



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Магнитогорский государственный технический университет им. Г.И. Носова»



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

ПРОГРАММНАЯ ИНЖЕНЕРИЯ

Направление подготовки (специальность)
01.03.02 Прикладная математика и информатика

Направленность (профиль/специализация) программы
Математическое и информационное обеспечение экономической деятельности

Уровень высшего образования - бакалавриат

Форма обучения
очная

Институт/ факультет	Институт естествознания и стандартизации
Кафедра	Прикладной математики и информатики
Курс	1
Семестр	2

Магнитогорск
2020 год

Рабочая программа составлена на основе ФГОС ВО по направлению подготовки 01.03.02 Прикладная математика и информатика (уровень бакалавриата) (приказ Минобрнауки России от 10.01.2018 г. № 9)

Рабочая программа рассмотрена и одобрена на заседании кафедры Прикладной математики и информатики

11.02.2020, протокол № 6

Зав. кафедрой С.И. Кадченко С.И. Кадченко

Рабочая программа одобрена методической комиссией ИЕиС

17.02.2020 г. протокол № 6

Председатель И.Ю. Мезин И.Ю. Мезин

Рабочая программа составлена:

доцент кафедры ПМИИ, канд. пед. наук Л.С. Рязанова Л.С. Рязанова

Рецензент:

доцент кафедры уравнений математической физики ФГАОУ ВО "ЮУрГУ (НИУ)",
канд. физ.-мат. наук Г.А. Закирова Г.А. Закирова



1 Цели освоения дисциплины (модуля)

формирование у обучающихся представления о задачах, методах и средствах программной инженерии как деятельности, нацеленной на создание программных продуктов, отвечающих потребностям заказчиков, с соблюдением плановых сроков и бюджета разработки. Формирование компетенций согласно ФГОС ВО

2 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина Программная инженерия входит в часть учебного плана формируемую участниками образовательных отношений образовательной программы.

Для изучения дисциплины необходимы знания (умения, владения), сформированные в результате изучения дисциплин/ практик:

Освоение информатики в рамках курса средней школы:

освоение дисциплин Информатика и Практикум на ЭВМ в объеме 1 семестра

Знания (умения, владения), полученные при изучении данной дисциплины будут необходимы для изучения дисциплин/практик:

Разработка интернет приложений

Теория языков программирования

3 Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля) и планируемые результаты обучения

В результате освоения дисциплины (модуля) «Программная инженерия» обучающийся должен обладать следующими компетенциями:

Код индикатора	Индикатор достижения компетенции
ПК-3	Способен владеть методами и средствами проектирования программного обеспечения и программных интерфейсов
ПК-3.1	Оценивает формализованное описание решений поставленных задач в соответствии с требованиями технического задания или других принятых в организации нормативных документов
ПК-3.2	Решает профессиональные задачи написания программного кода с использованием языков программирования, определения и манипулирования данными
ПК-3.3	Осуществляет контроль за оформлением программного кода в соответствии с установленными требованиями

4. Структура, объём и содержание дисциплины (модуля)

Общая трудоемкость дисциплины составляет 3 зачетных единиц 108 акад. часов, в том числе:

- контактная работа – 54,15 акад. часов;
- аудиторная – 51 акад. часов;
- внеаудиторная – 3,15 акад. часов
- самостоятельная работа – 18,15 акад. часов;
- подготовка к экзамену – 35,7 акад. часа

Форма аттестации - экзамен

Раздел/ тема дисциплины	Семестр	Аудиторная контактная работа (в акад. часах)			Самостоятельная работа студента	Вид самостоятельной работы	Форма текущего контроля успеваемости и промежуточной аттестации	Код компетенции
		Лек.	лаб. зан.	практ. зан.				
1. Предмет и основные понятия программной инженерии. Модели и процессы жизненного цикла программного обеспечения								
1.1 Общие принципы программной инженерии. Жизненный цикл ПО и процессы его разработки	2	1	2		1	Изучение литературы, выполнение лабораторной работы	Проверка выполнения лаб. работы	ПК-3.1, ПК-3.2
1.2 Основные модели жизненного цикла ПО. Стандарты программной инженерии.		1	2		1	Изучение литературы	Опрос, обсуждение	ПК-3.1, ПК-3.2
Итого по разделу		2	4		2			
2. Требования к программным средствам и спецификация требований								
2.1 Анализ требований к программному обеспечению. Анализ предметной области. Методы выделения требований. Методы описания и систематизации требований.	2	2	2/2И		2	Изучение литературы. Подготовка презентации	Опрос, обсуждение. Проверка презентаций	ПК-3.1, ПК-3.3
2.2 Использование различных видов графических диаграмм при описании требований. Основные понятия методов формальной		2	2		2	Изучение литературы	Опрос, обсуждение	ПК-3.1, ПК-3.3
Итого по разделу		4	4/2И		4			
3. Разработка программных средств. Парадигмы и технологии								

