

Рабочая программа составлена на основании федерального государственного образовательного стандарта высшего образования по направлению (специальности): 09.03.04 Программная инженерия

ФГОС введен в действие приказом №229 от 12.03.2015 г. , дата утверждения: 01.04.2015 г.

Место дисциплины в структуре учебного плана: Блок 1, базовая

Рабочая программа разработана на основе компетентностной модели выпускника по направлению (специальности): 09.03.04 Программная инженерия

Рабочая программа обсуждена на заседании кафедры

ВТ, протокол заседания кафедры №6 от 20.06.2017

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

Программу разработал:

доцент, к.т.н. Васюткина И. А.

Заведующий кафедрой:

доцент, к.т.н. Якименко А. А.

Ответственный за образовательную программу:

доцент Романов Е. Л.

1. Внешние требования

Таблица 1.1

Компетенция ФГОС: ОПК.3 готовность применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов; в части следующих результатов обучения:	
33. знать принципы и методологии структурного и объектно-ориентированного подходов к разработке программного обеспечения	
у10. уметь распараллеливать алгоритмы и программы для решения прикладных задач	
Компетенция ФГОС: ПК.1 готовность применять основные методы и инструменты разработки программного обеспечения; в части следующих результатов обучения:	
312. знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	
33. знать принципы построения архитектуры программного обеспечения и виды архитектур программного обеспечения	
у2. уметь использовать существующие типовые решения и шаблоны проектирования программного обеспечения	
у4. уметь применять современные компиляторы, отладчики и оптимизаторы программного кода	
у7. уметь применять методы и приемы отладки программного кода	
у8. уметь выявлять ошибки в программном коде	
у9. уметь применять выбранные языки программирования для написания программного кода	
Компетенция ФГОС: ПК.11 владение особенностями эволюционной деятельности как с технической точки зрения, так и с точки зрения бизнеса (работа с унаследованными системами, возвратное проектирование, реинженеринг, миграция и рефакторинг; в части следующих результатов обучения:	
31. знать методы и средства рефакторинга и оптимизации программного кода	
Компетенция ФГОС: ПК.16 способность формализовать предметную область программного проекта и разработать спецификации для компонентов программного продукта; в части следующих результатов обучения:	
у1. уметь разрабатывать технические спецификации на программные компоненты и их взаимодействие	
Компетенция ФГОС: ПК.22 способность создавать программные интерфейсы; в части следующих результатов обучения:	
36. знать методы и средства проектирования программных интерфейсов	
Компетенция ФГОС: ПК.3 владение навыками использования различных технологий разработки программного обеспечения; в части следующих результатов обучения:	
у2. уметь использовать методы и приемы алгоритмизации поставленных задач	
Компетенция ФГОС: ПК.5 владение стандартами и моделями жизненного цикла; в части следующих результатов обучения:	
31. знать методологии разработки программного обеспечения	

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>	
ПК.1.312 знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	
1. знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	Лекции; Самостоятельная работа
ПК.1.33 знать принципы построения архитектуры программного обеспечения и виды архитектур программного обеспечения	
2. знать принципы построения архитектуры программного обеспечения и виды архитектур программного обеспечения	Лекции; Самостоятельная работа

ПК.1.y2 уметь использовать существующие типовые решения и шаблоны проектирования программного обеспечения	
3. уметь использовать существующие типовые решения и шаблоны проектирования программного обеспечения	Лекции; Самостоятельная работа
ПК.1.y4 уметь применять современные компиляторы, отладчики и оптимизаторы программного кода	
4. уметь применять современные компиляторы, отладчики и оптимизаторы программного кода	Лабораторные работы; Самостоятельная работа
ПК.1.y7 уметь применять методы и приемы отладки программного кода	
5. уметь применять методы и приемы отладки программного кода	Лабораторные работы; Самостоятельная работа
ПК.1.y8 уметь выявлять ошибки в программном коде	
6. уметь выявлять ошибки в программном коде	Лабораторные работы; Самостоятельная работа
ПК.1.y9 уметь применять выбранные языки программирования для написания программного кода	
7. уметь применять выбранные языки программирования для написания программного кода	Лабораторные работы; Самостоятельная работа
ОПК.3.z3 знать принципы и методологии структурного и объектно-ориентированного подходов к разработке программного обеспечения	
8. знать принципы и методологии структурного и объектно-ориентированного подходов к разработке программного обеспечения	Лекции; Самостоятельная работа
ОПК.3.y10 уметь распараллеливать алгоритмы и программы для решения прикладных задач	
9. уметь распараллеливать алгоритмы и программы для решения прикладных задач	Лекции; Лабораторные работы; Самостоятельная работа
ПК.3.y2 уметь использовать методы и приемы алгоритмизации поставленных задач	
10. уметь использовать методы и приемы алгоритмизации поставленных задач	Лабораторные работы; Самостоятельная работа
ПК.5.z1 знать методологии разработки программного обеспечения	
11. знать методологии разработки программного обеспечения	Лекции; Самостоятельная работа
ПК.11.z1 знать методы и средства рефакторинга и оптимизации программного кода	
12. знать методы и средства рефакторинга и оптимизации программного кода	Лабораторные работы; Самостоятельная работа
ПК.16.y1 уметь разрабатывать технические спецификации на программные компоненты и их взаимодействие	
13. уметь разрабатывать технические спецификации на программные компоненты и их взаимодействие	Лабораторные работы; Самостоятельная работа
ПК.22.z6 знать методы и средства проектирования программных интерфейсов	
14. знать методы и средства проектирования программных интерфейсов	Лекции; Самостоятельная работа

