МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)

УТВЕРЖДАЮ
Первый по учебной работе
_____ **П. Е. Троян**
«12» 04 2016 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

ЯЗЫКИ ПРОГРАММИРОВАНИЯ

Уровень основной образовательной программы: бакалавриат
Направление(я) подготовки (специальность): 11.03.03 Конструирование и технология электронных средств
Профиль(и): Проектирование и технология электронно-вычислительных средств
Форма обучения: очная
Факультет Безопасности
Кафедра Комплексной информационной безопасности электронно-вычислительных систем (КИБЭВС)
Курс 2 **Семестр 3**

Учебный план набора 2013 года и последующих лет.

Распределение рабочего времени:

№	Виды учебной работы	Семестр 3	Всего	Единицы
1.	Лекции	36	36	часов
2.	Лабораторные работы	Не предусмотрено		часов
3.	Практические занятия	36	36	часов
4.	Курсовой проект/работа (КРС) (аудиторная)	Не предусмотрено		часов
5.	Всего аудиторных занятий (Сумма 1-4)	72	72	часов
6.	Из них в интерактивной форме	16	16	часов
7.	Самостоятельная работа студентов (СРС)	36	36	часов
8.	Всего (без экзамена) (Сумма 5,7)	108	108	часов
9.	Самост. работа на подготовку, сдачу экзамена	Не предусмотрено		
10.	Общая трудоемкость (Сумма 8,9)	108	108	часов
	(в зачетных единицах)	3	3	ЗЕТ

Зачет 3 семестр

Томск 2016

Согласована на портале № 5554

Рассмотрена и одобрена на заседании кафедры
протокол № 4 от «12» 4 2016 г.

Лист согласований

Рабочая программа составлена с учетом требований Федерального Государственного образовательного стандарта высшего профессионального образования (ФГОС ВПО) третьего поколения по направлению подготовки (специальности) 11.03.03 Конструирование и технология электронных средств "Проектирование и технология электронно-вычислительных средств", утвержденного приказом №1333 от 12.11.2015, рассмотрена и утверждена на заседании кафедры «12» 04 2016 г., протокол № 4.

Разработчики доцент каф. КИБЭВС _____ /А.С. Романов/

Зав. кафедрой КИБЭВС, профессор _____ /А.А. Шелупанов/

Рабочая программа согласована с факультетом, профилирующей и выпускающей кафедрами направления подготовки (специальности).

Декан Факультета Безопасности _____ /Е.М. Давыдова/

Зав. профилирующей кафедрой КИБЭВС _____ /А.А. Шелупанов/

Зав. выпускающей кафедрой КИБЭВС _____ /А.А. Шелупанов/

Эксперты:

Директор Центра системного проектирования _____ /А.А. Конев/

Ст. преподаватель каф. КИБЭВС _____ /М.А. Сопов/

Согласована на портале № 5554

1. Цели и задачи дисциплины:

Цель дисциплины «Языки программирования» - изучение основных принципов построения языков программирования и программирования на языках высокого уровня.

2. Место дисциплины в структуре ООП:

Дисциплина «Языки программирования» относится к дисциплинам по выбору профессионального цикла. Предшествующие дисциплины: Основы программирования, Математическая логика и теория алгоритмов. Последующие дисциплины: Базы данных, Системное программирование.

3. Требования к результатам освоения дисциплины:

Процесс изучения дисциплины направлен на формирование следующих компетенций:

– способность осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий (ОПК-6).

В результате изучения дисциплины студент должен

Знать:

– общие принципы построения и использования современных языков программирования высокого уровня;

– язык программирования высокого уровня (объектно-ориентированное программирование);

– возможности, классификацию и область применения макрообработки;

– обработки исключительных ситуаций.

Уметь:

– работать с интегрированной средой разработки программного обеспечения;

– использовать шаблоны классов и средства макрообработки;

– использовать динамически подключаемые библиотеки.

Владеть:

– навыками проектирования программного обеспечения с использованием средств автоматизации.

4. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 3 (три) зачетные единицы.

Вид учебной работы	Всего часов	Семестры
		3
Аудиторные занятия (всего)	72	72
В том числе:	-	-
Лекции	36	36
Лабораторные работы (ЛР)	Не предусмотрено	
Практические занятия (ПЗ)	36	36
Семинары (С)	Не предусмотрено	
Коллоквиумы (К)	Не предусмотрено	
Курсовой проект/(работа) (аудиторная нагрузка)	Не предусмотрено	
Другие виды аудиторной работы	Не предусмотрено	
Самостоятельная работа (всего)	36	36
В том числе:		
Курсовой проект (работа) (самостоятельная работа)	Не предусмотрено	
Расчетно-графические работы	Не предусмотрено	
Реферат	Не предусмотрено	
Другие виды самостоятельной работы		
Проработка лекционного материала	18	18
Подготовка к практическим занятиям	18	18

