

УТВЕРЖДАЮ

Первый проректор, проректор по
УР

_____ А.Е. Рудин

«30» июня 2020 года

Рабочая программа дисциплины

Б1.О.08

Методы современного программирования

Учебный план: ФГОС 3++_2020-2021_09.04.02_ВШПМ_ОО_ИТ в дизайне_2-1-40.plx

Кафедра: **21** Информационных и управляющих систем

Направление подготовки:
(специальность) 09.04.02 Информационные системы и технологии

Профиль подготовки: Информационные технологии в дизайне
(специализация)

Уровень образования: магистратура

Форма обучения: очная

План учебного процесса

Семестр (курс для ЗАО)	Контактная работа обучающихся		Сам. работа	Контроль, час.	Трудоёмкость, ЗЕТ	Форма промежуточной аттестации	
	Лекции	Практ. занятия					
2	УП	17	34	84	45	5	Экзамен
	РПД	17	34	84	45	5	
Итого	УП	17	34	84	45	5	
	РПД	17	34	84	45	5	

Рабочая программа дисциплины составлена в соответствии с федеральным государственным образовательным стандартом высшего образования по направлению подготовки 09.04.02 Информационные системы и технологии, утвержденным приказом Министерства образования и науки Российской Федерации от 19.09.2017 г. № 917

Составитель (и):

кандидат технических наук, Доцент

Белая Т.И.

От кафедры составителя:

Заведующий кафедрой информационных и
управляющих систем

Коваленко Александр
Николаевич

От выпускающей кафедры:

Заведующий кафедрой

Коваленко Александр
Николаевич

Методический отдел:

1 ВВЕДЕНИЕ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ

1.1 Цель дисциплины: Целями освоения учебной дисциплины «Методы современного программирования» являются:

а) формирование у будущих магистров теоретических знаний и практических навыков в разработке ПО для решения прикладных задач различных предметных областей;

б) показать значимость технологии разработки ПО как современного направления прикладной информатики;

в) развитие умений и навыков проектирования и разработки прикладных программ на языках высокого уровня;

г) знакомство с современными инструментальными системами для проектирования и разработки программ на языках программирования высокого уровня.

1.2 Задачи дисциплины:

Основными задачами изучения дисциплины являются:

- изучение принципов алгоритмизации и современных методов обработки информации с использованием языков программирования;

- ознакомление с техническими, алгоритмическими, программными и технологическими решениями, используемыми в данной области;

- выработка практических навыков аналитического и экспериментального исследования основных методов и средств, используемых в области, изучаемой в рамках данной дисциплины.

1.3 Требования к предварительной подготовке обучающегося:

Предварительная подготовка предполагает создание основы для формирования компетенций, указанных в п. 2, при изучении дисциплин:

Организация и управление исследованиями

Дополнительные главы информатики

2 КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

УК-6: Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки
Знать: теоретические основы и современные информационные технологии анализа, проектирования и разработки программного обеспечения
Уметь: проектировать и разрабатывать различные виды программного обеспечения на основе объектно-ориентированного подхода
Владеть: навыками разработки и отладки программ не менее, чем на одном из алгоритмических процедурных языков программирования высокого уровня
ОПК-5: Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем;
Знать: методы построения и оценки эффективности алгоритмов и их программной реализации
Уметь: применять в профессиональной деятельности современные языки программирования для разработки алгоритмических и программных решений
Владеть: Владеть современными средами и средствами разработки программного обеспечения

3 РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Наименование и содержание разделов, тем и учебных занятий	Семестр (курс для ЗАО)	Контактная работа		СР (часы)	Инновац. формы занятий	Форма текущего контроля
		Лек. (часы)	Пр. (часы)			
Раздел 1. Технологии программирования. Аппликативные и императивные языки программирования. Структуры данных и типизация в программировании	2					0

<p>Тема 1. Императивные языки программирования. Структуры, ориентированные на присваивание. Синтаксис и семантика императивных ЯП. Операторы. Переменные и присваивание. Состояние программы. Функциональное значение операторов и их операционная семантика. Технология императивного программирования. Метод утверждений и доказательство терминированности. Объявления типов в программировании. Скалярные элементы. Объявление типов посредством перечисления. Функциональное программирование. Практическое занятие №1 Технологии имитационного моделирования в проектировании ПО. Адаптивные технологии разработки ПО.</p>		2	10	12	ИЛ	
<p>Тема 2. Понятие абстрактного типа данных (АТД). Пользовательские типы. Тип структуры. Элементы структур. Агрегатные типы. Проектирование типов данных соответствующих задаче. Объектно-ориентированное программирование (ООП). Объектно-ориентированный подход. Основные принципы объектно-ориентированного программирования. Объект, сообщение, класс, экземпляр объекта и метод. Абстракция, инкапсуляция, наследование и полиморфизм. Объектно-ориентированное проектирование. Классы. Описания протокола класса. Инструментальная реализация принципов ООП. Объектно-ориентированные языки программирования.</p>		4		14	ИЛ	
<p>Раздел 2. Технологии проектирования программного обеспечения (ПО). Жизненный цикл ПО. Модели проектирования ПО. Руководство программным проектом</p>						О
<p>Тема 3. Конструирование ПО. Организация процесса конструирования ПО. Классический жизненный цикл ПО. Модели проектирования ПО. Макетирование. Стратегии конструирования ПО. Быстрая разработка приложений. Спиральная модель. Компонентно-ориентированная модель. Тяжеловесные и облегченные процессы. XP-процесс. Практическое занятие №2 Современные методы организации разработки ПО и управления программными проектами.</p>		4	12	10	ИЛ	
<p>Тема 4. Модели качества процессов конструирования. Руководство программным проектом. Процесс руководства проектом. Планирование проектных задач. Метрики программного проекта. Измерения, меры и метрики. Размерно-ориентированные метрики. Функционально-ориентированные метрики.</p>		2		16	ИЛ	