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: Технология Microsoft .NET. Язык программирования C#				
2. Технология Microsoft.Net. Общая среда выполнения CLR и система типов CTS. Библиотека классов .NET Framework. Сборки. JIT-компиляция.	0	2	1, 11	Слушает. Задает вопросы.

3. Особенности программирования на C#. Типы управляемых данных: структурные, ссылочные. Базовые типы: строки: классы String и StringBuilder, массивы: класс Array, перечисления класс Enum. Стандартный ввод/вывод. Форматирование.	0	2	1, 8	Слушает. Задает вопросы.
4. Классы и структуры. Статические конструкторы, константы. Свойства. Индексированные свойства. Особенности наследования в .Net. Интерфейсы. Стандартные интерфейсы: IComparable, ICloneable.	0	2	1, 8	Слушает. Задает вопросы.
5. Поточный ввод/вывод в файл. Сериализация и десериализация объектов. Интерфейс ISerializable.	0	2	1, 8	Слушает. Задает вопросы.
6. Исключения. Механизм обработки исключений в C#. Создание классов исключений.	0	2	1, 8	Слушает. Участвует в обсуждении отличий от C++.
7. Особенности наследования в C#. Интерфейсы. Множественное наследование интерфейсов. Коллизии имен в интерфейсах.	0	2	1, 8	Слушает. Участвует в обсуждении отличий от C++.
8. Делегаты. Примеры применения делегатов. Комбинированные делегаты.	0	2	1, 8	Слушает. Задает вопросы.
9. События. Модель делегирования событий.	0	3	1, 8	Слушает. Задает вопросы.
Дидактическая единица: Разработка приложений с GUI в .NET Framework				
10. Особенности разработки и структура Windows-приложения. Иерархия Windows Forms. Интегрированная среда визуальной разработки Windows-приложений Microsoft Visual Studio .Net.	0	2	14, 8	Слушает. Задает вопросы.
11. Построение приложений с использованием инструментальных средств разработки программ .NET SDK. Управляющие элементы. Диалоговые окна. Обмен данными с диалоговыми окнами. Модальные и немодальные формы. Диалоговые окна сохранения и загрузки файла.	0	4	14, 8	Слушает. Задает вопросы.
Дидактическая единица: Разработка многопоточных приложений. Клиент-серверные приложения.				

12. Класс Thread и его члены. Создание и запуск потоков. Приоритеты. Приостанов и возобновление работы потока. Фоновые потоки. Передача данных потоку.	0	3	2, 9	Слушает. Задает вопросы.
13. Разработка клиент-серверных приложений.	0	3	2, 9	Слушает. Задает вопросы.
Дидактическая единица: Разработка Web-приложений. Технологии ASP .NET, ADO.NET, AJAX				
14. Разработка клиента web-приложения. HTML. CSS. JavaScript.	0	3	11, 2, 3	Слушает. Задает вопросы.
15. Разработка серверной части web-приложения. Технология ASP.NET.	0	2	11, 2, 3	Слушает. Задает вопросы.
16. Проверка и запись данных в базу данных. Технология ADO.NET. Динамическое обновление страниц ASP.NET AJAX.	0	2	11, 2, 3	Слушает. Задает вопросы.

