

Кафедра автоматики

“ ” Г.

Рабочая программа составлена на основании федерального государственного образовательного стандарта высшего образования по направлению (специальности): 27.03.04 Управление в технических системах

ФГОС введен в действие приказом №1171 от 20.10.2015 г. , дата утверждения: 12.11.2015 г.

Место дисциплины в структуре учебного плана: Блок 1, вариативная

Рабочая программа разработана на основе компетентностной модели выпускника по направлению (специальности): 27.03.04 Управление в технических системах

Рабочая программа обсуждена на заседании кафедры

АВТ, протокол заседания кафедры №10/1 от 20.06.2017

Утверждена на совете факультета автоматики и вычислительной техники, протокол № 6 от 21.06.2017

Программу разработал:

доцент, к.т.н. Романников Д. О.

Заведующий кафедрой:

доцент, д.т.н. Жмудь В. А.

Ответственный за образовательную программу:

заведующий кафедрой Жмудь В. А.

1. Внешние требования

Таблица 1.1

Компетенция ФГОС: 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; в части следующих результатов обучения:	
у10. уметь использовать языки и системы программирования для решения профессиональных задач	
у7. уметь использовать специализированные программные средства при решении профессиональных задач	
у8. уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов	
Компетенция ФГОС: ОПК.7 способность учитывать современные тенденции развития электроники, измерительной и вычислительной техники, информационных технологий в своей профессиональной деятельности; в части следующих результатов обучения:	
з5. знать языки программирования и среды разработки	
з6. знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	
з7. знать принципы построения архитектуры программного обеспечения и виды архитектур программного обеспечения	
з8. знать типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения	
у4. уметь работать с программными пакетами для разработки, моделирования и исследования электрических схем	
Компетенция ФГОС: ПК.6 способность производить расчеты и проектирование отдельных блоков и устройств систем автоматизации и управления и выбирать стандартные средства автоматики, измерительной и вычислительной техники для проектирования систем автоматизации и управления в соответствии с техническим заданием; в части следующих результатов обучения:	
з8. знать базовые языки программирования промышленных универсальных контроллеров	

2. Требования НГТУ к результатам освоения дисциплины

Таблица 2.1

Результаты изучения дисциплины по уровням освоения (иметь представление, знать, уметь, владеть)	Формы организации занятий
<i>Технология программирования</i>	
ОПК.6.у10 уметь использовать языки и системы программирования для решения профессиональных задач	
1. уметь использовать языки и системы программирования для решения профессиональных задач	Лекции; Лабораторные работы; Самостоятельная работа
ОПК.6.у7 уметь использовать специализированные программные средства при решении профессиональных задач	
2. уметь использовать специализированные программные средства при решении профессиональных задач	Лекции; Лабораторные работы; Самостоятельная работа
ОПК.6.у8 уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов	
3. уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов	Лекции; Лабораторные работы; Самостоятельная работа
ОПК.7.з8 знать типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения	
4. знать типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения	Лекции; Лабораторные работы
ОПК.7.з5 знать языки программирования и среды разработки	

5.знать языки программирования и среды разработки	Лекции; Лабораторные работы; Самостоятельная работа
ОПК.7.36 знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	
6.знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	Лекции; Лабораторные работы; Самостоятельная работа
ОПК.7.37 знать принципы построения архитектуры программногo обеспечения и виды архитектур программногo обеспечения	
7.знать принципы построения архитектуры программногo обеспечения и виды архитектур программногo обеспечения	Лекции; Лабораторные работы; Самостоятельная работа
ОПК.7.у4 уметь работать с программными пакетами для разработки, моделирования и исследования электрических схем	
8.уметь работать с программными пакетами для разработки, моделирования и исследования электрических схем	Лекции; Лабораторные работы
ПК.6.38 знать базовые языки программирования промышленных универсальных контроллеров	
9.знать базовые языки программирования промышленных универсальных контроллеров	Лекции; Лабораторные работы

3. Содержание и структура учебной дисциплины

Таблица 3.1

Темы лекций	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 4				
Дидактическая единица: Технологии программирования и разработки ПО.				
1. Традиционные технологии программирования	0	2	1	Технология как совокупность знаний, навыков (рекомендаций) и инструмента. Исторический обзор технологий программирования (ТП). "Традиционное" ООП с использованием блок-схем, его связь с компьютерной архитектурой. "Структурное программирование" (СП) как пошаговое, нисходящее модульное программирование. Модульное иерархическое представление алгоритма и данных в технологии СП. Цикл проектирования программы при использовании СП. Программа как единство потока команд и потока данных. Первичность алгоритма по отношению к данным и их взаимосвязь в традиционных технологиях