Раздел 3. Синтез программных систем. Структурное проектирование ПО. Декомпозиция подсистем на модули. Языки описания программных проектов					
Тема 5. Особенности процесса синтеза программных систем. Особенности этапа проектирования. Структурирование системы. Моделирование управления системой. Декомпозиция подсистем на модули. Свойства модулей. Сложность программной системы. Основы проектирования программных систем. Классические методы проектирования. Языки описания программных проектов. Универсальный язык описания программных проектов (UML). Case – системы. Элементы описания Use Case. Практическое занятие №3 Современные методы синтеза используемые при разработке ПО.	2	12	16	ИЛ	О
Тема 6. Принципы объектно-ориентированного представления программных систем. Абстрагирование, инкапсуляция, модульность, иерархическая организация. Объекты и их общие характеристики. Отношения между объектами. Их виды. Классы. Их свойства и характеристики. Отношения между классами. Ассоциации классов. Унифицированный процесс разработки объектно-ориентированных ПС. Эволюционно-инкрементная организация жизненного цикла разработки. Управление риском. Этапы унифицированного процесса разработки. Примеры объектно-ориентированной разработки программной системы.	3		16	ИЛ	
Итого в семестре (на курсе для ЗАО)	17	34	84		
Консультации и промежуточная аттестация (Экзамен)		2,5	42,5		
Всего контактная работа и СР по дисциплине		53,5	126,5		

4 КУРСОВОЕ ПРОЕКТИРОВАНИЕ

Курсовое проектирование учебным планом не предусмотрено

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

5.1 Описание показателей, критериев и системы оценивания результатов обучения

5.1.1 Показатели оценивания

Код компетенции	Показатели оценивания результатов обучения	Наименование оценочного средства
ОПК-5	1. называет современные технологии, инструментальные среды, программно-технические платформы для решения профессиональных задач; современное программное и аппаратное обеспечение информационных и автоматизированных систем; 2. обосновывает выбор технологий для разработки оригинальных алгоритмов решения профессиональных задач; модернизирует программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач 3. разрабатывает оригинальные программные средства с использованием современных технологий, для решения профессиональных задач;	вопросы для устного опроса практическое задание практическое задание

УК-6	1. называет основные положения концепции непрерывного образования, современные пути и направления профессионального роста и самосовершенствования в сфере ИТ-образования	вопросы для устного опроса
	2. ставит задачи в сфере профессионального и личного развития, определять способы совершенствования в соответствии со своими особенностями	практическое задание
	3. совершенствует собственную познавательную деятельность	практическое задание

5.1.2 Система и критерии оценивания

Шкала оценивания	Критерии оценивания сформированности компетенций	
	Устное собеседование	Письменная работа
5 (отлично)	Полный, исчерпывающий ответ, явно демонстрирующий глубокое понимание предмета и широкую эрудицию в оцениваемой области, умение использовать теоретические знания для решения практических задач.	
4 (хорошо)	Ответ полный и правильный, основанный на проработке всех обязательных источников информации. Подход к материалу ответственный, но допущены в ответах несущественные ошибки, которые устраняются только в результате собеседования	
3 (удовлетворительно)	Ответ воспроизводит в основном только лекционные материалы, без самостоятельной работы с рекомендованной литературой. Демонстрирует понимание предмета в целом при неполных, слабо аргументированных ответах. Присутствуют неточности в ответах, пробелы в знаниях по некоторым темам, существенные ошибки, которые могут быть найдены и частично устранены в результате собеседования	
2 (неудовлетворительно)	Неспособность ответить на вопрос без помощи экзаменатора. Незнание значительной части принципиально важных элементов дисциплины. Многочисленные существенные ошибки.	

5.2 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности

5.2.1 Перечень контрольных вопросов

№ п/п	Формулировки вопросов
Семестр 2	
1	Примитивы синхронизации
2	Алгоритмы синхронизации
3	Атомарные снимки регистров
4	Ошибки параллельного программирования
5	Пулы потоков
6	Шаблоны программирования
7	Множество цепочек. Операции над цепочками.
8	Языки. Операции над языками. Итерация языка.
9	Гомоморфизм. Алгоритмы. Частичные алгоритмы. Полные алгоритмы. Рекурсивные алгоритмы. Задание алгоритмов.
10	Задание языков программирования. Синтаксис и семантика. Процесс компиляции. Лексический анализ.