Согласована на портале № 5554

Вид аттестации - зачет	-	-
Общая трудоемкость час	108	108
Зачетные Единицы Трудоемкости	3	3

5. Содержание дисциплины

Содержание курса включает основные сведения о характеристиках и свойствах языков программирования высокого уровня, принципы построения языков и технологии программирования, средства описания данных; средства описания действий; абстрактные типы данных: инкапсуляция, спецификация, реализация, параметризация, классы и объекты; обработка файлов; обработка исключительных ситуаций; параллельная обработка; макропроцессоры и макрогенераторы; современные интегрированные среды разработки программ; графический интерфейс пользователя; отладчики; генераторы кода/приложений; библиотеки программ и классов; стандарты языков программирования, общая характеристика языков ассемблера: назначение, принципы построения и использования; структура языка, основные группы команд, операторы, средства взаимодействия с операционной системой.

5.1. Разделы дисциплин и виды занятий

№ п/п	Наименование раздела дисциплины	Лекции	Лаборат. занятия	Практич. занятия.	Курсовой П/Р (КРС)	Самост. работа студента	Всего час. (без экзам)	Формируемые компетенции (ОК, ПК, ПСК)
1.	Классификация языков программирования	2	-	8	-	4	14	ОПК-6
2.	Средства описания данных и действий	4	-	8	-	4	16	ОПК-6
3.	Основные концепции объектно-ориентированного программирования. Абстрактные типы данных.	6	-	8	-	2	16	ОПК-6
4.	Обработка файлов	2	-	-	-	1	3	ОПК-6
5.	Обработка исключительных ситуаций	2	-	4	-	4	10	ОПК-6
6.	Параллельное программирование	4	-	-	-	2	6	ОПК-6
7.	Макропроцессоры и макрогенераторы	2	-	-	-	1	3	ОПК-6
8.	Современные интегрированные среды разработки программ; графический интерфейс пользователя	2	-	-	-	6	8	ОПК-6
9.	Отладчики; генераторы кода/приложений	2	-	-	-	1	3	ОПК-6
10.	Библиотеки программ и классов	2	-	-	-	1	3	ОПК-6
11.	Язык Ассемблер	2	-	-	-	1	3	ОПК-6
12.	Основы логического программирования	2	-	4	-	4	10	ОПК-6
13.	Основы функционального программирования	2	-	4	-	4	10	ОПК-6
14.	Стандарты языков программирования и документирование программ	2	-		-	1	3	ОПК-6
	Итого	**36**	**-**	**36**	**-**	**36**	**108**	

5.2. Содержание разделов дисциплины (по лекциям)

№ п/п	Наименование разделов	Содержание разделов	Трудоемкость (час.)	Формируемые компетенции (ОК, ПК, ПСК)
1.	Классификация языков программирования	Введение. Понятие языка программирования, программа, абстракция. Общие принципы построения и использования языков программирования; характеристики и свойства языков программирования.	2	ОПК-6

Согласована на портале № 5554

		Пути развития и классификация языков программирования. Основные понятия языков программирования. Синтаксис и семантика языков программирования.		
2.	Средства описания данных и действий	Средства описания данных. Типизация языка. Простые типы данных. Структурные типы данных. Структурные типы данных. Динамические структуры данных. Средства описания действий. Определение семантики средств описания действий. Элементы языков программирования. Выражения и операторы действия. Операторы управления. Блоки. Программы. Модули.	4	ОПК-6
3.	Основные концепции объектно-ориентированного программирования. Абстрактные типы данных.	Основные концепции объектно-ориентированного программирования. Абстрактные типы данных. Абстракции низкого и высокого уровней. Классы и объекты. Наследование, формы наследования, преимущества наследования. Инкапсуляция. Полиморфизм и его разновидности. Виртуальные функции. Отношения между классами, диаграммы классов. Преимущества и недостатки объектной модели.	6	ОПК-6
4.	Обработка файлов	Файлы. Файлы прямого и последовательного доступа. Типы доступа. Обработка файлов. Открытие и закрытие файла. Запись в файл, чтение из файла.	2	ОПК-6
5.	Обработка исключительных ситуаций	Обработка исключительных ситуаций. Простейшие способы обработки исключений. Обработка исключений при помощи меток. Структурный механизм обработки исключений. Восстановление вычислительного процесса. Определение семантики средств обработки исключений.		ОПК-6
6.	Параллельное программирование	Параллельная обработка. Параллельное и распределенное программирование. Модели параллельного программирования. Процессы. Взаимное исключение. Синхронизация процессов. Передача сообщений. Сигналы, семафоры, рандеву.	4	ОПК-6
7.	Макропроцессоры и макрогенераторы	Макропроцессоры и макрогенераторы. Принципы работы, способы реализации. Построение анализатора.	2	ОПК-6
8.	Современные интегрированные среды разработки программ. Графический интерфейс пользователя	Современные интегрированные среды разработки программ. Разработка графического интерфейс пользователя. Основные характеристики интегрированных средств разработки.	2	ОПК-6
9.	Отладчики; генераторы кода/приложений	Отладчики. Генераторы кода/приложений. Режимы отладки, использование контрольных точек. Отладка программ без использования средств отладки. Основное назначение генераторов кода и приложений	2	ОПК-6
10.	Библиотеки программ и классов	Библиотеки программ и классов. Принципы создания, правила ведения. Возможности пополнения и использования при собственной разработке.	2	ОПК-6
11.	Язык Ассемблер	Общая характеристика языков ассемблера: назначение, принципы построения и использования; структура языка, основные группы команд, операторы, средства взаимодействия с операционной системой.	2	ОПК-6
12.	Основы логического программирования	Основы логического программирования. Механизмы логического вывода. Прямая и обратная цепочки рассуждений. Формальные теории первого порядка. Унификация и хорновский клоз.	2	ОПК-6