2. Объектно-ориентированное программирование (ООП) как единство алгоритма и данных.	0	4	2	Объектно-ориентированное программирование (ООП) как единство алгоритма и данных. Технологическое и языковое (синтаксическое) определение класса и объекта. Основные понятия технологии ООП: объект, метод, класс, инкапсуляция (закрытость, самодостаточность), переопределение операций. "Эпизодическое" ООП как использование в традиционной технологии отдельных классов. "Тотальное ООП" как программирование "от класса к классу". Наследование. Полиморфизм. Взаимодействие объектов в программе по принципу "объект - метод - объект"
Дидактическая единица: Объектно-ориентированная среда разработки				
3. Платформенно-независимое и мобильное программное обеспечение (ПО)	0	4	3	Платформенно-независимое и мобильное программное обеспечение (ПО). Особенности Си как средства разработки мобильного, кроссплатформенного системного ПО. Системы с промежуточным кодом: C#, Java. Основные особенности языка Java: объектно-ориентированный подход, полиморфизм, ссылочные типы, интерпретация, динамическая загрузка и связывание классов Обеспечение платформенной независимости. Внутреннее представление Java-программы. Объектная модель программы в виртуальной машине (JVM). Классы Object и Class, их методы. Организация JVM. Система мета-объектов - описателей классов. Динамическая загрузка классов в JVM, динамическое связывание методов

4. Язык программирования Java. Сравнение с Си++	0	2	4	Язык программирования Java. Базовые типы данных. Класс String, его особенности и методы. Классы - обертки для базовых типов данных (int и Integer), их назначение. Ссылочные типы. Идентичность ссылочных типов в Java указателей на динамические объекты в Си++. Особенности работы со ссылочными типами, динамическое распределение памяти, "сбор мусора". Операции над ссылками.
Дидактическая единица: Распределенные клиент-серверные приложения				
5. Полиморфизм, абстрактные классы и интерфейсы. Вложенные и анонимные классы	0	4	5	Java как среда "тотального" объектно-ориентированного программирования (ООП). Наследование и приведение типов. Класс Object. Расширение и сужение ссылочного типа при переходе от класса к классу. Операция instanceof. Встроенный полиморфизм методов. Статические элементы классов. Их использование для задания констант и функций. Метод main. Инициализаторы и конструкторы. Многообразие вариантов инициализации объектов. Явный вызов конструкторов. Клонирование (копирование) объектов. Полиморфизм как основа создания абстракций и интерфейсов. Наследование в Java. Древоподобная система классов при одиночном наследовании. Класс Object, его назначение. Приведение типов ссылок при работе с системой наследуемых классов. Сужение и расширение. Динамические ошибки сужения. Работа с данными базового класса. Доступ к базовому классу (super). Встроенный полиморфизм методов в Java. Интерфейс (interface). Использование интерфейсов для доступа к объектам различных классов. Наследование интерфейсов. Абстрактный базовый класс.

6. Надежность и устойчивость программ	0	2	6	Надежность и устойчивость программного обеспечения. Исключения, их генерация и обработка. Классы исключений. Синтаксический контроль исключений. Исключения, генерируемые JVM
7. Разработка оконных приложений	0	4	5, 6	Классы оконных приложений (Frame и Applet). Работа с графикой (классы Graphics, Color, Font). Классы элементов управления (Component, Canvas, Label, Button, TextField, TextArea, Choice, List, MenuBar, Menu, MenuItem). Их основные методы. Конструирование объектов управления (списков, меню), их объявление в классе окна. Обработка событий в Java. Классы событий, интерфейсы слушателей, объявление слушателей событий. Способы создания обработчиков событий: использование общего класса-обработчика, классов-адаптеров, анонимных классов
Дидактическая единица: Объектно-ориентированная технология разработки приложений				
8. Приложения с потоковыми и параллельными вычислениями	0	4	7	Внутренний параллелизм в Java-программе. Потоки и их программирование. Класс Thread и интерфейс Runnable. Связь потока с классом-прародителем. Вложенные и анонимные классы. Видимость текущего объекта и объекта-родителя. Использование вложенных классов для создания взаимодействующих потоков. Взаимодействие и синхронизация потоков. Ошибки синхронизации, их особенности. Критическая секция и разделение ресурсов. Примитив synchronized. Блокирование потоков (sleep,suspend,resume), ожидание событий (wait,suspend)

