Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новгородский государственный университет имени Ярослава Мудрого»
Институт информационных и электронных систем
Кафедра прикладной математики и информатики

УТВЕРЖДАЮ
Директор ИЭИС
_____ С.И. Эминов
_____ 2017г.

СОВРЕМЕННЫЕ ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Учебный модуль по направлению подготовки
01.03.02 «Прикладная математика и информатика»
Рабочая программа

СОГЛАСОВАНО

Начальник учебного отдела
_____ О.Б. Широколобова
«11» 04 2017

Разработал

Доцент кафедры ПМИ НовГУ
_____ Т.В. Жгун
«10» февраля 2017 г.

Принято на заседании кафедры КПМИ
Протокол № 7 от 03 2017 г.
Заведующий кафедрой
_____ А.В. Колногоров

1 ЦЕЛИ И ЗАДАЧИ УЧЕБНОГО МОДУЛЯ

Целью учебного модуля (УМ) является формирование компетентности студентов в производственно-технологической деятельности, включающих организацию проектной деятельности, проектирование и разработку программных решений прикладного программного обеспечения.

Основными задачами УМ являются:

- формирование знаний о современных методологиях разработки программного обеспечения;
- развитие умений и навыков проектирования программных продуктов;
- формирование знаний о методиках планирования и оценки рисков;
- формирование навыков командной работы над проектом;
- знакомство с инженерными практиками повышения качества программного продукта;
- расширение знаний о современных языках и парадигмах программирования;
- развитие умений и навыков программирования с использованием современных интегрированных сред разработки.

2 МЕСТО УЧЕБНОГО МОДУЛЯ В СТРУКТУРЕ ОП НАПРАВЛЕНИЯ ПОДГОТОВКИ 01.03.02 «ПРИКЛАДНАЯ МАТЕМАТИКА И ИНФОРМАТИКА»

УМ «СОВРЕМЕННЫЕ ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ» входит в учебный план подготовки бакалавров по направлению 01.03.02- Прикладная математики и информатика, ввариативная часть блока Б1 и читается в 5семестре.

Для освоения модуля студенты используют знания и умения, формируемые в процессе изучения модулей «Алгоритмические языки», «Системное и прикладное программное обеспечение» и «Объектно-ориентированное программирование».

Настоящий модуль включает в себя основные понятия в области разработки программного обеспечения: методологии разработки ПО, подходы к проектированию и разработке программного обеспечения, техники и инструменты организации процесса разработки и контроля за качеством программных продуктов.

Знания и навыки, полученные при изучении модуля, используются при организации и выполнении проектов по разработке программного обеспечения в рамках профессиональной деятельности, а также при выполнении выпускной квалификационной работы.

3 ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ МОДУЛЯ

Процесс изучения УМ направлен на формирование компетенций:

- способность к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям (ОПК-3);
- способность к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения (ПК-7).

В результате освоения УМ студент должен знать, уметь и владеть:

Код компетенции	Уровень освое-ния компе-тенции	Знать	Уметь	Владеть
ОПК-3	базовый	основы разработки алгоритмических и программных решений в области системного и прикладного программирования.	разрабатывать алгоритмические и программные решения в области системного и прикладного программирования.	навыками: разработки алгоритмических и программных решений в области системного и прикладного программирования.
ПК-7	базовый	основы разработки и применения алгоритмических и программных решений в области системного и прикладного программирования.	Разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программирования.	навыками: разработки и применения алгоритмических и программных решений в области системного и прикладного программирования.

4 СТРУКТУРА И СОДЕРЖАНИЕ МОДУЛЯ

4.1 Трудоемкость учебного модуля

В структуре УМ выделены следующие учебные элементы модуля (УЭМ) в качестве самостоятельных разделов:

- Технологии проектирования программных продуктов
- Управление проектами**.**
- Кодирование и тестирование программных продуктов

Учебная работа (УР)	Всего	Распределение по семестрам	Коды формируемых компетенций
		5 сем.	
Трудоемкость модуля в зачетных единицах (ЗЕ)	6	6	
Распределение трудоемкости по видам УР в акад. часах	216	216	
- лекции	36	36	ОПК-3, ПК-7
- практические занятия	36	36	
- лабораторные работы	36	36	
- аудиторная СРС	18	18	
- внеаудиторная СРС	108	108	
Аттестация:- экзамен	36	36	