Согласована на портале № 5554

13.	Основы функциональ ного программиров ания	Основы функционального программирования. Свойства и отличительные особенности функциональных языков программирования. Модель вычислений. Лямбда-исчисление. Аппликация, абстракция, редукция, эквивалентное преобразование. Ленивые и жадные вычисления.	2	ОПК-6
14.	Стандарты языков программиров ания и документиров ание программ	Стандарты языков программирования и документирование программ. Стандарты языков программирования. Различия в стандартах языков программирования, стандартизация ISO. Стандарты документирования программ. ГОСТ ЕСПД. UML диаграммы.	2	ОПК-6
	Итого		**36**	

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

№ п/п	Наименование обеспечивающих (предыдущих) и обеспечиваемых (последующих) дисциплин	№ № разделов данной дисциплины из табл.5.1, для которых необходимо изучение обеспечивающих (предыдущих) и обеспечиваемых (последующих) дисциплин													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
Предшествующие дисциплины															
1.	Математическая логика и теория алгоритмов												+		
2.	Основы программирования		+		+				+	+		+			
Последующие дисциплины															
1.	Базы данных	+	+			+			+						
2.	Системное программирование		+	+	+		+				+	+			+

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

Перечень компетенций	Виды занятий					Формы контроля
	Л	Лаб	Пр.	КР/КП	СРС	
ОПК-6	+	Не предусмотрено	+	Не предусмотрено	+	Опрос на лекции, практическом занятии. Отчет по индивидуальной практической работе. Контрольная работа.

Л – лекция, Пр – практические и семинарские занятия, Лаб – лабораторные работы, КР/КП – курсовая работа/проект, СРС – самостоятельная работа студента

6. Методы и формы организации обучения

Технологии интерактивного обучения при разных формах занятий в часах

Методы / Формы	Лекции (час)	Практические занятия (час)	Всего
IT-методы		8	8
Обсуждение материала в ходе мультимедийных презентаций	8		8
Итого интерактивных занятий	8	8	16

7. Лабораторный практикум

№ п/п	№ раздела дисциплины из табл. 5.1	Наименование лабораторных работ	Трудо-емкость (час.)	Компетенции ОК, ПК, ПСК
Не предусмотрено				

Согласована на портале № 5554

8. Практические занятия (семинары)

№ п/п	№ раздела дисциплины из табл. 5.1	Тематика практических занятий (семинаров)	Трудо-емкость (час.)	Компетенции ОК, ПК, ПСК
1.	1	Анализ задачи. Абстракция программ и данных. Синтаксис языка программирования	4	ОПК-6
2.	1	Обработка текстов на естественном языке, синтаксические анализаторы	4	ОПК-6
3.	2	Вещественные числа. Ошибки при работе с вещественными числами	4	ОПК-6
4.	2	Рекурсия. Типы рекурсий	4	ОПК-6
5.	5	Генерирование и обработка исключительных ситуаций	4	ОПК-6
6.	3	Объектно-ориентированное программирование: модель предметной области и модель проектирования	4	ОПК-6
7.	3	Основные приемы объектно-ориентированного программирования	4	ОПК-6
8.	11	Язык логического программирования Prolog	4	ОПК-6
9.	12	Язык функционального программирования Haskell	4	ОПК-6
	Итого		**36**	

9. Самостоятельная работа

№ п/п	№ раздела дисциплины из табл. 5.1	Виды самостоятельной работы *(детализация)*	Трудо-емкость (час.)	Компе-тенции ОК, ПК, ПСК	Контроль выполнения работы (Опрос, тест, дом.задание, и т.д)
1.	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14	Проработка лекционного материала, подготовка к контрольной работе по курсу лекций	18	ОПК-6	Опрос на лекции, контрольная работа
2.	1, 2, 3, 5, 12, 13	Подготовка к практическим занятиям	18	ОПК-6	Опрос на практическом занятии

10. Примерная тематика курсовых проектов (работ)

Не предусмотрено

11. Рейтинговая система для оценки успеваемости студентов

Таблица 11.1 Балльные оценки для элементов контроля.