Таблица 3.2

Темы лабораторных работ	Активные формы, час.	Часы	Ссылки на результаты обучения	Учебная деятельность
Семестр: 4				
Дидактическая единица: Технология Microsoft .NET. Язык программирования C#				
1. Встроенные типы данных в C#. Массивы. Строки. Регулярные выражения.	2	4	10, 12, 13, 4, 5, 6, 7, 9	Выполняет учебное индивидуальное задание.
2. Типы данных, определяемые пользователем. Наследование. Обработка исключений в C#.	2	4	10, 12, 13, 4, 5, 6, 7	Выполняет учебное индивидуальное задание.
Дидактическая единица: Разработка приложений с GUI в .NET Framework				
3. Разработка GUI. Создание SDI-приложений. Обработка событий	2	4	10, 12, 13, 4, 5, 6, 7, 9	Выполняет учебное индивидуальное задание.
4. Создание MDI-приложений. Сериализация объектов. Стандартные диалоговые окна	2	4	10, 12, 13, 4, 5, 6, 7	Выполняет учебное индивидуальное задание.
Дидактическая единица: Разработка многопоточных приложений. Клиент-серверные приложения.				
5. Графика. Анимация. Разработка многопоточных приложений.	2	4	10, 12, 13, 4, 5, 6, 7	Выполняет учебное индивидуальное задание.
6. Разработка "клиент-серверных" приложений.	2	4	10, 12, 13, 4, 5, 6, 7	Выполняет учебное индивидуальное задание.
Дидактическая единица: Разработка Web-приложений. Технологии ASP .NET, ADO.NET, AJAX				
7. Разработка web-приложений в ASP.NET.	2	4	10, 12, 13, 4, 5, 6, 7	Выполняет учебное индивидуальное задание.
8. Верификация данных. Хранение и передача данных. Технология ADO.NET. Навигация по web-приложению.	2	4	10, 12, 13, 4, 5, 6, 7	Выполняет учебное индивидуальное задание.

9. Динамическое обновление web-страниц. Технология AJAX. web-службы.	2	4	10, 12, 13, 4, 5, 6, 7	Выполняет учебное индивидуальное задание.
--	---	---	------------------------	---

4. Самостоятельная работа обучающегося

№	Виды самостоятельной работы	Ссылки на результаты обучения	Часы на выполнение	Часы на консультации
Семестр: 4				
1	Курсовая работа	10, 13, 3, 4, 5, 6, 7, 9	30	2
Выполнение индивидуального задания на курсовую работу, подробная информация приведена в приложении №3 : Технология разработки объектно-ориентированных программ на языке C# в среде Visual Studio. Net : методические указания к лабораторным работам по дисциплине "Технология программирования и разработка программного обеспечения" для 2 курса АВТФ (специальность 230101 "Вычислительные машины, комплексы, системы и сети" по направлению 230100 "Информатика и вычислительная техника" / Новосиб. гос. техн. ун-т ; [сост. И. А. Васюткина]. - Новосибирск, 2010. - 81, [2] с. : ил., табл.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000145712				
2	Подготовка к занятиям	1, 11, 12, 14, 2, 8	20	6
Изучение теоретического материала лекций и выполнение заданий лабораторных работ.: Технология разработки объектно-ориентированных программ на языке C# в среде Visual Studio. Net : методические указания к лабораторным работам по дисциплине "Технология программирования и разработка программного обеспечения" для 2 курса АВТФ (специальность 230101 "Вычислительные машины, комплексы, системы и сети" по направлению 230100 "Информатика и вычислительная техника" / Новосиб. гос. техн. ун-т ; [сост. И. А. Васюткина]. - Новосибирск, 2010. - 81, [2] с. : ил., табл.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000145712				
3	Подготовка к аттестации	1, 11, 12, 8	10	2
Подготовка по экзаменационным вопросам, подробная информация приведена в приложении №2 : Технология разработки объектно-ориентированных программ на языке C# в среде Visual Studio. Net : методические указания к лабораторным работам по дисциплине "Технология программирования и разработка программного обеспечения" для 2 курса АВТФ (специальность 230101 "Вычислительные машины, комплексы, системы и сети" по направлению 230100 "Информатика и вычислительная техника" / Новосиб. гос. техн. ун-т ; [сост. И. А. Васюткина]. - Новосибирск, 2010. - 81, [2] с. : ил., табл.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000145712 Разработка приложений на C# с использованием СУБД PostgreSQL : учебное пособие / [И. А. Васюткина и др.] ; Новосиб. гос. техн. ун-т. - Новосибирск, 2015. - 141, [1] с. : ил., табл.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000220068				

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	Портал НГТУ
Консультирование	e-mail; Социальные сети
Контроль	e-mail
Размещение учебных материалов	Портал НГТУ; ЭБС

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.

Подробная информация о БРС приводится в Приложении №1 к рабочей программе.