4.2 Содержание и структура разделов учебного модуля

УЭМ 1 Технологии проектирования программных продуктов

1.1 Введение. Предмет модуля , её содержание и связь с другими дисциплинами учебного плана. Цели и задачи модуля

1.2 . Жизненный цикл ПО. Прототипирование. Опыт взаимодействия. Оценка дизайна

1.3. Сбор и анализ требований. Функциональные спецификации

1.4 Принципы декомпозиции задач. Объектно-ориентированный анализ. Шаблоны проектирования

УЭМ 2. Управление проектами

2.1 Организация жизненного цикла. Планирование задач. Классический подход. 2.2 Гибкие методологии Управление рисками. Подходы к управлению рисками. Дисциплина управления рисками MSF

УЭМ 3 . Кодирование и тестирование программных продуктов

3.1 Качество кода: ошибки и дефекты. Защитное программирование и его альтернативы. Протоколирование. Тестирование и отладка ПО. Системы отслеживания ошибок

3.2 Качество кода: сопровождаемость. Соглашения по стилям кодирования. Комментарии. Инспекции кода. Принципы минимализма. Технический долг и рефакторинг.

3.3 Качество кода: оптимизация программ. Профилирование. Техники оптимизации

34 . Системы контроля версий. Централизованные и распределённые системы. Организация командной работы.

3.5 Парадигмы программирования. Современные языки программирования

3.6 Кросс-платформенность. Многозадачность

Календарный план, наименование разделов учебного модуля с указанием трудоемкости по видам учебной работы представлены в технологической карте учебного модуля (приложение Б).

Лабораторный практикум

Лабораторный практикум состоит из лабораторных работ, наименование которых приведено в таблице

№ раздела УМ	Наименование лабораторных работ	Трудоемкость, ак.час
УМ1	Разработка программного прототипа.	10
УМ1, УМ2	Разработка технического задания на программный продукт.	10
УМ1, УМ2, УМ3	Разработка функциональных спецификаций. Планирование процесса разработки.	
УМ2, УМ3	Разработка и тестирование программного продукта. Подведение итогов планирования.	

Темы практических работ:

ПР-1 — Знакомство с инструментами прототипирования. Создание прототипа пользовательского интерфейса.

ПР-2 — Сбор требований к программному продукту: разбор подходов и примеров.

ПР-3 — Анализ требований к программному продукту: ознакомление с процессом анализа, разбор примеров.

ПР-4 — Декомпозиция задач. Практика объектно-ориентированного проектирования.

ПР-5 — Оценка трудоёмкости задач. Анализ проектных рисков

ПР-6 — Повышение качества программного кода. Обработка ошибок, комментирование, рефакторинг, оптимизация.

ПР-7 — Работа с распределённой системой контроля версий.

Темы СРС :

- — Изучение программных инструментов прототипирования приложений. Отработка навыков создания прототипов.
- — Сбор и оформление требований к программным продуктам.
- — Проектирование программной архитектуры: изучение инструментов.
- — Декомпозиция задач: упражнения и изучение примеров в литературе.
- — Оценка трудоёмкости задач, анализ рисков: перекрёстный контроль.
- — Повышение качества программного кода. Изучение языковых средств, инструментов рефакторинга, отладки и профилирования.
- — Практика командной работы над проектом с использованием системы контроля версий.
- — Знакомство с современными языками и платформами программирования: сферы применения, концепции, подходы, особенности.

Темы самостоятельных работ:

СР-1 оформление требований к программным продуктам

СР-2 Оценка трудоёмкости задач

4.4 Организация изучения учебного модуля

Методические рекомендации по организации изучения УМ с учетом использования в учебном процессе активных и интерактивных форм проведения учебных занятий даются в Приложении А.

5 КОНТРОЛЬ И ОЦЕНКА КАЧЕСТВА ОСВОЕНИЯ УЧЕБНОГО МОДУЛЯ

Контроль качества освоения студентами УМ и его составляющих осуществляется непрерывно в течение всего периода обучения с использованием балльно-рейтинговой системы (БРС), являющейся обязательной к использованию всеми структурными подразделениями университета.

