать сообщения об ошибках.	Уметь: реализовывать читаемость программного кода и документационное сопровождение процесса кодирования программ
	ПК-6.3 Владеет навыками отладки программного кода, работы в	Владеть: основными приемами разработки

Формируемые и контролируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Планируемые результаты обучения
	современных компиляторах, работы в отладчиках и оптимизаторах программного кода.	программного кода и тестирования программных решений

4. Структура и содержание дисциплины

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наименование оценочного средства)
Модуль 1	Лек	Базовые средства языка Haskell	6	2		-	
	Лек	Базовые алгоритмические структуры	6	2		-	
	Пр	Операции и стандартные библиотечные функции языка	6	2	24	-	Отчет по практической работе
	Пр	Базовые алгоритмические структуры		4	24	-	Отчет по практической работе
Итого:				10	48		

Схема расчета итогового балла

Текущий рейтинг (все занятия и промежуточные тесты) + Результат итогового теста и все делится на 2

5. Образовательные технологии

При изучении дисциплины используются следующие образовательные технологии:

- технологии традиционного обучения в форме лекций, практических работ и самостоятельной работы студентов;

6. Методические указания по освоению дисциплины

В организации работы студентов очной формы обучения над изучением учебного курса «Программирование на языках высокого уровня» важное место принадлежит аудиторным занятиям. В них излагается общая характеристика вопросов темы.

Практические занятия проводятся по наиболее сложным теоретическим проблемам дисциплины.

На каждом последующем практическом занятии студенты, при ответе на проблемные вопросы и в ходе выполнения сложных заданий, должны использовать знания, полученные при изучении предшествующих тем. Основным источником информации при подготовке к практическим занятиям является основная и дополнительная литература.

7. Оценочные средства

7.1. Паспорт оценочных средств

Семестр	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1	ПК-3, ПК-6	Тестовые задания №1-500 Вопросы к экзамену №1-50

7.2. Типовые задания или иные материалы, необходимые для текущего контроля

7.2.1.

Практическое задание

(наименование оценочного средства)

Типовой(ые) пример(ы) задания(ий)

1. Вычислить и вывести на печать значение функции Y . Исходные данные X , A и B ввести с клавиатуры.

$$y = (a + b) * a^{b+1} * \operatorname{tg} x$$

2. Даны три действительных числа. Возвести в квадрат те из них, значения которых неотрицательны.
3. Даны натуральное n . Вычислить сумму n членов ряда, общий член которого равен

$$A_k = \frac{(-1)^k \times (k + 1)}{k!} \quad k = 1, 2, \dots, n$$

4. Составить программу расчета таблицы значений функции $f(x)$ на интервале $a \leq x \leq b$ в n равностоящих точках. Границы интервала a, b и количество точек n ввести с клавиатуры. Результаты вывести на печать.

$$f(x) = 1 + \sin(2\pi x)$$

Найти количество положительных значений функции $f(x)$ на заданном интервале.

5. Даны натуральное число n , действительные числа a_1, \dots, a_n . Если в результате замены отрицательных элементов массива a_1, \dots, a_n их квадратами элементы будут образовывать неубывающую последовательность, то получить сумму элементов исходного массива, в противном случае получить их произведение.
6. Дана действительная матрица размером $n \times m$. Все элементы с наибольшим значением заменить нулями (таких элементов может быть несколько).
7. Составить программу с использованием функций на базе задания 8.
8. Написать программу, которая вводит с клавиатуры данные, содержащие информацию: фамилия студента, год рождения, оценки по 5-ти экзаменам и создает массив структур, отсортированный в порядке возрастания годов рождения.
9. Дан файл, компоненты которого являются действительными числами. Найти модуль суммы и квадрат произведения компонент файла.
10. Дан файл, содержащий сведения об ассортименте игрушек в магазине. Каждая компонента файла содержит информацию – название игрушки, цена, количество, возрастные границы, например, для детей от 2 до 5 лет. Написать программу, выдающую информацию следующего вида:
 - названия игрушек, которые подходят детям от 1 до 3 лет;
 - стоимость самой дорогой игрушки и ее наименование;

- название игрушки, которая по стоимости не превышает X руб. и подходит ребенку в возрасте от A до B лет.

Значения X, A, B вводятся по запросу с клавиатуры.

11. Даны натуральное число n, символы s_1, \dots, s_n , среди которых есть хотя бы одна точка. Преобразовать последовательность s_1, \dots, s_n заменив знаком «+» все цифры 3, встречающиеся после первой точки.
12. Найти самое длинное симметричное слово заданного предложения, например АССА.

7.2.2.