Элементы учебной деятельности	Максимальный балл на 1-ую КТ с начала семестра	Максимальный балл за период между 1КТ и 2КТ	Максимальный балл за период между 2КТ и на конец семестра	Всего за семестр
Посещение занятий	4	3	3	**10**
Практическая работа	23	25	22	**70**
Компонент своевременности	3	3	4	**10**
Контрольная работа			10	**10**
Итого максимум за период:	**30**	**31**	**39**	**100**
Нарастающим итогом	**30**	**61**	**100**	**100**

Таблица 11.2 Пересчет баллов в оценки за контрольные точки

Баллы на дату контрольной точки	**Оценка**
≥ 90 % от максимальной суммы баллов на дату КТ	5

Согласована на портале № 5554

От 70% до 89% от максимальной суммы баллов на дату КТ	4
От 60% до 69% от максимальной суммы баллов на дату КТ	3
< 60 % от максимальной суммы баллов на дату КТ	2

Таблица 11.3 – Пересчет суммы баллов в традиционную и международную оценку

Оценка (ГОС)	Итоговая сумма баллов, учитывает успешно сданный экзамен	Оценка (ECTS)
5 (отлично) (зачтено)	**90 - 100**	A (отлично)
4 (хорошо) (зачтено)	**85 – 89**	B (очень хорошо)
	75 – 84	C (хорошо)
	70 - 74	D (удовлетворительно)
3 (удовлетворительно) (зачтено)	**65 – 69**	
	60 - 64	E (посредственно)
2 (неудовлетворительно), (не зачтено)	**Ниже 60 баллов**	F (неудовлетворительно)

12. Учебно-методическое и информационное обеспечение дисциплины:

12.1 Основная литература

1. Мещеряков Р.В., Давыдова Е.М. Языки программирования: Учебник / Р. В. Мещеряков, Е. М. Давыдова. - 2-е изд., перераб. и доп. – Томск.: В-Спектр. 2007. - 290с. (100 экз.)

12.2 Дополнительная литература

1. Орлов С. А. Теория и практика языков программирования [Текст] : учебник / С. А. Орлов ; ред. Ю. Сергиенко ; рец.: Б. В. Соколов, А. Я. Пасмуров. - СПб. : ПИТЕР, 2014. - 688 с. (1 экз.)
2. Гради Буч. Объектно-ориентированный анализ и проектирование с примерами приложений на C++ / перевод с англ. И. Романовского и Ф. Андреева.- М: "Бином", СПб.: "Невский диалект". - 558с. (1 экз.)
3. Вольфенгаген В.Э. Конструкции языков программирования: Приемы описания : монография / Вячеслав Эрнстович Вольфенгаген ; Институт актуального образования "ЮрИнфоР-МГУ". Кафедра перспективных компьютерных исследований и информационных технологий. - М. : Центр ЮрИнфоР, 2001. - 278 с. (1 экз.)
4. Камаев В.А. Технологии программирования : Учебник для вузов / В.А. Камаев, В.В. Костерин. - М. : Высшая школа, 2005. - 360 с. (1 экз.)
5. Кручинин В.В. Технологии программирования : Учебное пособие / В.В. Кручинин. – Томск : ТУСУР, 2006. - 271 с. (45 экз.)
6. Ходашинский И.А. Язык ПРОЛОГ в примерах и задачах : учебное пособие для вузов / И.А. Ходашинский. - Томск : ТУСУР, 2006. - 279 с. (31 экз.)
7. Зюзьков В.М. Ленивое функциональное программирование : учебное пособие / В.М. Зюзьков. - Томск : Издательство Томского университета, 2007. - 293 с. (27 экз.)
8. Давыдов В.Г. Программирование и основы алгоритмизации : Учебное пособие для вузов / В. Г. Давыдов. - 2-е изд., стереотип. - М. : Высшая школа, 2005. - 448 с. (69 экз.)

12.3 Учебно-методические пособия и программное обеспечение

Для обеспечения дисциплины используются следующие УМП:

1. Романов А.С. Языки программирования: лабораторный практикум [Электронный ресурс]. – Режим доступа: http://kibevs.tusur.ru/sites/default/files/upload/manuals/yap-lab.pdf.
2. Романов А.С. Языки программирования: методические указания к практическим занятиям [Электронный ресурс]. – Режим доступа: http://kibevs.tusur.ru/sites/default/files/upload/manuals/yap-pract.pdf.
3. Романов А.С. Языки программирования: вопросы к контрольным работам [Электронный ресурс]. – Режим доступа: http://kibevs.tusur.ru/sites/default/files/upload/manuals/yap-kontr.pdf.