3.1 Основные понятия и принципы разработки ПО. Архитектура ПО. Парадигмы программирования.	2	2	2		2	Изучение литературы. Выполнение лабораторной работы	Проверка выполнения лабораторной работы	ПК-3.2, ПК-3.3
3.2 Структурное проектирование. Объектно-ориентированный анализ и проектирование.		2	6/2И		8	Изучение литературы. Выполнение лабораторной работы	Проверка выполнения лабораторной работы	ПК-3.2, ПК-3.3
Итого по разделу		4	8/2И		10			
4. Характеристики качества и аттестация программных средств.								
4.1 Понятие качества программного обеспечения и его основные характеристики. Методы контроля качества программного	2	2	4		0,5	Изучение литературы. Подготовка презентации	Опрос, обсуждение, проверка презентаций	ПК-3.2, ПК-3.3
4.2 Планирование аттестационного тестирования, его разновидности Инспектирование.		1	4/2И		0,5	Выполнение лаб.работ, изучение литературы	Проверка выполнения лаб.работ, опрос, обсуждение	ПК-3.3, ПК-3.2
Итого по разделу		3	8/2И		1			
5. Развитие и сопровождение программных средств								
5.1 Сопровождение ПО. Свойства сопровождаемого ПО. Реинжиниринг ПО. Наследуемые системы.	2	2	5		0,5	Изучение литературы, выполнение лаб.работ	Проверка выполнения лаб.работ, опрос, обсуждение	ПК-3.3, ПК-3.2
5.2 Повторное использование и переносимость ПО. Задачи и процессы переноса программ и данных на иные		2	5		0,65	Изучение литературы, выполнение лабораторных работ	Опрос, обсуждение, проверка выполнения лаб работ	
5.3 Промежуточный контроль								ПК-3.1, ПК-3.2, ПК-3.3
Итого по разделу		4	10		1,15			
Итого за семестр		17	34/6И		18,15		экзамен	
Итого по дисциплине		17	34/6И		18,15		экзамен	

5 Образовательные технологии

В ходе изучения дисциплины используются образовательные и информационные технологии:

1. Традиционные технологии обучения, предполагающие передачу информации в готовом виде, формирование учебных умений по образцу: лабораторные работы, семинары.

Использование традиционных технологий обеспечивает ориентирование студента в потоке информации, связанной с различными подходами к определению сущности, содержания, методов, форм развития и саморазвития личности; самоопределение в выборе оптимального пути и способов личностно-профессионального развития; систематизацию знаний, полученных студентами в процессе аудиторной и самостоятельной работы. Лабораторные занятия обеспечивают развитие и закрепление умений и навыков определения целей и задач саморазвития, а также принятия наиболее эффективных решений по их реализации.

2. Информационно-коммуникационные образовательные технологии – организация образовательного процесса, основанная на применении специализированных программных сред и технических средств работы с информацией.

Лабораторные занятия проводятся в компьютерных классах вычислительного центра ФГБОУ ВО «МГТУ». В ходе проведения лекционных занятий предусматривается:

- использование электронного демонстрационного материала по темам, требующим иллюстрации работы программных продуктов: MS Word, MS Excel, MS PowerPoint.

- в ходе проведения лабораторных работ предусматривается использование среды программирования PASCAL ABC, TurboDelphi, Visualstudio

- использование образовательного портала ФГБОУ ВО «МГТУ».