Таблица 6.1

Оцениваемые виды деятельности обучающихся	Мин. балл	Максимальный балл
Семестр: 4		
<i>Лабораторная №1:</i> Выполнение и защита	3	6
<i>Лабораторная №2:</i> Выполнение и защита	3	6
<i>Лабораторная №3:</i> Выполнение и защита	3	6
<i>Лабораторная №4:</i> Выполнение и защита	4	7
<i>Лабораторная №5:</i> Выполнение и защита	4	7
<i>Лабораторная №6:</i> Выполнение и защита	4	7
<i>Лабораторная №7:</i> Выполнение и защита	4	7
<i>Лабораторная №8:</i> Выполнение и защита	4	7
<i>Лабораторная №9:</i> Выполнение и защита	4	7
<i>Курсовая работа:</i> Итого	0	100
Контролирующие материалы приводятся в "Технология разработки объектно-ориентированных программ на языке C# в среде Visual Studio. Net : методические указания к лабораторным работам по дисциплине "Технология программирования и разработка программного обеспечения" для 2 курса АВТФ (специальность 230101 "Вычислительные машины, комплексы, системы и сети" по направлению 230100 "Информатика и вычислительная техника" / Новосиб. гос. техн. ун-т ; [сост. И. А. Васюткина]. - Новосибирск, 2010. - 81, [2] с. : ил., табл.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000145712 "		
<i>Экзамен:</i>	20	40
Контролирующие материалы приводятся в "Технология разработки объектно-ориентированных программ на языке C# в среде Visual Studio. Net : методические указания к лабораторным работам по дисциплине "Технология программирования и разработка программного обеспечения" для 2 курса АВТФ (специальность 230101 "Вычислительные машины, комплексы, системы и сети" по направлению 230100 "Информатика и вычислительная техника" / Новосиб. гос. техн. ун-т ; [сост. И. А. Васюткина]. - Новосибирск, 2010. - 81, [2] с. : ил., табл.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000145712 "		

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.3	з3. знать принципы и методологии структурного и объектно-ориентированного подходов к разработке программного обеспечения		+
	у10. уметь распараллеливать алгоритмы и программы для решения прикладных задач	+	
ПК.1	з12. знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования		+
	з3. знать принципы построения архитектуры программного обеспечения и виды архитектур программного обеспечения		+
	у2. уметь использовать существующие типовые решения и шаблоны проектирования программного обеспечения	+	
	у4. уметь применять современные компиляторы, отладчики и оптимизаторы программного кода	+	

	у7. уметь применять методы и приемы отладки программного кода	+	
	у8. уметь выявлять ошибки в программном коде	+	
	у9. уметь применять выбранные языки программирования для написания программного кода	+	
ПК.11	з1. знать методы и средства рефакторинга и оптимизации программного кода	+	
ПК.16	у1. уметь разрабатывать технические спецификации на программные компоненты и их взаимодействие	+	
ПК.22	з6. знать методы и средства проектирования программных интерфейсов		+
ПК.3	у2. уметь использовать методы и приемы алгоритмизации поставленных задач	+	
ПК.5	з1. знать методологии разработки программного обеспечения		+

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

7. Литература

Основная литература

1. Рихтер Д. CLR via C#. Программирование на платформе Microsoft .NET Framework 2.0 на языке C#. Мастер-класс : пер. с англ. / Джеффри Рихтер. - М., 2007. - 636 с. : ил.
2. Васюткина И. А. Разработка клиент-серверных приложений на языке C# : учебное пособие / И. А. Васюткина ; Новосиб. гос. техн. ун-т. - Новосибирск, 2016. - 110, [1] с. : ил., табл. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000230286
3. Павловская Т. А. C#. Программирование на языке высокого уровня : [учебник по направлению "Информатика и вычислительная техника"] / Т. А. Павловская. - СПб. [и др.], 2010. - 432 с. : ил.
4. Фаронов В. В. Программирование на языке C# : учебный курс / Валерий Фаронов. - СПб. [и др.], 2007. - 239 с. : ил.
5. Агуров П. В. ASP.NET. Сборник рецептов / Павел Агуров. - СПб., 2010. - 507 с. : ил., табл. - На обл. в подзаг.: архитектура веб-приложений, веб-формы и компоненты ASP.NET, работа с базами данных, отладка и обработка ошибок, защита приложений ASP.NET, создание отчетов в MS Excel.
6. Марченко А. Л. Основы программирования на C# 2.0 : учебное пособие / А. Л. Марченко. - М., 2007. - 551 с. : ил.
7. Джонсон Г. Разработка клиентских веб-приложений на платформе Microsoft .NET Framework : экзамен 70-528 MCTS / Гленн Джонсон, Тони Нортроп. - М. [и др.], 2007. - XX, 745 с. : ил. + 1 CD-ROM.

Дополнительная литература

1. Камаев В. А. Технологии программирования : [учебник для вузов по направлению "Информатика и вычислительная техника"] / В. А. Камаев, В. В. Костерин. - М., 2006. - 453, [1] с. : ил., табл.