Согласована на портале № 5554

4. Романов А.С. Языки программирования: задания на самостоятельную работу [Электронный ресурс]. – Режим доступа: http://kibevs.tusur.ru/sites/default/files/upload/manuals/yap-sam.pdf.

Для обеспечения дисциплины используется следующее программное обеспечение:

1. Виртуальная машина VMware Workstation или VirtualBox.
2. Операционная система Microsoft Windows 10.
3. Среда разработки Microsoft Visual Studio 2015.
4. Среда разработки Qt SDK.
5. Среда веб разработки NetBeanse.
6. Среда разработки для языка Prolog SWI Prolog.
7. Среда разработки для языка Haskell HugIDE.
8. СУБД MySQL.
9. СУБД SQLite.

12.4 Базы данных, информационно-справочные и поисковые системы

Не предусмотрены.

13. Материально-техническое обеспечение дисциплины:

1. Мультимедийная лекционная аудитория.
2. Компьютерный класс с выходом в Интернет.

14. Методические рекомендации по организации изучения дисциплины:

Не предусмотрено.

БИБЛИОТЕКА ТУСУР

Согласована на портале № 5554

Приложение к рабочей программе

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

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

УТВЕРЖДАЮ
Проректор по учебной работе
_____ **П. Е. Троян**

«___» _____ 2016 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

ЯЗЫКИ ПРОГРАММИРОВАНИЯ

Уровень основной образовательной программы:	бакалавриат
Направление(я) подготовки (специальность):	11.03.03 Конструирование и технология электронных средств
Профиль(и):	Проектирование и технология электронно-вычислительных средств
Форма обучения:	очная
Факультет	Безопасности
Кафедра	Комплексной информационной безопасности электронно-вычислительных систем (КИБЭВС)
Курс 2	**Семестр 3**

Учебный план набора 2013 года и последующих лет.

Зачет 3 семестр

Томск 2016

Согласована на портале № 5554

1 Введение

Фонд оценочных средств (ФОС) является приложением к рабочей программе дисциплины (практики) и представляет собой совокупность контрольно-измерительных материалов (типовые задачи (задания), контрольные работы, тесты и др.) и методов их использования, предназначенных для измерения уровня достижения студентом установленных результатов обучения.

ФОС по дисциплине (практике) используется при проведении текущего контроля успеваемости и промежуточной аттестации студентов.

Перечень закрепленных за дисциплиной (практикой) компетенций приведен в таблице 1.

Таблица 1 – Перечень закрепленных за дисциплиной компетенций

Код	Формулировка компетенции	Этапы формирования компетенции
ОПК-6	способность осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий	**Должен знать:** • общие принципы построения и использования современных языков программирования высокого уровня; • язык программирования высокого уровня (объектно-ориентированное программирование); • возможности, классификацию и область применения макрообработки; • обработки исключительных ситуаций. **Должен уметь:** • работать с интегрированной средой разработки программного обеспечения; • использовать шаблоны классов и средства макрообработки; • использовать динамически подключаемые библиотеки. **Должен владеть:** • навыками проектирования программного обеспечения с использованием средств автоматизации.

2 Реализация компетенций

1 Компетенция ОПК-6

ОПК-6: способность осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий.

Согласована на портале № 5554

Для формирования компетенции необходимо осуществить ряд этапов. Этапы формирования компетенции, применяемые для этого виды занятий и используемые средства оценивания представлены в таблице 2.

Таблица 2 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	• Должен знать общие принципы построения и использования современных языков программирования высокого уровня; • язык программирования высокого уровня (объектно-ориентированное программирование); • возможности, классификацию и область применения макрообработки; • обработки исключительных ситуаций.	• Должен уметь работать с интегрированной средой разработки программного обеспечения; • использовать шаблоны классов и средства макрообработки; • использовать динамически подключаемые библиотеки.	• Должен владеть навыками проектирования программного обеспечения с использованием средств автоматизации.
Виды занятий	• Лекции; • Практические занятия; • Самостоятельная работа студента	• Практические занятия; • Выполнение домашнего задания; • Самостоятельная работа студента	• Лекции; • Практические занятия; • Самостоятельная работа студента
Используемые средства оценивания	• Контрольная работа; • Зачет	• Оформление и защита домашнего задания; • Конспект самостоятельной работы;	• Оформление и защита домашнего задания; • Конспект самостоятельной работы; • Зачет

Общие характеристики показателей и критериев оценивания компетенции на всех этапах приведены в таблице 3.

