

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
федерального государственного бюджетного образовательного учреждения высшего образования
**«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(МИ ВлГУ)**

Отделение среднего профессионального образования

«УТВЕРЖДАЮ»
Заместитель директора по УР
_____ Д.Е. Андрианов
« 16 » 06 2020 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Сравнительный анализ языков программирования

для специальности 09.02.03 Программирование в компьютерных системах

Муром, 2020 г.

Рабочая программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта (далее - ФГОС) по специальности среднего профессионального образования (далее - СПО) 09.02.03 Программирование в компьютерных системах №804 от 28 июля 2014 года.

Кафедра-разработчик: информационных систем.

Рабочую программу составил: Борданов И.А.

(подпись)

от «02» июня 2020 г.

Рабочая программа рассмотрена и одобрена на заседании кафедры ИС.

Протокол № 18

от «02» июня 2020 г.

Заведующий кафедрой ИС *Андреианов Д.Е.*

(подпись)

СОДЕРЖАНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	6
3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ	9
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	11

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

Сравнительный анализ языков программирования

1.1. Область применения рабочей программы

Рабочая программа учебной дисциплины является частью программы подготовки специалистов среднего звена в соответствии с ФГОС по специальности 09.02.03 Программирование в компьютерных системах.

Рабочая программа учебной дисциплины может быть использована в дополнительном профессиональном образовании, для получения дополнительных компетенций, умений и знаний, необходимых для обеспечения конкурентоспособности выпускника на рынке труда и продолжения образования по специальности.

1.2. Место учебной дисциплины в структуре программы подготовки специалистов среднего звена:

Дисциплина ОП.14 Сравнительный анализ языков программирования является общепрофессиональной дисциплиной

Рабочая программа профессионального модуля – является частью рабочей программы основной профессиональной образовательной программы в соответствии с ФГОС по специальности (специальностям) СПО 09.02.03 Программирование в компьютерных системах в части освоения основного вида профессиональной деятельности (ВПД): Оператор электронно-вычислительных и вычислительных машин.

1.3. Цели и задачи учебной дисциплины - требования к результатам освоения учебной дисциплины:

Целью освоения дисциплины (модуля) "Сравнительный анализ языков программирования" является обучение студентов методологии сравнения и оценки различных языков программирования.

Задачи дисциплины:

- знакомство с историей развития языков программирования и основными тенденциями их развития;
- изучение методов сравнения и оценки языков программирования;
- знакомство с синтаксисом и семантикой языков программирования Си, Паскаль, Java и другими;
- получение практических навыков решения задач с применением различных языковых конструкций

В результате освоения учебной дисциплины обучающийся должен уметь:

- пользоваться своими знаниями для решения фундаментальных, прикладных и технологических задач в различных предметных областях (ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ОК-6, ОК-7, ОК-8, ОК-9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6);
- делать правильные выводы из сопоставления результатов теории и эксперимента, выбирать правильно параметры методов, адекватные размерности обучающих выборок (ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ОК-6, ОК-7, ОК-8, ОК-9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6);
- делать качественные и количественные выводы при переходе к предельным условиям в изучаемых проблемах (ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ОК-6, ОК-7, ОК-8, ОК-9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6);
- осваивать новые предметные области, теоретические подходы и экспериментальные методики (ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ОК-6, ОК-7, ОК-8, ОК-9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6);
- планировать оптимальное проведение обучения по прецедентам (ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ОК-6, ОК-7, ОК-8, ОК-9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6);

- эффективно использовать информационные технологии и компьютерную технику для достижения необходимых теоретических и прикладных результатов (ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ОК-6, ОК-7, ОК-8, ОК-9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6).

В результате освоения учебной дисциплины обучающийся должен знать:

- фундаментальные понятия и методы теории алгоритмов (ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4);
- современные проблемы анализа алгоритмов, постановки задачи, выбора языка программирования, построения полного решения задачи (ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4);
- методы и подходы решения практических алгоритмических задач, от постановки задачи до финального алгоритма (ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4);
- инструментальные языковые средства решения задач построения, распознавания и порождения (ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4).