Для оценки качества освоения модуля используются формы контроля: текущий – регулярно в течение всего семестра; рубежный – на девятой неделе семестра; семестровый – по окончании изучения УМ.

Оценка качества освоения модуля осуществляется с использованием фонда оценочных средств, разработанного для данного модуля, по всем формам контроля в соответствии с положением от 27.09.2011 № 32 «Об организации учебного процесса по основным образовательным программам высшего профессионального образования».

Содержание видов контроля и их график отражены в технологической карте учебного модуля (Приложение Б).

6 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

Учебно-методическое и информационное обеспечение учебного модуля представлено Картой учебно-методического обеспечения (Приложение Б).

Технологическая карта модуля с оценкой различных видов учебной деятельности по этапам контроля приведена в приложении В.

7 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОГО МОДУЛЯ

Для осуществления образовательного процесса по модулю необходимо:

– аудитория, оборудованная техническими средствами для демонстрации лекций-визуализаций;

– компьютерный класс для проведения лабораторных и самостоятельных работ. На персональных компьютерах должна быть установлена среда программирования *Android SDK*.

Кафедра ПМИ располагает данными средствами в полном объеме.

ВОПРОСЫ ДЛЯ ПОДГОТОВКИ

1. Проектирование ПО:

1.1. Программный прототип. Назначение и виды программных прототипов. Инструменты прототипирования.
1.2. Опыт взаимодействия (UX). Способы оценки дизайна. Методика эвристической оценки.
1.3. Виды требований. Способы документирования требований: ТЗ, Issue tracking.
1.4. Анализ требований. Подходы к проектированию программной архитектуры. Способы описания архитектуры, уровни детализации.
1.5. Объектно-ориентированный анализ. Элементы объектной модели. Признаки объектов и классов.
1.6. Объектно-ориентированный анализ. Отношения между классами и объектами. Процесс проектирования.
1.7. Шаблоны проектирования. Назначение, виды, примеры.

2. Управление проектами:

2.1. Понятие жизненного цикла ПО. Процессы и стадии жизненного цикла. Модели жизненного цикла.
2.2. Подходы к управлению проектами. Понятие «проектного треугольника». Общий обзор методологий управления проектами.
2.3. Подходы к планированию задач. Критический путь. Диаграмма Ганта.
2.4. Подходы к планированию задач. Agile. Расстановка приоритетов. Оценка трудоёмкости. Диаграммы сгорания задач.
2.5. Управление рисками. Подходы. Простые методы борьбы с рисками
2.6. Дисциплина управления рисками MSF. Цикл управления рисками. Планирование рисков, организация профилактики и мониторинга.

3. Программирование

3.1. Методы обработки исключительных ситуаций. Подходы к обработке ошибок.
3.2. Протоколирование действий программы. Виды протоколов, их назначение, адресаты и примерное содержание. Инструменты протоколирования.
3.3. Тестирование ПО. Решаемые задачи. Уровни тестирования.

3.4. Модульное тестирование. Цели. Проверяемые условия. Место тестов в разработке через тестирование (TDD). Инструменты модульного тестирования.
3.5. Отладка программ. Приёмы. Возможности отладчиков.
3.6. Виды дефектов. Инструменты трекинга дефектов: основные компоненты и способ применения. Место систем баг-трекинга в процессе разработки.
3.7. Сопровождаемость программного продукта. Понятие технического долга. Минимализм. Критерии хорошего именования. Принцип единственной ответственности. Комментарии.
3.8. Соглашения по кодированию (стили кодирования). Инспекции кода: виды и решаемые задачи. Рефакторинг.
3.9. Цели оптимизации программ. Понятие преждевременной оптимизации. Профилирование программ: возможности профилировщиков, способы сбора данных. Выбор тестовых данных для профилирования.
3.10. Подходы к оптимизации. Оптимизация алгоритмов и структур данных. Параллелизм и его ограничения. Кэширование. Низкоуровневые оптимизации.
3.11. Языки программирования: направления в развитии языков можно выделить, особенности этих направлений, решаемые задачи.
3.12. Примеры языков программирования, их ключевые особенности и ниши, в которых они нашли применение.
3.13. Кросс-платформенность. Подходы к организации, плюсы и минусы. Примеры.
3.14. Многозадачность: виды и их применение. Способы управления многозадачностью.