3. Интерактивные формы обучения, предполагающие организацию обучения как продуктивной творческой деятельности в режиме взаимодействия студентов друг с другом и с преподавателем

Использование интерактивных образовательных технологий способствует повышению интереса и мотивации учащихся, активизации мыслительной деятельности и творческого потенциала студентов, делает более эффективным усвоение материала, позволяет индивидуализировать обучение и ввести экстренную коррекцию знаний.

В рамках дисциплины «Компьютерная графика» предусматривается 66 часа аудиторных занятий (лекций, лабораторных работ), проводимых в семинара-дискуссии, беседы, межгруппового диалога.

4. Проблемная технология обучения

Методика, предлагаемая для изучения дисциплины ориентирована на лабораторные работы поисково-исследовательского типа, семинары проблемно-информационного характера и подготовку презентаций.

Лабораторные работы поисково-исследовательского типа

Обмен информацией, полученной студентами в ходе самостоятельного поиска и исследования по поставленной проблеме, рекомендуется организовать в рамках лабораторных работ. Ценность данной формы занятий в том, что в процессе обсуждения можно высказать собственное мнение и попытаться доказать его правильность.

При изучении дисциплины для каждого раздела предлагается перечень вопросов для самоконтроля. Возможны три варианта использования данных вопросов при изучении теоретического материала: либо для контроля полученных студентами знаний по окончании изучения раздела, либо для обсуждения каждого вопроса как мини-проблемы в ходе лабораторной работы, либо то и другое в определенном

сочетании. Допускается иная постановка вопросов преподавателем, а самостоятельная формулировка студентами вопросов для обсуждения при выполнении лабораторной работы только приветствуется. Лабораторные работы поисково-исследовательского типа не только способствует углубленной проработке теоретического материала предмета на протяжении всего изучения курса, но и развивают творческую самостоятельность студентов, способность к обобщениям, укрепляя их интерес к исследованиям, содействуя выработке практических навыков работы.

Предполагается подготовка презентаций по теме «Кривые Безье», «Алгоритмы построения фракталов» и др. с последующим выступлением на занятиях.

6 Учебно-методическое обеспечение самостоятельной работы обучающихся

Представлено в приложении 1.

7 Оценочные средства для проведения промежуточной аттестации

Представлены в приложении 2.

8 Учебно-методическое и информационное обеспечение дисциплины (модуля)

а) Основная литература:

1. Лаврищева, Е. М. Программная инженерия. Парадигмы, технологии и CASE-средства : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. — Москва : Издательство Юрайт, 2020. — 280 с. — (Высшее образование). — ISBN 978-5-534-01056-5. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/452156> (дата обращения: 14.04.2020)

2. Черткова, Е. А. Программная инженерия. Визуальное моделирование программных систем : учебник для вузов / Е. А. Черткова. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 147 с. — (Высшее образование). — ISBN 978-5-534-09172-4. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/452749> (дата обращения: 14.04.2020).

б) Дополнительная литература:

1. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 432 с. — (Высшее образование). — ISBN 978-5-534-07604-2. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/452137> (дата обращения: 14.04.2020).

2. Загоруйко, Ю. А. Искусственный интеллект. Инженерия знаний : учебное пособие для вузов / Ю. А. Загоруйко, Г. Б. Загоруйко. — Москва : Издательство Юрайт, 2020. — 93 с. — (Высшее образование). — ISBN 978-5-534-07198-6. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/455500> (дата обращения: 14.04.2020).

в) Методические указания:

1. Пахомов, А. Н. Мультипликация [Текст] : учеб.-метод. пособие к занятиям по ком-пьютерной графике / А. Н. Пахомов, Н. М. Мещерякова. - Магнитогорск : [Изд-во МаГУ], 2011. - Библиогр.: с. 87-88. Количество экземпляров – 5

2. Варфоломеева Т. Н. Практикум по программированию в DELPHI. Основные элементы библиотеки визуальных компонентов [Электронный ресурс] : практикум / Т. Н. Варфоломеева, С. А. Повитухин ; МГТУ. - Магнитогорск : МГТУ, 2018. - 1 электрон. опт. диск (CD-ROM). - Режим доступа: <https://magtu.informsystema.ru/uploader/fileUpload?name=3519.pdf&show=dcatalogues/1/1514333/3519.pdf&view=true>. - Макрообъект. - ISBN 978-5-9967-1173-4. - ISBN 978-5-9967-1104-8.