9. Ввод-вывод и обмен данными. Распределенные системы и сетевые приложения	0	6	8, 9	Ввод-вывод и обмен данными в Java. Многоуровневая система потоков ввода-вывода: потоки-источники данных, потоки-представления. Форматы представления текста и особенности работы с ним в Java. Ввод-вывод текстовых файлов. Классы Reader, Writer. Буферизованный ввод-вывод. Понятие двоичного файла. Форматы внутреннего представления примитивных типов данных: целое со знаком и без, символ, вещественное. Двоичные потоки в Java. Объектовые потоки. Сериализация. Алгоритм сериализации системы связанных объектов Понятие клиента и сервера. Пакет Java.net. Установление логических соединений через сокеты. Протоколы прикладного уровня. Работа с протоколами http с использованием классов URL и URLConnection.
10. Проектирование сложных программных систем	0	4	7, 8, 9	Этапы проектирования распределенной системы. Распределений функций: тонкий и толстый клиент. Постоянные и короткие соединения (транзакции). Проектирование протокола. Использование потоков при обслуживании постоянных соединений. Проектирование однопоточкового сервера с короткими соединениями и многопоточкового сервера с постоянными соединениями на примере программы обмена сообщениями (чат). Проектирование сложной распределенной системы на примере p2p-сети с установлением непосредственных соединений между клиентами

Таблица 3.2

Темы лабораторных работ	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 4				

Дидактическая единица: Технологии программирования и разработки ПО.				
1. Изучение среды проектирования NetBeans IDE	4	4	1, 2	Знакомство со средой проектирования Java - NetBeans. Перенос программного кода Си++ готовой программы, подготовка и отладка программы
Дидактическая единица: Объектно-ориентированная среда разработки				
2. Разработка оконного приложения в Java	4	8	3, 4	Изучение стандартных средств визуального проектирования оконного приложения. Разработка собственных классов обработки событий для пользовательского интерфейса с программируемой структурой (массив текстовых полей и т.п.)
Дидактическая единица: Распределенные клиент-серверные приложения				
3. Разработка многопоточковых приложений	4	8	5, 6	Разработка приложения с движущимися графическими объектами нескольких видов, управляемыми отдельными потоками: создание, уничтожение и изменение параметров объектов.
Дидактическая единица: Объектно-ориентированная технология разработки приложений				
4. Ввод-вывод данных	6	8	7, 9	Сохранение данных о движущихся графических объектах в тестовом, двоичном и сериализуемом файле
5. Разработка распределенного приложения	0	8	8, 9	Разработка приложения, имеющего клиентскую и серверную компоненты для передачи через сеть данных о движущихся графических объектах

4. Самостоятельная работа обучающегося

№	Виды самостоятельной работы	Ссылки на результаты обучения	Часы на выполнение	Часы на консультации
Семестр: 4				
1	Курсовая работа	2, 3	20	3
, подробная информация приведена в приложении №3 : Технология программирования : методические указания к курсовой работе по направлению 09.03.01 "Информатика и вычислительная техника" очной формы обучения / Новосиб. гос. техн. ун-т ; [сост. Д. О. Романников]. - Новосибирск, 2016. - 14, [2] с.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000228566				
2	Подготовка к занятиям	1	12	0
, подробная информация приведена в приложениях №1 №2 : Технология программирования : методические указания к лабораторным работам по направлению 230100 "Информатика и вычислительная техника" очной формы обучения / Новосиб. гос. техн. ун-т ; [сост.: Д. О. Романников, А. В. Марков]. - Новосибирск, 2015. - 34 с.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000220033				
3	Дополнительная учебная деятельность	5, 6	12	0