Технологическая карта модуля с оценкой различных видов учебной деятельности по этапам контроля приведена в приложении В.

ПРИЛОЖЕНИЯ (ОБЯЗАТЕЛЬНЫЕ):

А – Методические рекомендации по организации изучения учебного модуля

Б – Технологическая карта

В – Карта учебно-методического обеспечения УМ

Г – Паспорта компетенций

ЛИТЕРАТУРА

7.1 Основная литература

1. Гагарина Л.Г. Технология разработки программного обеспечения : учеб. пособие для вузов / Под ред.Л.Г.Гагариной. - М. : Форум-Инфра-М, 2009. - 399,[1]с. **Ф1-5**

2. Макконнелл Стив. Совершенный код = *Code complete* : Практ.руководство по разработке програм.обеспечения / Пер.с англ.под общ.ред.В.Г.Вшивцева. - 2-е изд. - М. : Русская редакция:Питер, 2005. - 867с. Сигла хранения Ф1-3

7.2 Дополнительная литература

1. Маршалл Донис. Надежный код. Оптимизация цикла разработки программного обеспечения = *Solid Code. Optimizing the Software Development Life Cycle. Best Practices* : практ. рекомендации : пер. с англ. / Донис Маршалл, Джон Бруно ; предисл. Джона Роббинса. - М. : Русская редакция ; СПб: БХВ-Петербург, 2010. - XIX, 297, [2] с. **Ф1-2**

ПРИЛОЖЕНИЕ А

(обязательное)

Методические рекомендации по организации изучения учебного модуля

«СОВРЕМЕННЫЕ ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ»

Учебный модуль «Системное и прикладное программное обеспечение и операционные системы» разделен на 3 раздела, по которым предусмотрены лекционные и практические занятия и лабораторные работы. В таблице А.1 отражены разделы модуля, технологии и формы проведения занятий, задания по самостоятельной работе студента и ссылки на необходимую литературу.

Образовательный процесс по дисциплине строится на основе комбинации образовательных технологий.

Интегральную модель образовательного процесса по дисциплине формируют технологии методологического уровня:

- модульно-рейтинговое обучение;
- контекстное обучение;
- технология поэтапного формирования умственных действий;
- технология развивающего обучения;
- элементы технологии развития критического мышления.

Реализация данной модели предполагает использование следующих технологий стратегического уровня, задающих организационные формы взаимодействия субъектов образовательного процесса, осуществляемых с использованием определенных тактических процедур:

– лекционные (вводная лекция, информационная лекция, обзорная лекция, лекция-консультация, лекция-визуализация, проблемная лекция);

– лабораторные работы (углубление знаний, полученных на теоретических занятиях, интерактивное выполнение, разработка, тестирование и отладка программ по темам лабораторных работ);

– тренинговые (формирование определенных умений и навыков, формирование алгоритмического мышления);

– активизации познавательной деятельности (приемы технологии развития критического мышления через чтение и письмо, работа с литературой, подготовка отчетов по лабораторным работам);

– самоуправления (самостоятельная работа студентов, самостоятельное изучение материала).

Рекомендуется использование информационных технологий при организации коммуникации со студентами для представления информации, выдачи рекомендаций и консультирования по оперативным вопросам (электронная почта), при проведения контроля выполнения лабораторных работ, использование мультимедиа-средств при проведении лекционных занятий.

Таблица 5 – Формы проведения лекционных занятий

Раздел модуля	Форма проведения	Задания на СРС	Дополнит. литература и интернет--ресурсы
ЭУМ 1. Технологии проектирования программных продуктов			
1. Введение. Предмет модуля , её объём, содержание и связь с другими дисциплинами учебного плана. Цели и задачи модуля . Литература.	Вводная лекция	решить задачи (ауд. СРС)	Брауде Э. Технология разработки программного обеспечения. Питер, 2004, http://mexalib.com/view/22479
2. Жизненный цикл ПО. Прототипирование. Опыт взаимодействия. Оценка дизайна.	Информационная лекция, лекция-визуализация	решить задачи (ауд. СРС)	Кормен Т. Лейзер Ч. Алгоритмы. Построение и анализ http://mexalib.com/view/22479
3. Сбор и анализ требований. Функциональные спецификации.	Информационная лекция лекция-визуализация	решить задачи (ауд. СРС)	Эккель Б. Философия Java . http://mexalib.com/view/22479
4. Принципы декомпозиции задач. Объектно-ориентированный анализ. Шаблоны проектирования.	Информационная лекция лекция-визуализация	решить задачи (ауд. СРС)	Брауде Э. Технология разработки программного обеспечения. Питер, 2004, http://mexalib.com/view/22479
ЭУМ 2. Управление проектами			
5. Организация жизненного цикла. Планирование задач. Классический подход. Гибкие методологии.	Информационная лекция лекция-визуализация	решить задачи (ауд. СРС)	Брауде Э. Технология разработки программного обеспечения. Питер, 2004,