Таблица 3 – Общие характеристики показателей и критериев оценивания компетенции по этапам

Показатели и критерии	Знать	Уметь	Владеть
Отлично (высокий	Обладает	Обладает диапазоном	Контролирует работу,

Согласована на портале № 5554

уровень)	фактическими и теоретическими знаниями в пределах изучаемой области с пониманием границ применимости	практических умений, требуемых для развития творческих решений, абстрагирования проблем	проводит оценку, совершенствует действия работы
Хорошо (базовый уровень)	Знает факты, принципы, процессы, общие понятия в пределах изучаемой области	Обладает диапазоном практических умений, требуемых для решения определенных проблем в области исследования	Берет ответственность за завершение задач в исследовании, приспосабливает свое поведение к обстоятельствам в решении проблем
Удовлетворительно (пороговый уровень)	Обладает базовыми общими знаниями	Обладает основными умениями, требуе-мыми для выполнения простых задач	Работает при прямом наблюдении

Формулировка показателей и критериев оценивания данной компетенции приведена в таблице 4.

Таблица 4 – Показатели и критерии оценивания компетенции на этапах

Показатели и критерии	**Знать**	**Уметь**	**Владеть**
Отлично (высокий уровень)	• *знает методы поиска, хранения, обработки и анализа информации, возможности их реализации в различных современных языках программирования,* • *знает принципы построения современных актуальных языков программирования.* • *знает принципы объектно-ориентированного, анализа, программирования и проектирования* • *знает принципы обработки*	• *умеет работать с информацией из различных источников и баз данных, представлять её в требуемом формате путем написания программ на различных современных языках программирования,* • *умеет обосновать выбор языка программирования для решения конкретной задачи, способен переписать исходных код с одного языка программирования*	• *уверенно владеет информационными, компьютерными и сетевыми технологиями для полноценного использования основных возможностей нескольких языков программирования,* • *владеет навыками проектирования программного обеспечения с использованием средств автоматизации* • *владеет основными современными языками программирования.* • *владеет навыками объектно-*

Согласована на портале № 5554

	исключительных ситуаций.	*на другой.* • *умеет работать в различных интегрированных средах разработки программного обеспечения* • *использует шаблоны классов и средства макрообработки, умеет проводить объектную декомпозицию для любой предметной области* • *использует динамически подключаемые библиотеки.* • *комплексно подходит к вопросу возникновения исключительных ситуаций, предвидит их и применяет средства обработки.* • *умеет разрабатывать динамически подключаемые библиотеки.*	*ориентированного программирования.* • *владеет навыками обработки исключительных ситуаций и макрообработки в различных языках программирования.*
Хорошо (базовый уровень)	• *знает методы поиска, хранения, обработки и анализа информации, возможности их реализации в базовом языке программирования,* • *знает принципы построения нескольких языков высокого уровня.* • *понимает принципы объектно-ориентированного анализа,*	• *умеет работать с информацией из различных источников и баз данных, представлять её в требуемом формате, применяя базовый язык программирования,* • *свободно применят язык программирования высокого уровня.* • *выявляет исключительные ситуации по ходу решения задачи и*	• *владеет информационными, компьютерными и сетевыми технологиями для полноценного использования основных возможностей базового языка программирования,* • *владеет базовыми навыками проектирования программного обеспечения с использованием средств*

Согласована на портале № 5554

	проектирования и программирования. • *понимает принципы обработки исключительных ситуаций.*	*умеет их обработать.* • *понимает принципы объектно-ориентированного программирования, способен дополнять существующую предметную область.* • *умеет использовать динамически подключаемые библиотеки.*	*автоматизации* • *владеет как минимум одним языком программирования высокого уровня* • *владеет навыками обработки исключительных ситуаций*
Удовлетворительно (пороговый уровень)	• *знает теоретические основы основных методов поиска, хранения, обработки и анализа информации,* • *дает определения основных понятий;* • *знает основные методы решения типовых задач программирования*	• *умеет работать с информацией из различных источников и баз данных,* • *умеет работать с интегрированными средами разработки программного обеспечения со справочной литературой;* • *умеет воспроизводить программы, в которых применяются шаблоны классов, средства макрообработки, динамически подключаемые библиотеки*	• *владеет теоретическими знаниями в области информационных, компьютерных и сетевых технологий,* • *владеет терминологией предметной области знания;* • *способен анализировать принципы работы готовых алгоритмов решения типовых профессиональных задач.*

3 Типовые контрольные задания

Для реализации вышеперечисленных задач обучения используются следующие материалы:

– типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы, в составе:

Согласована на портале № 5554

Контрольная работа:

1. Основные концепции языков программирования.

Выполнение домашних заданий:

1. Анализ задачи. Абстракция программ и данных. Синтаксис языка программирования.
2. Обработка текстов на естественном языке, синтаксические анализаторы.
3. Вещественные числа. Ошибки при работе с вещественными числами.
4. Рекурсия. Типы рекурсий.
5. Генерирование и обработка исключительных ситуаций.
6. Объектно-ориентированное программирование. Модель предметной области и модель проектирования.
7. Основные приемы объектно-ориентированного программирования.
8. Язык логического программирования Prolog.
9. Язык функционального программирования Haskell.