В результате освоения дисциплины обучающийся должен владеть следующими общими (ОК) и профессиональными (ПК) компетенциями:

- ОК-1 Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес;
- ОК-2 Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество;
- ОК-3 Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность;
- ОК-4 Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития;
- ОК-5 Использовать информационно-коммуникационные технологии в профессиональной деятельности;
- ОК-6 Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями;
- ОК-7 Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий;
- ОК-8 Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации;
- ОК-9 Ориентироваться в условиях частой смены технологий в профессиональной деятельности;
- ПК 1.1 Выполнять разработку спецификаций отдельных компонент;
- ПК 1.2 Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля;
- ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств;
- ПК 1.4 Выполнять тестирование программных модулей;
- ПК 1.5 Осуществлять оптимизацию программного кода модуля;
- ПК 1.6 Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций;

1.4. Количество часов на освоение программы учебной дисциплины:

Максимальной учебной нагрузки обучающегося 93 часа, в том числе:
обязательной аудиторной нагрузки обучающегося 64 часа;
самостоятельной нагрузки обучающегося 29 часов.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
	7 семестр
Максимальная учебная нагрузка (всего)	93
Обязательная аудиторная учебная нагрузка (всего)	64
В том числе:	
лекционные занятия	20
практические занятия	16
лабораторные работы	28
контрольные работы	
курсовая работа	
Самостоятельная работа обучающегося (всего)	29
Итоговая аттестация в форме	Экзамен

2.2. Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающихся	Объем часов	Уровень освоения
1	2	3	4
	7 семестр		
Раздел 1	Общие сведения о языках программирования		
Тема 1.1 История развития языков программирования	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> История развития языков программирования.	2	1
	<i>Практические занятия.</i> Использование систем управления версиями.	2	2
	<i>Лабораторные работы.</i> Оформление текстовых документов в соответствии с ГОСТ.	4	3
	<i>Самостоятельная работа обучающихся.</i> Типы перечисления. Организация подпрограмм в различных языках программирования. Параметры функций и их передача. Полиморфизм языков программирования. Исключения в языке PL/1. Исключения в Ada. Обработка ошибок в языке Eiffel. Параллельное программирование. Что такое параллелизм. Взаимные исключения. Языки параллельного программирования Occam, Linda. Декомпозиция программ. Раздельная компиляция. Пакеты, модули.	14	3
Раздел 2	Процедурные языки программирования		
Тема 2.1 Процедурные	<i>Содержание учебного материала</i>		

языки программирования. Особенности языка программирования C++	<i>Лекционные занятия.</i> Процедурные языки программирования. Особенности языка программирования C++.	2	1
	<i>Лабораторные работы.</i> Разработка объектно-ориентированной программы на C++. Разработка консольных java-приложений.	8	3
	<i>Самостоятельная работа обучающихся.</i> Массивы и контроль соответствия типов. Операторы goto. Стековая архитектура. Представление вещественных чисел. Динамическая диспетчеризация. Динамический полиморфизм в языке C++.	7	3
Тема 2.2 Объектно-ориентированное программирование в C++	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Объектно-ориентированное программирование в C++.	2	1
	<i>Самостоятельная работа обучающихся.</i> Распределение памяти, динамическое распределение памяти.	1	3
Тема 2.3 Особенности языка программирования Python	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Особенности языка программирования Python.	2	1
Тема 2.4 Язык программирования GO	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Java. Особенности ООП в GO.	2	1
	<i>Лабораторные работы.</i> Обработка строк. Использование регулярных выражений в java-приложениях. Коллекции в java. Графический интерфейс Python. Разработка сетевых приложений на Python.	16	3
	<i>Самостоятельная работа обучающихся.</i> Работа с базами данных в Python. Структура байт-кода GO.	3	3
Раздел 3	Языки функционального программирования		
Тема 3.1 Обзор функциональных языков программирования	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Обзор функциональных языков программирования.	2	1
Тема 3.2 Язык программирования GO	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Язык программирования GO.	2	1
	<i>Практические занятия.</i> Язык программирования GO. Распределенные приложения в GO.	4	2
Тема 3.3 Язык программирования Python	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Язык программирования Haskell. Компилятор GHCi. Базовые типы. Символы. Списки. Выражения. Функции Haskell: полиморфизм и перегрузка. Числовые функции. Функции высшего порядка.	4	1
	<i>Практические занятия.</i> Основы работы с GO.	6	2