5.2.2 Типовые тестовые задания

не предусмотрено

5.2.3 Типовые практико-ориентированные задания (задачи, кейсы)

1. Реализовать классический паттерн producer-consumer с небольшими дополнительными условиями. Программа должна состоять из четырех потоков: один - главный, producer, consumer и interruptor. На стандартный ввод программы подается строка - список чисел произвольной длины. Задача producer-потока - получить на вход список чисел, и по очереди использовать каждое значение из этого списка для обновления переменной разделяемой между потоками. После этого поток должен дождаться реакции consumer-потока, и продолжить обновление значений только после того как поток-consumer принял это изменение. Задача consumer-потока отреагировать на каждое изменение переменной data и набирать сумму полученных значений. После того как достигнуто последнее обновление, функция потока должна вернуть результирующую сумму. Также этот поток должен защититься от попыток потока-interruptor его остановить. Задача потока-interruptor проста: пока происходит процесс обновления значений, он должен постоянно пытаться остановить поток consumer. Как только поток producer произвел последнее обновление, этот поток завершается. Функция run_threads должна запускать все три потока, дожидаться их выполнения, и возвращать результат потока-consumer. Для обеспечения межпоточного взаимодействия допускается использование только pthread API

2. Реализовать интерфейс множества в виде lock-free структуры данных. Задание должно состоять из одного файла LockFreeSetImpl.java содержащего класс, реализующий интерфейс LockFreeSet, приложенный к задаче. Спецификация реализации описана в документации к интерфейсу.

3. Реализовать с использованием графовой модели вычислений следующую задачу, разбив её на чаты и обеспечив максимальную потенциальную утилизацию ресурсов (изображение == матрица MxN):

1. На вход подаются генерированные случайным образом изображения

2. Различные алгоритмы ищут || на изображении:

- Набор максимальных элементов (максимальная яркость)
- Набор минимальных элементов (минимальная яркость)
- Набор элементов, равных заданному в командной строке значению (0-255)

3. По результатам предыдущего шага на изображении выделяются все найденные элементы (например, квадрат с максимальной яркостью вокруг точки)

4. Результат предыдущего шага передается на:

- Расчёт обратного изображения (инверсия яркости)
- Расчёт средней яркости изображения

5. Результат расчёта средней яркости выводится в файл в произвольном формате, если указан соответствующий флаг запуска приложения

6. Результат расчёта обратного изображения никуда не идёт

Предполагается использовать:

- broadcast_node
- join_node
- function_node
- limiter_node

5.3 Методические материалы, определяющие процедуры оценивания знаний, умений, владений (навыков и (или) практического опыта деятельности)

5.3.1 Условия допуска обучающегося к промежуточной аттестации и порядок ликвидации академической задолженности

Проведение промежуточной аттестации регламентировано локальным нормативным актом СПбГУПТД «Положение о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся»

5.3.2 Форма проведения промежуточной аттестации по дисциплине

Устная Письменная Компьютерное тестирование Иная

5.3.3 Особенности проведения промежуточной аттестации по дисциплине

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1 Учебная литература

Автор	Заглавие	Издательство	Год издания	Ссылка
6.1.1 Основная учебная литература				
Мейер, Б.	Объектно-ориентированное программирование и программная инженерия	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа	2019	http://www.iprbookshop.ru/79706.html
Алфёров, В. В., Миронов, Ю. М.	Вычислительная техника и сети в отрасли	Москва: Московская государственная академия водного транспорта	2018	http://www.iprbookshop.ru/67596.html

Козырева, Г. Ф.	Функциональное логическое программирование	и	Саратов: Ай Пи Эр Медиа	2018	http://www.iprbookshop.ru/71596.html
6.1.2 Дополнительная учебная литература					
Иванилова, С. В.	Управление инновационными проектами		Москва: Дашков и К, Ай Пи Эр Медиа	2018	http://www.iprbookshop.ru/66843.html
Тюльпинова, Н. В.	Алгоритмизация программирование	и	Саратов: Вузовское образование	2019	http://www.iprbookshop.ru/80539.html
Дорофеев, А. С., Дорофеев, Р. С., Рогачева, С. А., Сосинская, С. С.	Разработка баз данных		Саратов: Ай Пи Эр Медиа	2018	http://www.iprbookshop.ru/70276.html

6.2 Перечень профессиональных баз данных и информационно-справочных систем

Портал Росстандарта по стандартизации [Электронный ресурс]. URL: <http://standard.gost.ru/wps/portal/>

6.3 Перечень лицензионного и свободно распространяемого программного обеспечения

MicrosoftOfficeProfessional
Microsoft Windows
JetBrains Toolbox
GNU Prolog
Haskell
IntelliJ IDEA
Microsoft Visual C++ 2010 Express
Microsoft Visual Studio Code
Microsoft Visual Studio Community
Notepad++
Python

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

Аудитория	Оснащение
Компьютерный класс	Мультимедийное оборудование, компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду
Лекционная аудитория	Мультимедийное оборудование, специализированная мебель, доска
Учебная аудитория	Специализированная мебель, доска