			http://mexalib.com/view/22479
6. Управление рисками. Подходы к управлению рисками. Дисциплина управления рисками MSF.	Информационная лекция лекция-визуализация	решить задачи (ауд. СРС)	Брауде Э. Технология разработки программного обеспечения. Питер, 2004, http://mexalib.com/view/22479
ЭУМ 3. Кодирование и тестирование программных продуктов			
7. Качество кода: ошибки и дефекты. Защитное программирование и его альтернативы. Протоколирование. Тестирование и отладка ПО. Системы отслеживания ошибок.	Информационная лекция лекция-визуализация	решить задачи (ауд. СРС)	Брауде Э. Технология разработки программного обеспечения. Питер, 2004, http://mexalib.com/view/22479
8. Качество кода: сопровождаемость. Соглашения по стилям кодирования. Комментарии. Инспекции кода. Принципы минимализма. Технический долг и рефакторинг.	Информационная лекция лекция-визуализация	решить задачи (ауд. СРС)	Брауде Э. Технология разработки программного обеспечения. Питер, 2004, http://mexalib.com/view/22479
9. Качество кода: оптимизация программ. Профилирование. Техники оптимизации.	Информационная лекция лекция-визуализация	решить задачи (ауд. СРС)	Брауде Э. Технология разработки программного обеспечения. Питер, 2004, http://mexalib.com/view/22479
10. Системы контроля версий. Централизованные и распределённые системы. Организация командной	Информационная лекция лекция-визуализац	решить задачи (ауд. СРС)	Брауде Э. Технология разработки программного обеспечения. Питер,

работы.	ия		2004, http://mexalib.com/view/22479
11. Парадигмы программирования. Современные языки программирования.	Информационная лекция лекция-визуализация	решить задачи (ауд. СРС)	Брауде Э. Технология разработки программного обеспечения. Питер, 2004, http://mexalib.com/view/22479
12. Кросс-платформенность. Многозадачность.	Информационная лекция лекция-визуализация	решить задачи (ауд. СРС)	Брауде Э. Технология разработки программного обеспечения. Питер, 2004, http://mexalib.com/view/22479

Приложение Б
(обязательное)

Технологическая карта
учебного модуля «СОВРЕМЕННЫЕ ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ »
семестр 5, ЗЕТ 6, вид аттестации дифференц. зачет, ., акад.часов 216, баллов рейтинга 300

№ и наименование раздела учебного модуля, КП/КР	№ не-де-ли сем.	Трудоемкость, ак.час					Форма текущего контроля успев. (в соотв. с паспортом ФОС)	Макс.. кол-во баллов рейтин-га
		Аудиторные занятия				СРС		
		ЛЕК	ПЗ	ЛР	АСРС			
УЭМ 1. **Технологии проектирования программных продуктов**	**1-5**	**12**	**12**	**12**	**6**	**36**		
1.1Введение. . Жизненный цикл ПО. Прототипирование.		4	4	4	4	12		
1.2. Сбор и анализ требований. Функциональные спецификации		4	4	4	4	12	**ЛР-1**	50
1.3Принципы декомпозиции задач.		4	4	4	4	12		
УЭМ 2. **Управление проектами.**	**6-10-**	**12**	**12**	**12**	**6**	**36**	**СР-1**	50
2.1. Организация жизненного цикла.		6	6	6	3	18		
2.2. Управление рисками.		6	6	6	3	18	**ЛР-2**	50
УЭМ 3. **Кодирование и тестирование программных продуктов**	**11-18**	**12**	**12**	**12**	**6**	**36**		
3.1Качество кода: ошибки и дефекты..		**2**	**2**	**2**	**2**	**3**	**СР-2**	5050
3.2Качество кода: сопровождаемость. Соглашения по стилям.		2	2	2	2	3	**ЛР-3**	50
3.3Качество кода: оптимизация программ.		**2**	**2**	**2**	**2**	**3**		
3.4Системы контроля версий.		**2**	**2**	**2**	**2**	**3**	**ЛР-4**	50
3.5Парадигмы программирования. Современные языки программирования		**2**	**2**	**2**	**2**	**3**		
3.6Кросс-платформенность. Многозадачность		**2**	**2**	**2**	**2**	**3**		
Рубежная аттестация								
Дифференцированный зачет						36		
Итого:		36	36	36	18	108		300