	GO: Рекурсия и работа со списками. GO: Функции высшего порядка.		
Раздел 4	Логическое программирование		
Тема 4.1 Логическое программирование. Пролог	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия. Логическое программирование. Пролог.</i>	2	1
	<i>Практические занятия. Программирование структур данных на языке Пролог. Списки. Визуальное программирование на языке Пролог.</i>	4	2
	<i>Самостоятельная работа обучающихся. Логическое программирование. Чистое логическое программирование. Унификация. Язык Prolog.</i>	4	3
Всего:		93	

Для характеристики уровня освоения учебного материала используются следующие обозначения:

1. - ознакомительный (узнавание новых объектов, свойств);
2. - репродуктивный (выполнение деятельности по образцу, инструкции или под руководством);
3. - продуктивный (планирование и самостоятельное выполнение деятельности, решение проблемных задач).

3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально – техническому обеспечению

Лаборатория ГИС и САПР

Сервер; 12 персональных компьютеров; проектор Sanyo PDG-DSU20; экран настенный Drapper Apex Star

Программное обеспечение:

Microsoft Visual Studio (Программа Microsoft Azure Dev Tools for Teaching (Order Number: IM126433))

Python 3 (PSF License Agreement)

NumPy (Модифицированная лицензия BSD)

SciPy (BSD)

Matplotlib (matplotlib licence)

Pandas (BSD)

3.2. Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, интернет – ресурсов, дополнительной литературы.

Основные источники:

1. Кауфман В.Ш. Языки программирования. Концепции и принципы [Электронный ресурс]/ Кауфман В.Ш.— Электрон. текстовые данные.— Саратов: Профобразование, 2019.— 464 с.. <http://www.iprbookshop.ru/88014.html>
2. Липпман С. Язык программирования C++. Полное руководство [Электронный ресурс]/ Липпман С., Лажоие Ж.— Электрон. текстовые данные.— Саратов: Профобразование, 2019.— 1104 с.. <http://www.iprbookshop.ru/89862.html>
3. Сузи Р.А. Язык программирования Python [Электронный ресурс]: учебное пособие/ Сузи Р.А.— Электрон. текстовые данные.— Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020.— 350 с.. <http://www.iprbookshop.ru/97589.html>

Дополнительные источники:

1. Фарафонов, А. С. Программирование на языке высокого уровня : методические указания к проведению лабораторных работ по курсу «Программирование» / А. С. Фарафонов. — Липецк : Липецкий государственный технический университет, ЭБС АСВ, 2013. — 32 с. . <https://www.iprbookshop.ru/22912.html>
2. Васюткина И.А. Технология разработки объектно-ориентированных программ на JAVA [Электронный ресурс]: учебно-методическое пособие/ Васюткина И.А.— Электрон. текстовые данные.— Новосибирск: Новосибирский государственный технический университет, 2012.— 152 с.. <http://www.iprbookshop.ru/45047.html>
3. Теория и реализация языков программирования : учебное пособие / В. А. Серебряков, М. П. Галочкин, Д. Р. Гончар, М. Г. Фуругян. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 372 с.. <https://www.iprbookshop.ru/102068.html>
4. Кауфман В.Ш. Языки программирования. Концепции и принципы — М.: ДМК Пресс, 2013. - 464 с.. <http://www.iprbookshop.ru/6932.html>
5. Молдованова О.В. Языки программирования и методы трансляции [Электронный ресурс]: учебное пособие/ Молдованова О.В.— Электрон. текстовые данные.— Новосибирск: Сибирский государственный университет телекоммуникаций и информатики, 2012.— 134 с.. <http://www.iprbookshop.ru/54809.html>