Тестовые задания

1. Языки программирования высокого уровня характеризует:
 - ☒ наличие понятия типа данных и близость к естественному языку
 - ☐ зависимость от архитектуры конкретного компьютера
 - ☐ близость к машинному языку
 - ☐ высокая стоимость системы программирования
2. Основное отличие компиляторов от интерпретаторов:
 - ☒ компиляторы генерируют целевой код, интерпретаторы нет
 - ☐ интерпретаторы генерируют целевой код, компиляторы нет
 - ☐ компиляторы в процессе трансляции исполняют исходный код, а интерпретаторы нет
 - ☐ компиляторы участвуют в построении исходного кода, а интерпретаторы нет
3. Стадия трансляции, на которой может быть выявлена ошибка “пропущена точка с запятой”:
 - ☒ на стадии синтаксического анализа
 - ☐ на стадии лексического анализа
 - ☐ на стадии семантического анализа
 - ☐ на стадии оптимизации кода
4. Стадия трансляции занимается проверкой типов в выражениях:
 - ☒ семантический анализ
 - ☐ синтаксический анализ
 - ☐ лексический анализ
 - ☐ генерация кода
5. Таблица символов в процессе трансляции:
 - ☒ для хранения имён переменных и имён функций
 - ☐ для хранения значений переменных
 - ☐ для хранения результатов выполнения процедур
 - ☐ для хранения значений констант
6. Какие из следующих выражений типизированы верно, то есть не приводят к ошибкам типа?
 - ☒ $(3 :: \text{Int}) + (5 :: \text{Integer})$
 - ☐ $(3 :: \text{Double}) + (5 :: \text{Float})$

- ☐ (3 :: Integer) + (5 :: Integer)
- ☐ (3.2 :: Double) + (5 :: Double)
- ☒ (3.0 :: Integer) + (5 :: Integer)

7. Язык программирования – это:

8. специально созданная система обозначений слов, букв, чисел

- набор слов для написания программы
- определенная последовательность бит
- двоичные коды для компьютера

9. Языки программирования высокого уровня являются:

10. машинно-независимыми

- машинно-зависимыми
- набором нулей и единиц
- ограниченными по объему информации

11. Величины в языках программирования характеризуются:

12. типом, именем, значением

- однородностью, предназначением, полезностью
- местоположением, принадлежностью, значением
- элементами, размером, значением

13. Какие из следующих выражений типизированы верно, то есть не приводят к ошибкам типа?

14. алфавит, синтаксис, семантика

- алфавит, орфография, диалектика
- переменные, процедуры, функции
- модули, описания, реализация

15. Синтаксис языка программирования - это:

16. система правил, определяющая допустимые конструкции языка

- набор правил расстановки знаков препинания
- интерпретация отдельных языковых конструкций языка
- фиксированный набор основных символов, допускаемых для составления программы

17. Семантика языка программирования - это:

- система правил, определяющая допустимые конструкции языка

18. система правил однозначного истолкования языковых конструкций языка

- набор металингвистических формул Бэкуса-Наура
- синтаксическая конструкция, определяющая свойства программных объектов

19. «Массивом» называется:

- ограниченная апострофами последовательность любых символов
- самый простой оператор языка программирования
- набор переменных, начинающихся с одной буквы

20. последовательность фиксированного числа однотипных переменных, имеющих общее имя

Темы письменных работ

Письменные работы по курсу не предусмотрены.

7.3.Оценочные средства для промежуточной аттестации по итогам освоения дисциплины

7.3.1. Вопросы к промежуточной аттестации

Семестр _____ 6 _____

№ п/п	Вопросы к зачету с оценкой
1.	Алгоритм: определение, свойства и способы записи.
2.	Изобразительные средства алгоритмов: словесный, формульно-словесный, блок-схема.
3.	Транслятор: назначение, классификация, этапы процесса трансляции.
4.	Представление текстовых данных: таблицы кодировки.
5.	Идентификаторы: назначение и правила построения.
6.	Основные этапы компьютерного решения задач.
7.	Порядок разработки иерархических схем реализации алгоритмов.
8.	Носители информации и единицы измерения.
9.	Процедуры: создание и использование в программах.
10.	Операторы: понятие, назначение, классификация
11.	Переменные: назначение, классификация, описание в программе, работа с данными.
12.	Константы: назначение, классификация, описание в программе, работа с данными.
13.	Выражения, операнды, операции и их приоритет. Унарные, бинарные, тернарные операции.
14.	Стандартные функции для работы с файлами.
15.	Текстовые файлы: назначение, описание в программе, принципы работы с данными.
16.	Типизированные файлы: назначение, описание в программе, принцип работы с данными.
17.	Общая структура программы на языке высокого уровня Haskell
18.	Цикл с предусловием
19.	Цикл с параметром
20.	Цикл с постусловием
21.	Алгоритм ветвления
22.	Целый тип данных
23.	Синтаксис алгоритмических языков
24.	Определение языка программирования.
25.	Вещественные типы данных.
26.	Логический тип данных.
27.	Символьный тип данных.
28.	Структурированные типы данных: классификация, назначение, область применения.
29.	Массивы: назначение, классификация, описание в программе, принцип работы с данными.
30.	Ключевые слова, разделители и знаки операций.
31.	Организация диалога: процедура ввода данных с клавиатуры.
32.	Строки: назначение, описание в программе, принцип работы с данными.
33.	Организация диалога: процедура вывода данных на экран.
34.	Форматный вывод данных.
35.	Оператор присваивания.
36.	Условный оператор, его модификации.
37.	Оператор выбора.
38.	Логические вычисления и операции отношения.