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 "Znaniy.com" : <http://znaniy.com/>
5. :

8. Методическое и программное обеспечение

8.1 Методическое обеспечение

1. Разработка приложений на С# с использованием СУБД PostgreSQL : учебное пособие / [И. А. Васюткина и др.] ; Новосиб. гос. техн. ун-т. - Новосибирск, 2015. - 141, [1] с. : ил., табл.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000220068
2. Технология разработки объектно-ориентированных программ на языке С# в среде Visual Studio. Net : методические указания к лабораторным работам по дисциплине "Технология программирования и разработка программного обеспечения" для 2 курса АВТФ (специальность 230101 "Вычислительные машины, комплексы, системы и сети" по направлению 230100 "Информатика и вычислительная техника" / Новосиб. гос. техн. ун-т ; [сост. И. А. Васюткина]. - Новосибирск, 2010. - 81, [2] с. : ил., табл.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000145712

8.2 Специализированное программное обеспечение

- 1 Microsoft Windows
- 2 Visual Studio

9. Материально-техническое обеспечение

Презентационное оборудование

№	Наименование	Назначение
1	Презентационное оборудование (мультимедиа-проектор, экран, компьютер для управления)	При чтении лекций применяются презентации

Компьютерный класс

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

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

Кафедра вычислительной техники

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

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

УЧЕБНОЙ ДИСЦИПЛИНЫ

Технология программирования

Образовательная программа: 09.03.04 Программная инженерия, профиль: Технологии разработки программного обеспечения

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.3 готовность применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	з3. знать принципы и методологии структурного и объектно-ориентированного подходов к разработке программного обеспечения	Особенности наследования в С#. Интерфейсы. Множественное наследование интерфейсов. Коллизии имен в интерфейсах.		Экзамен, вопросы 13-14.
ОПК.3	у10. уметь распараллеливать алгоритмы и программы для решения прикладных задач	Класс Thread и его члены. Создание и запуск потоков. Приоритеты. Приостанов и возобновление работы потока. Фоновые потоки. Передача данных потоку.	Курсовая работа, разделы 5	
ПК.1/ПТ готовность применять основные методы и инструменты разработки программного обеспечения	з3. знать принципы построения архитектуры программного обеспечения и виды архитектур программного обеспечения	Разработка клиента web-приложения. HTML. CSS. JavaScript. Разработка клиент-серверных приложений.		Экзамен, вопросы 26, 28
ПК.1/ПТ	з12. знать синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования	Делегаты. Примеры применения делегатов. Комбинированные делегаты. Исключения. Механизм обработки исключений в С#. Создание классов исключений. Классы и структуры. Статические конструкторы, константы. Свойства. Индексированные свойства. Особенности наследования в .Net. Интерфейсы. Стандартные интерфейсы: IComparable, ICloneable. Особенности наследования в С#. Интерфейсы. Множественное наследование интерфейсов. Коллизии имен в интерфейсах. Особенности программирования на С#. Типы управляемых данных: структурные, ссылочные. Базовые типы: строки: классы String и StringBuilder, массивы: класс Array, перечисления класс Enum. Стандартный ввод/вывод.		Экзамен, вопросы 2-24, 27, 29-32

		Форматирование. Поточковый ввод/вывод в файл. Сериализация и десериализация объектов. Интерфейс ISerializable. События. Модель делегирования событий.		
ПК.1/ПТ	у2. уметь использовать существующие типовые решения и шаблоны проектирования программного обеспечения	Проверка и запись данных в базу данных. Технология ADO.NET. Динамическое обновление страниц ASP.NET AJAX. Разработка клиента web-приложения. HTML. CSS. JavaScript. Разработка серверной части web-приложения. Технология ASP.NET.	Курсовая работа, раздел 2	
ПК.1/ПТ	у4. уметь применять современные компиляторы, отладчики и оптимизаторы программного кода	Верификация данных. Хранение и передача данных. Технология ADO.NET. Навигация по web-приложению. Встроенные типы данных в C#. Массивы. Строки. Регулярные выражения. Графика. Анимация. Разработка м ногопоточковых приложений. Динамическое обновление web-страниц. Технология AJAX. web-службы. Разработка GUI. Создание SDI-приложений. Обработка событий Разработка web-приложений в ASP.NET. Разработка "клиент-серверных" приложений. Создание MDI-приложений. Сериализация объектов. Стандартные диалоговые окна Типы данных, определяемые пользователем. Наследование. Обработка исключений в C#.	Курсовая работа, раздел 5	
ПК.1/ПТ	у7. уметь применять методы и приемы отладки программного кода	Верификация данных. Хранение и передача данных. Технология ADO.NET. Навигация по web-приложению. Встроенные типы данных в C#. Массивы. Строки. Регулярные выражения. Графика. Анимация. Разработка м ногопоточковых приложений. Динамическое обновление web-страниц. Технология AJAX. web-службы. Разработка GUI. Создание SDI-приложений. Обработка событий Разработка web-приложений в ASP.NET. Разработка "клиент-серверных" приложений. Создание MDI-приложений. Сериализация объектов. Стандартные диалоговые окна Типы данных, определяемые пользователем. Наследование. Обработка исключений в C#.	Курсовая работа, раздел 5	