Интернет-ресурсы:

1. электронно-библиотечная система "IPRBooks"
(<http://www.iprbookshop.ru>)

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
пользоваться своими знаниями для решения фундаментальных, прикладных и технологических задач в различных предметных областях	Тестирование. Защита лабораторных занятий. Контрольные работы
делать правильные выводы из сопоставления результатов теории и эксперимента, выбирать правильно параметры методов, адекватные размерности обучающих выборок	Тестирование. Защита лабораторных занятий. Контрольные работы
делать качественные и количественные выводы при переходе к предельным условиям в изучаемых проблемах	Тестирование. Защита лабораторных занятий. Контрольные работы
осваивать новые предметные области, теоретические подходы и экспериментальные методики	Тестирование. Защита лабораторных занятий. Контрольные работы
планировать оптимальное проведение обучения по прецедентам	Тестирование. Защита лабораторных занятий. Контрольные работы
эффективно использовать информационные технологии и компьютерную технику для достижения необходимых теоретических и прикладных результатов	Тестирование. Защита лабораторных занятий. Контрольные работы
фундаментальные понятия и методы теории алгоритмов	Тестирование. Защита лабораторных занятий. Контрольные работы
современные проблемы анализа алгоритмов, постановки задачи, выбора языка программирования, построения полного решения задачи;	Тестирование. Защита лабораторных занятий. Контрольные работы
методы и подходы решения практических алгоритмических задач, от постановки задачи до финального алгоритма	Тестирование. Защита лабораторных занятий. Контрольные работы
инструментальные языковые средства решения задач построения, распознавания и порождения;	Тестирование. Защита лабораторных занятий. Контрольные работы

**Фонд оценочных материалов (средств) по дисциплине
Сравнительный анализ языков программирования**

**1. Оценочные материалы для проведения текущего контроля успеваемости
по дисциплине**

ОК 1, ОК 2, ОК 3, ОК 4, ОК 5, ОК 6, ОК 7, ОК 8, ОК 9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6

Блок "Знать"

Понятие языка программирования

Процедурные языки программирования

Языки, ориентированные на данные

Объектно-ориентированные языки

Директивы препроцессора в C++. include, define.

Директивы препроцессора в C++. Условная компиляция.

Исключения в C++. Генерация и перехват.

Исключения в C++. Генерация внутри функции. Повторная генерация.

Исключения в C++. Стандартные и собственные классы исключений.

Объектно-ориентированное программирование в C++. Основные принципы ООП.

Классы.

Объектно-ориентированное программирование в C++. Объекты. Конструкторы.

Деструкторы.

Объектно-ориентированное программирование в C++. Структуры. Объединения.

Непроцедурные языки.

Особенности языка Erlang.

Свойства Java.

Базовые типы в Java.

Основные отличия Java от C++.

Объектная модель Java.

Переносимость Java на различные архитектуры.

Безопасность в Java.

Блок "Уметь"

ОК 1, ОК 2, ОК 3, ОК 4, ОК 5, ОК 6, ОК 7, ОК 8, ОК 9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6

Построить систему классов для описания плоских геометрических фигур: круг, квадрат, прямоугольник. Предусмотреть методы для создания объектов, перемещения на плоскости.

Построить описание класса, содержащего информацию о почтовом адресе организации. Предусмотреть возможность раздельного изменения составных частей адреса.

Составить описание класса для представления комплексных чисел. Обеспечить выполнение операций сложения, вычитания и умножения комплексных чисел.

Составить описание класса для объектов-векторов, задаваемых координатами концов в трехмерном пространстве. Обеспечить операции сложения и вычитания векторов с получением нового вектора (суммы или разности), вычисления длины вектора.

Составить описание класса прямоугольников со сторонами, параллельными осям координат. Предусмотреть возможность перемещения прямоугольников на плоскости, изменения размеров.