, подробная информация приведена в приложениях №1 №2 : Технология программирования : методические указания к лабораторным работам по направлению 230100 "Информатика и вычислительная техника" очной формы обучения / Новосиб. гос. техн. ун-т ; [сост.: Д. О. Романников, А. В. Марков]. - Новосибирск, 2015. - 34 с.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000220033 Технология программирования : методические указания к курсовой работе по направлению 09.03.01 "Информатика и вычислительная техника" очной формы обучения / Новосиб. гос. техн. ун-т ; [сост. Д. О. Романников]. - Новосибирск, 2016. - 14, [2] с.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000228566				
4	Подготовка к аттестации	5, 6, 7	20	3
, подробная информация приведена в приложениях №1 №2 : Технология программирования : методические указания к лабораторным работам по направлению 230100 "Информатика и вычислительная техника" очной формы обучения / Новосиб. гос. техн. ун-т ; [сост.: Д. О. Романников, А. В. Марков]. - Новосибирск, 2015. - 34 с.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000220033 Технология программирования : методические указания к курсовой работе по направлению 09.03.01 "Информатика и вычислительная техника" очной формы обучения / Новосиб. гос. техн. ун-т ; [сост. Д. О. Романников]. - Новосибирск, 2016. - 14, [2] с.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000228566				

5. Технология обучения

Для организации и контроля самостоятельной работы обучающихся, а также проведения консультаций применяются информационно-коммуникационные технологии (табл. 5.1).

Таблица 5.1

Деятельность	Информационно-коммуникационные технологии
Информирование	e-mail; Портал НГТУ
Консультирование	e-mail; Личный типовой сайт
Контроль	e-mail; Портал НГТУ
Размещение учебных материалов	e-mail; Среда электронного обучения НГТУ

6. Правила аттестации обучающихся по учебной дисциплине

Для аттестации обучающихся по дисциплине используется балльно-рейтинговая система (БРС), позволяющая выставять оценки по традиционной шкале и 15-уровневой ECTS. Краткая информация о БРС приведена в табл. 6.1.

Таблица 6.1

Оцениваемые виды деятельности обучающихся	Максимальный балл
Семестр: 4	
<i>Лабораторная:</i>	40
<i>Курсовая работа:</i>	20
<i>Экзамен:</i>	40

В таблице 6.2 представлено соответствие форм контроля заявляемым требованиям к результатам освоения дисциплины.

Таблица 6.2

Коды компетенций ФГОС	Результаты обучения	Формы контроля	
		Защита КП/КР	Экзамен
ОПК.6	у10. уметь использовать языки и системы программирования для решения профессиональных задач		+
	у7. уметь использовать специализированные программные средства при решении профессиональных задач		+
	у8. уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов		+
ОПК.7	з5. знать языки программирования и среды разработки		+
	з6. знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	+	+
	з7. знать принципы построения архитектуры программного обеспечения и виды архитектур программного обеспечения		+
	з8. знать типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения	+	+
	у4. уметь работать с программными пакетами для разработки, моделирования и исследования электрических схем		+
ПК.6	з8. знать базовые языки программирования промышленных универсальных контроллеров		+

Фонд оценочных средств по дисциплине представлен в приложении № 1 к рабочей программе.

7. Литература

Основная литература

1. Васильев А. Н. Java. Объектно-ориентированное программирование : для магистров и бакалавров : базовый курс по объектно-ориентированному программированию : [учебное пособие] / А. Н. Васильев. - СПб. [и др.], 2011. - 395, [1] с. : ил., табл.
2. Машнин Т. С. Современные Java технологии на практике / Тимур Машнин. - СПб., 2010. - 552 с. : ил., табл. + 1 CD-ROM.
3. Соломон М. К. Oracle. Программирование на языке Java / Мартин К. Соломон, Нирва Мориссо-Леруа, Джули Басу ; [пер. И. Дранишников]. - М., 2010. - 484 с. : ил.
4. Флэнаган Д. Java. Справочник / Дэвид Флэнаган ; [пер. с англ. К. Волкова и В. Шальнева]. - СПб. ;, 2004. - 1039 с. : ил.
5. Лафоре Р. Объектно-ориентированное программирование в C++ / Р. Лафоре ; [пер. с англ. А. Кузнецова, М. Назарова, В. Шрага]. - СПб. [и др.], 2011. - 923 с. : ил.

Дополнительная литература

1. Соломон М. К. Oracle. Программирование на языке Java / Мартин К. Соломон, Нирва Мориссо-Леруа, Джули Басу ; [пер. И. Дранишников]. - М., 2010. - 484 с. : ил.
2. Эккель Б. Философия Java / Брюс Эккель ; [пер. с англ. Е. Матвеев]. - СПб. [и др.], 2010. - 637 с.

3. Машнин Т. С. Современные Java технологии на практике / Тимур Машнин. - СПб., 2010. - 552 с. : ил., табл. + 1 CD-ROM.