ПК.1/ПТ	у8. уметь выявлять ошибки в программном коде	Верификация данных. Хранение и передача данных. Технология ADO.NET. Навигация по web-приложению. Встроенные типы данных в C#. Массивы. Строки. Регулярные выражения. Графика. Анимация. Разработка многопоточковых приложений. Динамическое обновление web-страниц. Технология AJAX. web-службы. Разработка GUI. Создание SDI-приложений. Обработка событий Разработка web-приложений в ASP.NET. Разработка "клиент-серверных" приложений. Создание MDI-приложений. Сериализация объектов. Стандартные диалоговые окна Типы данных, определяемые пользователем. Наследование. Обработка исключений в C#.	Курсовая работа, раздел 5	
ПК.1/ПТ	у9. уметь применять выбранные языки программирования для написания программного кода	Разработка web-приложений в ASP.NET. Создание MDI-приложений. Сериализация объектов. Стандартные диалоговые окна	Курсовая работа, раздел 5	
ПК.11/СЭ владение особенностями эволюционной деятельности как с технической точки зрения, так и с точки зрения бизнеса (работа с унаследованными системами, возвратное проектирование, реинженеринг, миграция и рефакторинг	з1. знать методы и средства рефакторинга и оптимизации программного кода	Верификация данных. Хранение и передача данных. Технология ADO.NET. Навигация по web-приложению. Встроенные типы данных в C#. Массивы. Строки. Регулярные выражения. Графика. Анимация. Разработка многопоточковых приложений. Динамическое обновление web-страниц. Технология AJAX. web-службы. Разработка GUI. Создание SDI-приложений. Обработка событий Разработка web-приложений в ASP.NET. Разработка "клиент-серверных" приложений. Создание MDI-приложений. Сериализация объектов. Стандартные диалоговые окна Типы данных, определяемые пользователем. Наследование. Обработка исключений в C#.	Курсовая работа, разделы 2-5	
ПК.16/А способность формализовать предметную область программного проекта и разработать спецификации для компонентов программного	у1. уметь разрабатывать технические спецификации на программные компоненты и их взаимодействие	Разработка web-приложений в ASP.NET. Разработка "клиент-серверных" приложений. Создание MDI-приложений. Сериализация объектов. Стандартные диалоговые окна	Курсовая работа, разделы 2	

продукта				
ПК.22/П способность создавать программные интерфейсы	з6. знать методы и средства проектирования программных интерфейсов	Особенности разработки и структура Windows-приложения. Иерархия Windows Forms. Интегрированная среда визуальной разработки Windows-приложений Microsoft Visual Studio .Net. Построение приложений с использованием инструментальных средств разработки программ .NET SDK. Управляющие элементы. Диалоговые окна. Обмен данными с диалоговыми окнами. Модальные и немодальные формы. Диалоговые окна сохранения и загрузки файла.		Экзамен, вопрос 33
ПК.3/ПТ владение навыками использования различных технологий разработки программного обеспечения	у2. уметь использовать методы и приемы алгоритмизации поставленных задач	Разработка GUI. Создание SDI-приложений. Обработка событий Разработка web-приложений в ASP.NET. Разработка "клиент-серверных" приложений. Создание MDI-приложений. Сериализация объектов. Стандартные диалоговые окна	Курсовая работа, разделы 2, 4	
ПК.5/ПТ владение стандартами и моделями жизненного цикла	з1. знать методологии разработки программного обеспечения	Разработка клиента web-приложения. HTML. CSS. JavaScript. Технология Microsoft.Net. Общая среда выполнения CLR и система типов CTS. Библиотека классов .NET Framework. Сборки. JIT-компиляция.		Экзамен, вопросы 1, 25

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.3, ПК.1/ПТ, ПК.11/СЭ, ПК.16/А, ПК.22/П, ПК.3/ПТ, ПК.5/ПТ.

Экзамен проводится в устной форме по билетам. Билет содержит два теоретических вопроса: вопросы выбираются из диапазона вопросов (список вопросов приведен в паспорте экзамена).

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 дисциплины.