(Трудоемкость разделов УЭМ не должна быть, как правило, меньше двух академических часов)

Критерии оценки качества освоения студентами модуля (в соответствии с Положениями «Об организации учебного процесса по основным образовательным программам высшего профессионального образования» и «О фонде оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации студентов и итоговой аттестации выпускников»):

отлично – (90-100) % от 50 х Т
хорошо – (70-89) % от 50 х Т
удовлетворительно – (50-69) % от 50 х Т ,
неудовлетворительно – менее 50 % от 50 х Т

Т=6 — трудоемкость модуля в зачетных единицах

Приложение В

(обязательное)

Карта учебно-методического обеспечения

Учебного модуля

СОВРЕМЕННЫЕ ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Направление 01.03.02 – прикладная математика и информатика
Формы обучения очная Курс 3 Семестр 5

Часов**:** всего 216 , лекций 36 , практ. Зан 36, лаб. раб. 36 , СРС 108

Обеспечивающая кафедра ПМИ

Таблица 1- Обеспечение учебного модуля учебными изданиями

Библиографическое описание* издания (автор, наименование, вид, место и год издания, кол. стр.)	Кол. экз. в библ. НовГУ	Наличие в ЭБС
Учебники и учебные пособия		
1. Гагарина Л.Г. Технология разработки программного обеспечения : учеб. пособие для вузов / Под ред.Л.Г.Гагариной. - М. : Форум-Инфра-М, 2009. - 399,[1]с.	**Ф1-5**	
2 Орлов С.А. Технология разработки программного обеспечения [Электронный ресурс] : учеб. пособие / Орлов С.А., Цилькер Б.Я. ; СПб.: Питер, 2012. — 608 с. - Электрон. версия печ. публ. - Режим доступа: *http://bookwebmaster. narod.ru/programming.html, свободный. - Загл. с тит. экрана (дата обращения*: 02.01.2017).		да
Учебно-методические издания		
1 Рабочая программа модуля с приложениями «СОВРЕМЕННЫЕ ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ » /Авт.-сост. Т.В. Жгун ; НовГУ. – В.Новгород, 2017. – 24 с.		
2		

Таблица 2 – Информационное обеспечение учебного модуля

Название программного продукта, интернет-ресурса	**Электронный адрес**	**Примечание**
Общероссийский математический портал Math-Net.Ru [Электронный ресурс] / Математич. институт им. В. А. Стеклова РАН. - М.; 2002-2015. - Режим доступа: http:// http://www.mathnet.ru/about.phtml?option_lang=rus, свободный - Загл. с экрана (дата обращения: 17.01.2017).	*http://www.mathnet.ru.ru/*	общероссийский математический портал
Бесплатные учебники компьютерщику [Электронный ресурс] / Сайт бесплатной электронной библиотеки - Режим доступа: *http://bookwebmaster.narod.ru/programming.html, свободный. - Загл. с тит. экрана (дата обращения*: 17.01.2017).	*http://bookwebmaster.narod.ru/programming.html*	Сайт бесплатной электронной библиотеки
InfTech. Information Technologies. Информационные Технологии [Электронный ресурс] : [офиц. сайт] / Режим доступа: *http://www.inftech.webservis.ru/ /,* свободный. - Загл. с экрана (дата обращения: 17.01.2017).	*http://inftech.webservis.ru/it/conference/isanditc/2000/section3/rus/arrus16.html*	Сайт информационных технологий
Visual Studio [Электронный ресурс] : [офиц. сайт] / Microsoft-2015. - Режим доступа: http://www.visualstudio.com/ru-ru/, свободный. - Загл. с экрана (дата обращения: 17.01.2017).	http://www.microsoft.com/visualstudio/ru-ru	Среда программирования Microsoft Visual Studio 2010
Lazarus [Электронный ресурс] : [офиц. сайт] Lazarus and Free Pascal Team , -1993-2013. - Режим доступа: http://www.lazarus.freepascal.org/свободный. - Загл. с экрана (дата обращения: 17.01.2017).	http://www.lazarus.freepascal.org/	Среда программирования Lazarus
Интернет-университет информационных технологий [Электронный ресурс] : [офиц. сайт] / НОУ «ИНТУИТ», 2003 – 2015. - Режим доступа: *http://www.intuit.ru/* свободный. - Загл. с экрана (дата обращения: 17.01.2017).	*: http://www.intuit.ru/*	Интернет-университет информационных технологий [Электронный ресурс].
Библиотека MSDN. [Электронный ресурс] : [офиц. сайт] / Microsofe, -2014. - Режим доступа: *http://msdn.microsoft.com/library/* свободный. - Загл. с экрана (дата обращения: 17.01.2017).	: *http://msdn.microsoft.com/library/*	Библиотека MSDN. [Электронный ресурс].