Интернет-ресурсы

1. ЭБС НГТУ : <http://elibrary.nstu.ru/>
2. ЭБС «Издательство Лань» : <https://e.lanbook.com/>
3. ЭБС IPRbooks : <http://www.iprbookshop.ru/>
4. ЭБС "Znaniium.com" : <http://znaniium.com/>
5. :

8. Методическое и программное обеспечение

8.1 Методическое обеспечение

1. Технология программирования : методические указания к лабораторным работам по направлению 230100 "Информатика и вычислительная техника" очной формы обучения / Новосиб. гос. техн. ун-т ; [сост.: Д. О. Романников, А. В. Марков]. - Новосибирск, 2015. - 34 с.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000220033
2. Технология программирования : методические указания к курсовой работе по направлению 09.03.01 "Информатика и вычислительная техника" очной формы обучения / Новосиб. гос. техн. ун-т ; [сост. Д. О. Романников]. - Новосибирск, 2016. - 14, [2] с.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000228566

8.2 Специализированное программное обеспечение

1 Windows

9. Материально-техническое обеспечение

Компьютерный класс

№	Наименование	Назначение
1	Компьютерный класс (Компьютеры объединены в локальную сеть с выходом в Internet)	Компьютеры объединены в локальную сеть с выходом в Internet

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»

Кафедра автоматики

“УТВЕРЖДАЮ”
ДЕКАН АВТФ
к.т.н., доцент И.Л. Рева

“ ” Г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

УЧЕБНОЙ ДИСЦИПЛИНЫ

Технология программирования

Образовательная программа: 27.03.04 Управление в технических системах, профиль: Автоматика и управление

1. Обобщенная структура фонда оценочных средств учебной дисциплины

Обобщенная структура фонда оценочных средств по дисциплине **Технология программирования** приведена в Таблице.

Таблица

Формируемые компетенции	Показатели сформированности компетенций (знания, умения, навыки)	Темы	Этапы оценки компетенций	
			Мероприятия текущего контроля (курсовой проект, РГЗ(Р) и др.)	Промежуточная аттестация (экзамен, зачет)
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	у7. уметь использовать специализированные программные средства при решении профессиональных задач	Объектно-ориентированное программирование (ООП) как единство алгоритма и данных. Изучение среды проектирования NetBeans IDE		Экзамен, вопросы 1-4
ОПК.6	у8. уметь использовать элементарные навыки алгоритмизации и программирования на одном из языков высокого уровня как средство программного моделирования изучаемых объектов и процессов	Платформенно-независимое и мобильное программное обеспечение (ПО) Разработка оконного приложения в Java		Экзамен, вопросы 5-7
ОПК.6	у10. уметь использовать языки и системы программирования для решения профессиональных задач	Изучение среды проектирования NetBeans IDE Традиционные технологии программирования		Экзамен, вопросы 8-10
ОПК.7 способность учитывать современные тенденции развития электроники, измерительной и вычислительной техники, информационных технологий в своей профессиональной деятельности	з5. знать языки программирования и среды разработки	Полиморфизм, абстрактные классы и интерфейсы. Вложенные и анонимные классы Разработка многопоточковых приложений Разработка оконных приложений		Экзамен, вопросы 11-13
ОПК.7	з6. знать синтаксис выбранного языка программирования, особенности программирования на этом языке,	Надежность и устойчивость программ Разработка многопоточковых приложений Разработка оконных приложений	Курсовая работа, разделы...	Экзамен, вопросы 14-17

	стандартные библиотеки языка программирования			
ОПК.7	з7. знать принципы построения архитектуры программного обеспечения и виды архитектур программного обеспечения	Ввод-вывод данных Приложения с потоковыми и параллельными вычислениями Проектирование сложных программных систем		Экзамен, вопросы 18-21
ОПК.7	з8. знать типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения	Разработка оконного приложения в Java Язык программирования Java. Сравнение с Си++	Курсовая работа, разделы...	Экзамен, вопросы 22-25
ОПК.7	у4. уметь работать с программными пакетами для разработки, моделирования и исследования электрических схем	Ввод-вывод и обмен данными. Распределенные системы и сетевые приложения Проектирование сложных программных систем Разработка распределенного приложения		Экзамен, вопросы 26-30
ПК.6/ПК способность производить расчеты и проектирование отдельных блоков и устройств систем автоматизации и управления и выбирать стандартные средства автоматики, измерительной и вычислительной техники для проектирования систем автоматизации и управления в соответствии с техническим заданием	з8. знать базовые языки программирования промышленных универсальных контроллеров	Проектирование сложных программных систем		Экзамен, вопросы 31-40

2. Методика оценки этапов формирования компетенций в рамках дисциплины.