Составить описание класса для определения одномерных массивов целых чисел (векторов). Предусмотреть возможность обращения к отдельному элементу массива с контролем выхода за пределы массива, возможность задания произвольных границ индексов при создании объекта, вывода на экран элемента массива по заданному индексу и всего массива

Составить описание класса, обеспечивающего представление матрицы произвольного размера с возможностью изменения числа строк и столбцов, вывода на экран подматрицы любого размера и всей матрицы.

Описать класс «Домашняя библиотека», частью которого является класс «Книга». Предусмотреть возможность работы с произвольным числом книг, добавления книг в библиотеку, удаления книг из нее, вывода информации о книгах на экран. Использовать конструктор и методы класса.

Описать класс «Дата» – день, месяц (число), год. Предусмотреть возможность задания месяца в виде строки, возможность вывода даты на экран, а также возможности прибавления к дате указанного количества дней и вычитания их из нее.

Описать класс «Студенческая группа», частью которого является класс «Студент». Предусмотреть возможность работы с переменным числом студентов, поиска студента по фамилии, добавления и удаления записей, вывода информации о студентах на экран.

Описать класс «Строка», позволяющий выполнять основные операции: задание строки произвольной длины, поиск в строке подстроки, объединение двух строк, возвращение длины строки, вывод строки на экран.

Описать класс «Экзаменационная ведомость», частью которого является класс «Студент», хранящий информацию о студенте и его оценке на экзамене. Предусмотреть возможность задания произвольного количества студентов, сортировки студентов по фамилии, вычисления среднего балла за экзамен, а также вывода на экран количества различных оценок (неудовлетворительно, удовлетворительно, хорошо, отлично).

Описать класс «Файл», содержащий сведения о произвольном файле на диске. Реализовать методы, позволяющие получать отдельные компоненты полного пути файла: каталог файла, имя файла, расширение файла. Реализовать методы получения размера файла, удаления файла.

Описать класс «Список строк», позволяющий хранить произвольное количество строк, связанных в список. Предусмотреть методы добавления и удаления строк из списка, поиска нужной строки по подстроке, вывода списка строк на экран. Использовать конструктор класса.

Описать класс «Автобусный рейс», позволяющий хранить сведения о начальной и конечной станциях, а также о свободных и занятых местах (включая их номера). Предусмотреть методы покупки и возврата билета на определенное место, получения количества свободных мест. Использовать конструктор класса.

Блок "Владеть"

ОК 1, ОК 2, ОК 3, ОК 4, ОК 5, ОК 6, ОК 7, ОК 8, ОК 9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6

Разработать на C++ собственные классы исключений с описанием ошибки в виде строки и массива строк.

Разработать на C++ межплатформенную программу работы с файлом (открытие, чтение, запись, поиск в файле, закрытие).

Разработать на C++ межплатформенный класс взаимного исключения.

Разработать на C++ межплатформенный класс для работы с потоками (запуск, остановка, ожидание завершения).

Разработать на Java программу, подключающуюся к БД и отображающую содержание какой-либо таблицы

Общее распределение баллов текущего контроля по видам учебных работ для студентов

Рейтинг-контроль 1	Лабораторные и практические работы	15
Рейтинг-контроль 2	Лабораторные и практические работы	15
Рейтинг-контроль 3	Лабораторные и практические работы	15

Посещение занятий студентом		5
Дополнительные баллы (бонусы)		5
Выполнение семестрового плана самостоятельной работы		5

2. Промежуточная аттестация по дисциплине

Перечень вопросов к экзамену / зачету / зачету с оценкой.

Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)

ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ОК-6, ОК-7, ОК-8, ОК 9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6

Темы для устного опроса

Понятие языка программирования

Процедурные языки программирования

Языки, ориентированные на данные

Объектно-ориентированные языки

Директивы препроцессора в C++. include, define.

Директивы препроцессора в C++. Условная компиляция.

Исключения в C++. Генерация и перехват.

Исключения в C++. Генерация внутри функции. Повторная генерация.

Исключения в C++. Стандартные и собственные классы исключений.

Объектно-ориентированное программирование в C++. Основные принципы ООП.

Классы.

Объектно-ориентированное программирование в C++. Объекты. Конструкторы.