На основании приведенных далее критериев можно сделать общий вывод о сформированности компетенций ОПК.3, ПК.1/ПТ, ПК.11/СЭ, ПК.16/А, ПК.22/П, ПК.3/ПТ, ПК.5/ПТ, 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.

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»
Кафедра автоматизированных систем управления
Кафедра автоматики
Кафедра вычислительной техники

Паспорт экзамена

по дисциплине «Технология программирования», 4 семестр

1. Методика оценки

Экзамен проводится в устной форме, по билетам. Билет содержит 2 вопроса. Билет формируется по следующему правилу: вопросы выбираются произвольно из диапазона вопросов 1 - 33. 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. Классификация типов данных в C#. Типы данных в FCL и C#. Создание переменных. Константы. Преобразования встроенных типов.
2. Архитектура клиент-сервер. Создание сокетного соединения. Передача данных по сети.

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

2. Критерии оценки

- Ответ на экзаменационный билет считается **неудовлетворительным**, если студент при ответе на вопросы не дает определений основных понятий, имеются существенные недочеты, ответы на дополнительные вопросы не полные и носят обрывочный характер, оценка составляет 0 - 19 *баллов*.
- Ответ на экзаменационный билет засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает определение основных понятий, ошибки в ответе не носят существенного характера, ответы на дополнительные вопросы достаточно полные, оценка составляет 20 - 29 *баллов*.
- Ответ на экзаменационный билет засчитывается на **базовом** уровне, если студент при ответе на вопросы формулирует основные понятия, может показать и объяснить

применение методов и средств языка при решении практических задач, легко ориентируется в изученном материале, оценка составляет 30 - 35 *баллов*.

- Ответ на экзаменационный билет засчитывается на **продвинутом** уровне, если студент при ответе на вопросы проводит сравнительный анализ подходов, способен анализировать и принимать эффективные решения поставленных задач, приводит конкретные примеры из практики, оценка составляет 36 - 40 *баллов*.
-

3. Шкала оценки

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

Итоговая оценка по экзамену выставляется по 100-балльной шкале (баллы, полученные за лабораторные работы и на экзамене), по буквенной шкале ECTS и в традиционной форме (в соответствии с действующим Положением о балльно-рейтинговой системе оценки достижений студентов НГТУ).

4. Вопросы к экзамену по дисциплине «Технология программирования»

1. Технология Microsoft.Net. Общая среда выполнения CLR и система типов CTS. Библиотека классов .NET Framework. Сборки. JIT-компиляция.
2. Классификация типов данных в C#. Типы данных в FCL и C#. Создание переменных. Константы. Преобразования встроенных типов.
3. Операции. Операторы языка C#. Отличия от операторов языка C++. Перегрузка операций в классах.
4. Массивы в C#. Одномерные, двумерные, ступенчатые. Класс Array и его методы.
5. Строки в C#. Классы String, StringBuilder. Операции со строками. Основные статические и динамические методы классов.
6. Регулярные выражения. Синтаксис описания регулярного выражения. Класс Regex и его методы. Свойства класса Match.
7. Структуры в C#. Определение конструктора в структурах. Создание переменных.
8. Процедуры и функции. Список формальных параметров. Функции с побочным эффектом.
9. Понятие объектно-ориентированное программирование. Три основных принципа языков объективно-ориентированного программирования.
10. Что такое класс? Члены класса. Модификаторы доступа к классам и членам класса. Создание экземпляра класса. Класс Object.
11. Методы-свойства и индексы класса. Назначение и описание.
12. Статические поля и методы класса. Абстрактные методы и классы.
13. Наследование в C#. Конструкторы. Изменение методов родителя в классе наследника.
14. Интерфейсы. Наследование в интерфейсах. Реализация методов интерфейсов в классах. Коллизия имен в интерфейсах.
15. Исключения. Генерация исключения. Обработка исключений в C#. Создание классов исключений.
16. Что такое делегат? Создание делегата. Функции высших порядков, обратного вызова. Комбинирование делегатов.
17. События. Что такое обработчик события? Как происходит подключение к прослушиванию события?
18. Что такое поток вывода/вывода? Виды потоков. Иерархии классов потоков.