Промежуточная аттестация по **дисциплине** проводится в 4 семестре - в форме экзамена, который направлен на оценку сформированности компетенций ОПК.6, ОПК.7, ПК.6/ПК.

Кроме того, сформированность компетенций проверяется при проведении мероприятий текущего контроля, указанных в таблице раздела 1.

В 4 семестре обязательным этапом текущей аттестации является курсовая работа. Требования к выполнению курсовой работы, состав и правила оценки сформулированы в паспорте курсовой работы.

Общие правила выставления оценки по дисциплине определяются балльно-рейтинговой системой, приведенной в рабочей программе учебной дисциплины.

На основании приведенных далее критериев можно сделать общий вывод о сформированности компетенций ОПК.6, ОПК.7, ПК.6/ПК, за которые отвечает дисциплина, на разных уровнях.

Общая характеристика уровней освоения компетенций.

Ниже порогового. Уровень выполнения работ не отвечает большинству основных требований, теоретическое содержание курса освоено частично, пробелы могут носить существенный характер, необходимые практические навыки работы с освоенным материалом сформированы не достаточно, большинство предусмотренных программой обучения учебных заданий не выполнены или выполнены с существенными ошибками.

Пороговый. Уровень выполнения работ отвечает большинству основных требований, теоретическое 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 виды заданий выполнены с ошибками.

Базовый. Уровень выполнения работ отвечает всем основным требованиям, теоретическое 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 из выполненных заданий, возможно, содержат ошибки.

Продвинутый. Уровень выполнения работ отвечает всем требованиям, теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному.

Паспорт экзамена

по дисциплине «Технология программирования», 4 семестр

1. Методика оценки

Экзамен проводится в письменной форме, по билетам. Билет формируется по следующему правилу: первый вопрос выбирается из диапазона вопросов 1-20, второй вопрос из диапазона вопросов 21-40 (список вопросов приведен ниже). В ходе экзамена преподаватель вправе задавать студенту дополнительные вопросы из общего перечня (п. 4).

Форма экзаменационного билета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет АВТФ

Билет № _____

к экзамену по дисциплине «Технология программирования»

1. Иерархия классов. Объявление классов.
2. CAP теорема.
3. Создайте класс MyStack (методы добавления и удаления), который представляет собой структуру данных «стек», при помощи использования `std::vector`. Приведите пример использования созданного метода.

Утверждаю: зав. кафедрой _____ должность, ФИО
(подпись) (дата)

2. Критерии оценки

- Ответ на экзаменационный билет считается неудовлетворительным, если студент при ответе на вопросы не даёт определений основных понятий, не способен показать причинно-следственные связи явлений, при решении задачи допускает принципиальные ошибки, оценка составляет менее 10 баллов.
- Ответ на экзаменационный билет засчитывается на пороговом уровне, если студент при ответе на вопросы даёт определение основных понятий, может показать причинно-следственные связи явлений, при решении задачи допускает не принципиальные ошибки, например, вычислительные, оценка составляет от 10 до 20 баллов.
- Ответ на экзаменационный билет засчитывается на базовом уровне, если студент при ответе на вопросы формулирует основные понятия, законы, даёт характеристику процессов, явлений, проводит анализ причин, условий, может представить качественные характеристики процессов, не допускает ошибок при решении задачи, оценка составляет от 21 до 30 баллов.
- Ответ на экзаменационный билет засчитывается на продвинутом уровне, если студент при ответе на вопросы проводит сравнительный анализ подходов, проводит комплексный анализ, выявляет проблемы, предлагает механизмы решения, способен представить количественные

характеристики определённых процессов, приводит конкретные примеры из практики, не допускает ошибок и способен обосновать выбор метода решения задачи, оценка составляет от 31 до 40 баллов.

3. Шкала оценки

В общей оценке по дисциплине экзаменационные баллы учитываются в соответствии с правилами балльно-рейтинговой системы, приведёнными в рабочей программе дисциплины.