Деструкторы.

Объектно-ориентированное программирование в C++. Структуры. Объединения.

Непроцедурные языки.

Особенности языка Erlang.

Свойства Java.

Базовые типы в Java.

Основные отличия Java от C++.

Объектная модель Java.

Переносимость Java на различные архитектуры.

Безопасность в Java.

Задания для выполнения практических работ

ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ОК-6, ОК-7, ОК-8, ОК 9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6

Разработать на C++ собственные классы исключений с описанием ошибки в виде строки и массива строк.

Разработать на C++ межплатформенную программу работы с файлом (открытие, чтение, запись, поиск в файле, закрытие).

Разработать на C++ межплатформенный класс взаимного исключения.

Разработать на C++ межплатформенный класс для работы с потоками (запуск, остановка, ожидание завершения).

Разработать на Java программу, подключающуюся к БД и отображающую содержание какой-либо таблицы

Методические материалы, характеризующих процедуры оценивания

ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ОК-6, ОК-7, ОК-8, ОК 9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6

Темы для устного опроса

Понятие языка программирования

Процедурные языки программирования

Языки, ориентированные на данные

Объектно-ориентированные языки

Директивы препроцессора в C++. include, define.

Директивы препроцессора в C++. Условная компиляция.

Исключения в C++. Генерация и перехват.

Исключения в C++. Генерация внутри функции. Повторная генерация.

Исключения в C++. Стандартные и собственные классы исключений.

Объектно-ориентированное программирование в C++. Основные принципы ООП.

Классы.

Объектно-ориентированное программирование в C++. Объекты. Конструкторы.

Деструкторы.

Объектно-ориентированное программирование в C++. Структуры. Объединения.

Непроцедурные языки.

Особенности языка Erlang.

Свойства Java.

Базовые типы в Java.

Основные отличия Java от C++.

Объектная модель Java.

Переносимость Java на различные архитектуры.

Безопасность в Java.

Задания для выполнения практических работ

ОК-1, ОК-2, ОК-3, ОК-4, ОК-5, ОК-6, ОК-7, ОК-8, ОК 9, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6

Разработать на C++ собственные классы исключений с описанием ошибки в виде строки и массива строк.

Разработать на C++ межплатформенную программу работы с файлом (открытие, чтение, запись, поиск в файле, закрытие).

Разработать на C++ межплатформенный класс взаимного исключения.

Разработать на C++ межплатформенный класс для работы с потоками (запуск, остановка, ожидание завершения).

Разработать на Java программу, подключающуюся к БД и отображающую содержание какой-либо таблицы

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	Уровень сформированности компетенций
Более 80	«Отлично»	Содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их	Высокий уровень

		выполнения оценено числом баллов, близким к максимальному	
66-80	«Хорошо»	Содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	<i>Продвинутый уровень</i>
50-65	«Удовлетворительно»	Содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки	<i>Пороговый уровень</i>
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	<i>Компетенции не сформированы</i>

3. Задания в тестовой форме по дисциплине

Примеры заданий:

Вопрос: Какой из перечисленных языков программирования является интерпретируемым?

C++

C#

[*] Python

Вопрос: Как объявить переменную в Python?

int x;

var x;

[*] x = 10;

Вопрос: Как объявить функцию в C++?

[*] void myFunction() {}

function myFunction() {}

def myFunction() {}

Вопрос: Как объявить массив целых чисел в C#?


```
array<int> myArray;  
[*] int[] myArray;  
List<int> myArray;
```

Вопрос: Какой оператор используется для проверки условия в Python?

```
for  
[*] if  
check
```

Вопрос: Какой тип данных используется для хранения десятичных чисел с плавающей точкой в Python?

```
float32  
[*] float  
double
```

Вопрос: Какой оператор используется для выделения памяти под объект в C++?

```
alloc  
[*] new  
malloc
```

Вопрос: Какой символ используется для обозначения комментария в Python?

```
//  
[*] #  
/* */
```

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке <https://www.mivlgu.ru/iop/mod/quiz/view.php?id=53098>

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.