№ п/п	Вопросы к зачету с оценкой
39.	Цикл: классификация, назначение, примеры использования в программах.
40.	Стандартные процедуры и функции языка С
41.	Указатели и динамическая память.
42.	Упорядочение массива.
43.	Назначение и организация комментариев в программах.
44.	Структурированные типы данных: классификация, назначение, область применения.
45.	Базовые алгоритмические структуры, используемые при проектировании алгоритмов линейных, ветвящихся и циклических вычислительных процессов, их виды и способы изображения.
46.	Массивы: назначение, классификация, описание в программе, принцип работы с данными.
47.	Структуры: назначение, описание в программе, принцип работы с данными.
48.	Процедуры: создание и использование в программах.
49.	Типизированные файлы: назначение, описание в программе, принцип работы с данными.
50.	Текстовые файлы: назначение, описание в программе, принципы работы с данными.

7.3.2. Критерии и нормы оценки

Семестр	Форма проведения промежуточной аттестации	Критерии и нормы оценки	
1	Зачет с оценкой по накопительному рейтингу	«отлично»	учащийся глубоко усвоил весь программный материал, исчерпывающе, последовательно, грамотно и логически его излагает, свободно справляется с практическими заданиями, проявляет знание источников, умеет ими пользоваться при ответах, умеет обобщать и излагать материал, не допуская ошибок.
		«хорошо»	учащийся знает программный материал, излагает его по существу, знает понятийный аппарат по теме вопроса, не допускает существенных упущений и неточностей в ответе на вопрос, может правильно применять теоретические положения и владеет необходимыми умениями и навыками при выполнении практических заданий.
		«удовлетворительно»	учащийся знает основной программный материал в минимальном объеме, знаком с основной рекомендованной литературой, недостаточно правильные формулировки, т.е.

Семестр	Форма проведения промежуточной аттестации	Критерии и нормы оценки	
			владеет программным материалом в объеме, необходимом для дальнейшей учебы и работы
		«неудовлетворительно»	учащийся обнаруживает существенные пробелы в знании основного программного материала, допускает принципиальные ошибки при применении теоретических знаний, которые не позволяют ему продолжить обучение или приступить к практической деятельности без дополнительной подготовки по дисциплине

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Обязательная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно-методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
1	Кубенский А. А.	Функциональное программирование: Учебник и практикум - М.: Издательство Юрайт	Учебник и практикум	2017	http://www.biblio-online.ru/book/658E3C89-AAD5-498B-8B34-A29E1750D810

8.2. Дополнительная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно-методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
1	Салмина Н.Ю	Функциональное программирование и интеллектуальные системы: учебное пособие - Томск: Томский государственный университет систем управления и радиоэлектроники	Учебное пособие	2016	http://www.iprbookshop1.ru/72216.html
2	Селиванова И. В. -	Основы функционального программирования [Электронный ресурс]: курс лекций - Курск: Изд-во Курск. гос. ун-та	Учебное пособие	2011	ftp://elibrary.kursksu.ru/1etrud/000681.pdf

8.3. Перечень профессиональных баз данных и информационных справочных систем

—

8.4. Перечень программного обеспечения

№ п/п	Наименование ПО	Реквизиты договора (дата, номер, срок действия)
1	Haskell Platform	50

8.5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

№ п/п	Наименование оборудованных учебных кабинетов, лабораторий, мастерских и др. объектов для проведения практических и лабораторных занятий, помещений для самостоятельной работы обучающихся (номер аудитории)	Перечень основного оборудования
	Компьютерный класс УЛК-314	Стол ученический-26 шт., стол преподавательский-1 шт., стулья-28 шт., доска аудиторная (меловая)-1шт., компьютер с выходом в сеть интернет- 17 шт.
	Лекционная аудитория Г-322	Стол ученический трехместный (моноблок) - 60 шт., стол преподавательский, стул преподавательский, доска аудиторная (меловая), кафедра напольная, экран навесной, стационарный проектор, процессор, мышь компьютерная пространственная, пульт для проектора
	Класс для самостоятельной работы Г-401	Стол ученический-26 шт., стул-26 шт., компьютер с выходом в сеть интернет- 16 шт.