Темы для самостоятельной работы:

1. Основные концепции, история, сайты, компиляторы и расширения языка «Фортран».
2. Основные концепции, история, сайты, компиляторы и расширения языка «Пролог».
3. Основные концепции, история, сайты, компиляторы и расширения языка «Java».
4. Основные концепции, история, сайты, компиляторы и расширения языка «ASP».
5. Основные концепции, история, сайты, компиляторы и расширения языка «PHP».
6. Основные концепции, история, сайты, компиляторы и расширения языка «Perl».
7. Основные концепции, история, сайты, компиляторы и расширения языка «Ruby».
8. Основные концепции, история, сайты, компиляторы и расширения языка «Python».
9. Построение компиляторов. Основные концепции.
10. Компиляторы для C++. Сравнительный анализ.
11. Языки программирования для сети Интернет. Сравнительный анализ.
12. Объектно-ориентированное программирование. Основные положения, правила создания классов и объектов.
13. Защита программ от несанкционированного копирования.
14. Защита программ от анализа.
15. Взлом программ с защитами. Дизассемблеры.
16. Оценка качества программ. Методы, стандарты и основные положения.
17. Методы оценки стоимости программных продуктов.
18. Оценка сложности вычисления операций.
19. Методы оценки трудоемкости создания программ.

Согласована на портале № 5554

20. Нормативы написания исходного кода.
21. Документирование программ. Основные стандарты.
22. Распространение программ. Shareware, Freeware.
23. Вычисление контрольных сумм файлов программ. Алгоритмы и реализации.
24. Шуточные и эзотерические языки программирования.
25. Методы обфускации исходного кода.
26. Программирование для операционной системы MacOS.
27. Программирование для мобильной платформы Android.
28. Облачные вычисления.
29. Стандарт HTML5.
30. Программирование искусственного интеллекта.
31. Унифицированный язык моделирования UML.
32. Модель компонентных объектов COM.
33. Интеллектуальный анализ данных (Data Mining).
34. Сравнительный анализ языков функционального программирования.
35. Обработка текстов на естественных языках.
36. Основные концепции, история, сайты, компиляторы и расширения языка «Scala»
37. MPI (Message Passing Interface)
38. Автоматическое построение онтологий для извлечения знаний из текста.
39. Библиотека Qt.
40. Параллельное программирование для графических ускорителей. Технология CUDA.
41. Кроссплатформенное программирование. Сравнительный анализ языков программирования.
42. Сравнительный анализ языков логического программирования.
43. Искусственные нейронные сети. Современное состояние вопроса.
44. Динамические библиотеки.
45. Технология ActiveX.
46. Методологии тестирования программного обеспечения.
47. Автоматизированное тестирование программного обеспечения.
48. Основные концепции, история, сайты, компиляторы и расширения языка «R».
49. Основные подходы и принципы разработки дизайна программного обеспечения
50. Особенности, основные подходы и принципы разработка видеоигр.
51. Системы контроля версий программ. Сравнительный анализ.
52. Continuous Integration, непрерывная интеграция и автоматизация сборок проекта.
53. Методы монетизации программного обеспечения.
54. Программирование для мобильной платформы iOS.

Согласована на портале № 5554

55. Программирование для мобильной платформы Windows Phone.
56. Основные концепции, история, сайты, компиляторы и расширения языка «Wolfram language».
57. Аудит информационной безопасности веб-сайта.
58. Паттерны объектно-ориентированного проектирования.
59. Программирование микроконтроллеров.
60. Управление разработкой программного обеспечения. Стандарты, методы, особенности.

Вопросы к зачету:

1. Язык программирования. Общие принципы построения и использования языков программирования.
2. Стандарты языков программирования.
3. Лямбда-исчисление. Аппликация, абстракция, редукция, преобразование.
4. Списки и функциональные выражения в функциональных языках программирования.
5. Механизмы и средства взаимодействия программы с операционной системой.
6. Функциональное программирование. Основные положения. Основные отличия от других типов языков программирования.
7. Классификация языков программирования. Близость языков программирования к естественному языку.
8. Унификация и хорновский клоз в логических языках программирования.
9. Модель вычислений функциональных языков программирования.
10. Языки программирования низкого уровня.
11. Средства разработки графического интерфейса пользователя. Эргономические свойства человеко-машинного интерфейса.
12. Процедурные языки программирования. Основные отличия от других типов языков.
13. Обоснование выбора языка программирования.
14. Перегрузка в языках программирования.
15. Логические языки программирования. Основные положения и понятия. Отличия от других типов языков программирования.
16. Объектно-ориентированные языки программирования. Основные отличия от других концепций языков программирования.
17. Полиморфизм в объектно-ориентированном программировании. Виртуальные функции. Таблицы виртуальных функций.
18. Структура языка программирования. Синтаксис и семантика языков программирования. Расширенная форма Бэкуса-Наура.