Таблица 3 – Дополнительная литература

Библиографическое описание* издания (автор, наименование, вид, место и год издания, кол. стр.)	Кол. экз. в библ. НовГУ	Наличие в ЭБС
Гагарина Л.Г. Технология разработки программного обеспечения : учеб. пособие для вузов / Под ред.Л.Г.Гагариной. - М. : Форум-Инфра-М, 2009. - 399,[1]с. Ф1-5	5	
. Макконнелл Стив. Совершенный код = *Code complete* : Практ.руководство по разработке програм.обеспечения / Пер.с англ.под общ.ред.В.Г.Вшивцева. - 2-е изд. - М. : Русская редакция:Питер, 2005. - 867с. Сигла хранения Ф1-3	3	
. Маршалл Донис. Надежный код. Оптимизация цикла разработки программного обеспечения = *Solid Code. Optimizing the Software Development Life Cycle. Best Practices* : практ. рекомендации : пер. с англ. / Донис Маршалл, Джон Бруно ; предисл. Джона Роббинса. - М. : Русская редакция ; СПб: БХВ-Петербург, 2010. - XIX, 297, [2] с. Ф1-2	2	

СОГЛАСОВАНО
НБ НовГУ: _____ _____ _____
должность подпись расшифровка

Действительно для учебного года _____/_____
Зав. кафедрой _____ А.В, Колногоров
_____ _____ 20 г.
Действительно для учебного года _____/_____
Зав. кафедрой _____ А.В, Колногоров
_____ _____ 20 г.
Действительно для учебного года _____/_____
Зав. кафедрой _____ _____
подпись И.О.Фамилия
_____ _____ 20….. г.

Приложение Г
(обязательное)

Паспорта компетенции ОПК-3, ПК-7

ОПК-3 способность к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям;

ПК-7 способность к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения.

Код компетенции	Уровень освоения компетенции	Знать	Уметь	Владеть
ОПК-3	базовый	основы разработки алгоритмических и программных решений в области технологии разработки программного обеспечения.	разрабатывать алгоритмические и программные решения в области разработки программного обеспечения.;	навыками: разработки алгоритмических и программных решений в области разработки программного обеспечения.; .
ПК-7	базовый	основы разработки и применения алгоритмических и программных решений в области разработки программного обеспечения.	разрабатывать и применять алгоритмические и программные решения в области разработки программного обеспечения	навыками: разработки и применения алгоритмических и программных решений в области разработки программного обеспечения.

Уровни	**Показатели**	**Оценочная шкала**		
		удовлетворительно	**хорошо**	**отлично**
Базовый уровень	Знает основы разработки алгоритмических и программных решений в области системного и прикладного программирования.	Испытывает трудности в определении понятий, положений и методов, испытывает трудности при использовании методов системного и прикладного программирования..	Недостаточно четко объясняет понятия, положения и методы, не всегда корректно использует методы системного и прикладного программирования. .	Четко объясняет значение понятий, положений и методов, полностью владеет методами системного и прикладного программирования. .