4. Вопросы к экзамену по дисциплине «Технология программирования»

1. Объекты. Классы.
2. Иерархия классов. Объявление классов.
3. Методы – элементы класса (описание, вызов). Перекрытие методов. Порядок вызова методов.
4. Основные принципы ООП: инкапсуляция, наследование (общая форма наследования), полиморфизм.
5. Windows API: Основные оконные функции.
6. Windows API: создание формы. Обработка кнопок.
7. Windows API: реализация собственных контрольных элементов на форме.
8. Основы работы с Visual Studio: создание проекта, компиляция.
9. Основы работы с Visual Studio: настройки сборщика (*linker*).
10. Отладка программы в Visual Studio: точки останова.
11. Отладка программы в Visual Studio: локализация ошибок.
12. Отладка программы в Visual Studio: трассировка ошибок.
13. Виртуальные методы. Таблица виртуальных методов.
14. Применение виртуальных функций, связанное с использованием указателей на объекты.
15. Чистые (*pure*) виртуальные функции. Абстрактные классы.
16. Понятие потоков и процессов.
17. Методы синхронизации потоков.
18. Передача данных между процессами.
19. Методы распараллеливания.
20. Выражение параллелизма с помощью библиотек поддержки многопоточности.
21. Явная и неявная синхронизация.
22. Проектирование многопоточного приложения: Декомпозиция данных.
23. Проектирование многопоточного приложения: Функциональная декомпозиция.
24. Верхний предел производительности. Закон Амдаля.
25. Потоки ввода/вывода.
26. Байтовые потоки ввода.
27. Байтовые потоки вывода.
28. Символьные потоки ввода/вывода.
29. Фильтры потоков.
33. Объектные потоки ввода/вывода.
34. Паттерны программирования: Одиночка.
35. Паттерны программирования: Декоратор, Фабричный метод.
36. Паттерн (принцип) программирования: Получение ресурса – есть инициализация (RAII).
37. Паттерны программирования: Мост.
38. Основы работы с сетью: функции для работы с сокетами.
39. Способы передачи данных в компьютерной сети.
40. Способы не блокировочной работы с сокетами.
41. Безопасность при передаче данных по сети.
42. CAP теорема.
43. Паттерн программирования: Наблюдатель.

Паспорт курсовой работы

по дисциплине «Технология программирования», 4 семестр

1. Методика оценки.

Задание на выполнение курсовой работы выдается в виде текстового описания функций программы, которые необходимо реализовать.

Пояснительная записка должна включать в указанной ниже последовательности: титульный лист, задание, содержание, введение, основную часть, заключение, список использованных источников, приложения. Этапы выполнения работы:

1. Выполнить декомпозицию задачи.
2. Определить какие классы должны быть реализованы для написания программы и взаимосвязи между классами.
3. Определить какие структуры данных и алгоритмы должны быть использованы для реализации программы.
4. Выполнить реализацию программы.
5. Реализовать unit-тесты и функциональные тесты в разработанной программе.
6. Сделать выводы о проделанной работе.

2. Критерии оценки.

- работа считается **не выполненной**, если выполнены не все части, отсутствует декомпозиция задачи, выбранные алгоритмы не обоснованы, функциональность выполнена с ошибками, оценка составляет менее 5 баллов.
- работа считается выполненной **на пороговом** уровне, если ее части выполнены формально: декомпозиция выполнена не полностью, выбранные алгоритмы не обоснованы и (или) имеют ошибки, не полностью реализована требуемая функциональность, оценка составляет от 5 до 10 баллов.
- работа считается выполненной **на базовом** уровне, если декомпозиция выполнена в полном объеме, выбранные алгоритмы обоснованы, но имеют незначительные ошибки, функциональность реализована в полном объеме, оценка составляет от 11 до 15 баллов.
- работа считается выполненной **на продвинутом** уровне, если декомпозиция выполнена в полном объеме, выбранные алгоритмы обоснованы, функциональность реализована в полном объеме, оценка составляет от 16 до 20 баллов.

3. Шкала оценки.

В общей оценке по дисциплине баллы за работы учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

4. Примерный перечень тем курсового проекта (работы).

- Разработка динамического объекта на базе xml структур.
- Разработка менеджера.
- Разработка части xpath библиотеки.

- Разработка программы для поиска файлов.

5. Перечень вопросов к защите курсового проекта (работы).

Примерные вопросы к защите:

1. При каких условиях вызывается деструктор класса?
2. Почему в C++ принято определять виртуальный деструктор?
3. Какие есть контейнеры в C++?
4. Что такое итераторы? Для чего они используются?
5. Какие функции в C++ позволяют работать с heap?