Согласована на портале № 5554

19. Наследование в объектно-ориентированном программировании. Множественное наследование. Проблемы множественного наследования.
20. Инкапсуляция в объектно-ориентированном программировании. Контроль доступа.
21. Понятие класса и объекта в объектно-ориентированном программировании. Атрибуты, методы, конструктор и деструктор, статические члены класса.
22. Диаграммы классов UML. Основные элементы и обозначения.
23. Данные. Средства описания данных. Типизация языка.
24. Преобразование типов. Контроль соответствия типов данных.
25. Объектно-ориентированное программирование. Основные концепции объектно-ориентированного программирования.
26. Современные интегрированные среды разработки программ. Основные компоненты среды программирования.
27. Декомпозиция программ.
28. Трансляторы. Интерпретация и компиляция.
29. Макропроцессоры и макрогенераторы.
30. Потоки и процессы. Сходства и различия.
31. Мониторы и защищаемые переменные в параллельном программировании.
32. Семафоры в параллельном программировании. Типы семафоров.
33. Отладчики. Генераторы кода и приложений.
34. Параллельная обработка данных и параллелизм. Параллельное и распределенное программирование.
35. Основные проблемы параллельного и распределенного программирования.
36. Оценка максимально возможного параллелизма.
37. Основные модели параллельного программирования.
38. Оптимизатор. Основные функции оптимизатора.
39. Обработка исключительных ситуаций. Иерархия исключительных ситуаций. Виды исключительных ситуаций.
40. Операторы обработки исключительных ситуаций в различных языках программирования.
41. Схема обработки исключительных ситуаций Б. Мейера.
42. Элементарные типы данных.
43. Перегрузка данных, операторов, методов.
44. Составные типы данных.
45. Механизмы логического вывода. Прямая и обратная цепочки рассуждений.
46. Пространство имен, область видимости, время жизни переменных.
47. Ошибки при работе с вещественными числами. Смешанная арифметика.

Согласована на портале № 5554

48. Операторы выбора и условные операторы.
49. Вещественные числа. Способы представления. Операции над вещественными числами.
50. Оператор присваивания. Операторы цикла.
51. Распределение памяти при выполнении программы.
52. Куча. Менеджер кучи. Фрагментация динамической памяти.
53. Концепция виртуальной памяти. Страничная организация памяти.
54. Сегментный принцип организации памяти. Сегментация памяти.
55. Указатели. Операции над указателями. Типизированные и нетипизированные указатели.
56. Динамические структуры данных. Реализация динамических структур данных с помощью указателей.
57. Библиотеки программ и классов. Статические и динамические библиотеки. Критерии проектирования библиотек.
58. Подпрограммы. Формальные и фактические параметры подпрограмм.
59. Передача параметров подпрограмме.
60. Программный стек и его изменение.
61. Рекурсивный и итерационный методы решения задач. Виды рекурсий.
62. Общая характеристика языков ассемблера: назначение, принципы построения и использования; структура языка.
63. Сериализация и десериализация. Методы сериализации объектов в базу данных.
64. Динамическая диспетчеризация.
65. Родовые (настраиваемые) сегменты.
66. Шаблоны.
67. Вариантные записи.
68. Средства описания действий над данными. Операторы, выражения, модули, блоки в языках программирования.
69. Ленивые и жадные вычисления в процедурном и функциональном программировании.
70. Языки высокого уровня.
71. Показатели качества программных средств.
72. Отношения между классами в объектно-ориентированном программировании.
73. Файлы, обработка файлов. Типы доступа к файлам.
74. Абстрактные типы данных: инкапсуляция, спецификация, реализация, параметризация.
75. Основные группы команд, операторы, средства взаимодействия с операционной системой в языках ассемблера.

Согласована на портале № 5554

4 Методические материалы

Для обеспечения процесса обучения и решения задач обучения используются следующие материалы:

– методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций, в составе:

Методические материалы:

1. Мещеряков Р.В., Давыдова Е.М. Языки программирования: Учебник / Р. В. Мещеряков, Е. М. Давыдова. - 2-е изд., перераб. и доп. – Томск.: В-Спектр. 2007. - 290с. (100 экз.)

2. Романов А.С. Языки программирования: лабораторный практикум [Электронный ресурс]. – Режим доступа: http://kibevs.tusur.ru/sites/default/files/upload/manuals/yap-lab.pdf.

3. Романов А.С. Языки программирования: методические указания к практическим занятиям [Электронный ресурс]. – Режим доступа: http://kibevs.tusur.ru/sites/default/files/upload/manuals/yap-pract.pdf.

4. Романов А.С. Языки программирования: вопросы к контрольным работам [Электронный ресурс]. – Режим доступа: http://kibevs.tusur.ru/sites/default/files/upload/manuals/yap-kontr.pdf.

5. Романов А.С. Языки программирования: задания на самостоятельную работу [Электронный ресурс]. – Режим доступа: http://kibevs.tusur.ru/sites/default/files/upload/manuals/yap-sam.pdf.

Согласована на портале № 5554