19. Сериализация/десериализация. Для чего она применяется? Форматы сериализации? Как задать сериализацию объектов класса? Как управлять сериализацией?
 20. Что такое буферизация потоков? Для чего используется буферизация.
 21. Generic-классы и методы. Наложение ограничений на параметризованный тип.
 22. Многопоточность. Создание потоков, запуск, останов. Приоритеты потоков.
- Приостанов и возобновление потоков.
23. Синхронизация потоков. Монитор. Управление монитором.
 24. Архитектура клиент-сервер. Создание сокетного соединения. Передача данных по сети.
 25. Понятие о web-приложении. Архитектура и логические уровни web-приложения.
- Технология ASP.NET. Жизненный цикл web-страницы.
26. «Тонкие» клиенты. Языковые средства реализации «тонких» клиентов.
 27. Серверные страницы ASP.NET. Состояние вида ViewState. Передача данных между страницами.
 28. Валидация данных HTML и ASP- страниц.
 29. Технология AJAX. Применение в программировании.
 30. Технология ADO.NET. Понятие поставщик данных. Способы работы с данными.
 31. Подключенный уровень. Способы подключения к базе данных. Выполнение SQL-команд.
 32. Автономный уровень. Получение данных базы данных. Обновление данных в базе данных.
 33. Построение GUI. Управляющие элементы. Диалоговые окна. Обмен данными с диалоговыми окнами. Модальные и немодальные формы. Диалоговые окна сохранения и загрузки файла.

Паспорт курсовой работы

по дисциплине «Технология программирования», 4 семестр

1. Методика оценки.

Курсовая работа выполняется по индивидуальным заданиям и оценивается по 100 балльной шкале. В рамках работы студент должен разработать программный продукт, протестировать его и оформить отчет по работе.

Разделы курсовой работы:

1. Развернутая постановка задачи.
2. Проектирование объектов хранения.
3. Разработка структуры базы данных.
4. Проектирование интерфейсов пользователя.
5. Реализация приложения.
6. Оформление отчета.
7. Защита.

2. Критерии оценки.

- работа считается **не выполненной**, если студент не выполнил задание полностью или выполнил с очень существенными ошибками, оформил пояснительную записку, но имеются многочисленные замечания, не может объяснить используемые алгоритмы и технологии, оценка составляет 0 - 49 баллов.
- работа считается выполненной **на пороговом** уровне, если студент выполнил задание, но есть небольшие недочеты, оформил пояснительную записку, но имеются несущественные замечания, грамотно объясняет используемые алгоритмы и технологии, оценка составляет 50 - 72 баллов.
- работа считается выполненной **на базовом** уровне, если студент полностью выполнил задание, оформил пояснительную записку согласно требованиям, грамотно и полно объясняет используемые алгоритмы и технологии, оценка составляет 73 - 86 баллов.
- работа считается выполненной **на продвинутом** уровне, если студент полностью выполнил задание, применил оригинальные алгоритмы и программные подходы, оформил пояснительную записку согласно требованиям, грамотно и полно объясняет используемые алгоритмы и технологии, оценка составляет 87 - 100 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 курсовой работы.

1. Реализация графических часов.

2. Игровая программа «Тетрис».
3. Игра «Сапер». (Балл -3)
4. Сетевая игра "Шашки" .
5. Разработка моделирующей программы «Аквариум»
6. Моделирующая игра «Жизнь»
7. Сетевая игра «Морской бой»
8. Разработать программу «Фотоальбом».
9. Разработать многопоточную программу, моделирующую работу товарного склада.
10. Разработать программу построения Линейных фракталов.
11. Разработать моделирующую программу «Волчий остров
12. Разработать игровую программу - Две лисы и 20 кур.
13. Сетевая программа «On-line конференция».
14. Разработка программы «Чат».
15. Разработка клиент-серверной программы «Страница преподавателя».
16. Разработать сетевую программу «Контроль успеваемости студентов».
17. Реализовать клиент-серверное приложение «Электронная система учета работы пункта проката видео».
18. АРМ кассира кинотеатра.
19. Разработка игровой сетевой программы «Крестики-нолики».
20. Разработка «Интернет-магазина» (можно на 2-х)
 - Клиентская часть:
 - Административная часть:

Требования по оформлению записки.

1. Техническое задание.
 - Введение (описание предметной области);
 - Назначение разработки, ее актуальность;
 - Требования к программе;
 - Постановка задачи (формулирование задач, подлежащих решению).
2. Проектирование (разработка диаграмм UML: диаграммы взаимодействия, диаграммы последовательности действий, диаграммы классов).
3. Структурное описание разработки.
 - Описание структуры приложения, его модулей.
 - Описание классов: спецификации данных, методов.
4. Функциональное описание разработки
 - Описание алгоритмов и методов решения.
5. Описание пользовательского интерфейса.
6. Тестирование программы (при необходимости).
7. Руководство пользователя.
8. Руководство программиста. (Установка программы, необходимые программные и аппаратные средства).
9. Заключение
10. Список литературы.
11. Приложение (текст программы с комментариями).

5. Перечень вопросов к защите курсовой работы.

Вопросы задаются по теме курсовой работы, примененным алгоритмам и технологиям.