3) Логунова О. С. Сборник заданий по информатике для программистов

Магнитогорск : МГТУ, 2011. - 1 электрон. опт. диск (CD-ROM). - Режим доступа: <https://magtu.informsystema.ru/uploader/fileUpload?name=1246.pdf&show=dcatalogues/1/1123424/1246.pdf&view=true>. - Макрообъект.

г) Программное обеспечение и Интернет-ресурсы:

Программное обеспечение

Наименование ПО	№ договора	Срок действия лицензии
MS Windows 7 Professional(для классов)	Д-1227-18 от 08.10.2018	11.10.2021
MS Office 2007 Professional	№ 135 от 17.09.2007	бессрочно
7Zip	свободно	бессрочно
Anaconda Python	свободно	бессрочно
NotePad++	свободно	бессрочно
Borland Turbo Delphi	№112301 от 23.11.2005	бессрочно
MS Visual Studio 2013 Professional(для класса)	Д-1227-18 от 08.10.2018	11.10.2021
ABC Pascal	свободно	бессрочно
Adobe Reader	свободно	бессрочно
Far Manager	свободно	бессрочно

Профессиональные базы данных и информационные справочные системы

Название курса	Ссылка
Электронная база периодических изданий East View Information Services,	https://dlib.eastview.com/
Национальная информационно-аналитическая система – Российский индекс научного цитирования (РИНЦ)	URL: https://elibrary.ru/project_risc.asp
Поисковая система Академия Google (Google Scholar)	URL: https://scholar.google.ru/
Информационная система - Единое окно доступа к информационным ресурсам	URL: http://window.edu.ru/
Российская Государственная библиотека. Каталоги	https://www.rsl.ru/ru/4readers/catalogues/
Электронные ресурсы библиотеки МГТУ им. Г.И. Носова	http://magtu.ru:8085/marcweb2/Default.asp
Университетская информационная система РОССИЯ	https://uisrussia.msu.ru
Международная наукометрическая реферативная и полнотекстовая база	http://webofscience.com
Международная реферативная и полнотекстовая справочная база данных	http://scopus.com
Международная реферативная база данных по чистой и прикладной	http://zbmath.org/

9 Материально-техническое обеспечение дисциплины (модуля)

Материально-техническое обеспечение дисциплины включает:

Учебные аудитории для проведения занятий лекционного типа Доска, мультимедийный проектор, экран

Учебные аудитории для проведения лабораторных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации Персональные компьютеры с пакетом MS Office, вы-ходом в Интернет и с доступом в электронную информационно-образовательную среду университета;

Комплекс тестовых заданий для проведения рубежного и промежуточного контроля

Помещения для самостоятельной работы обучающихся Персональные компьютеры с пакетом MS Office, вы-ходом в Интернет и с доступом в электронную информационно-образовательную среду университета

Помещения для хранения и профилактического обслуживания учебного оборудования Шкафы для хранения учебно-методической документации, учебного оборудования и учебно-наглядных пособия

Приложение 1

Примерное задание для самостоятельной работы

Анализ рисков

Цель: осуществить анализ рисков методами «Матрица компромиссов» и «Таблица анализа рисков».

Теоретические вопросы

Для более глубокого понимания компромиссов программного проекта, часто бывает очень полезно изобразить заказчику зависимость основных характеристик проекта в виде треугольника компромиссов (рис. 3).

Рис. 3. Треугольник компромиссов

Утвержденное равновесие с заказчиком достигается в том случае, если исполнитель с учетом запрашиваемых параметров назвал и зафиксировал время (сроки) и ресурсы (смету). Следует учитывать, что любое изменение одной из сторон треугольника обязательно влечет изменение на двух оставшихся.

Для эффективного достижения компромиссов в течение всего жизненного цикла программного проекта на начальных этапах следует выявить и зафиксировать приоритет факторов (ресурсы, время, возможности). Если один из факторов фиксируется как неизменный, то влиять на него в течение проекта практически невозможно. Второй фактор согласовывается по следующему принципу: он будет обладать некоторым приоритетом в случае необходимости достижения компромиссов. Последний фактор просто принимается в соответствии с первыми двумя. Матрица компромиссов представлена в табл. 4

Таблица 4 – Матрица компромиссов

Фиксируется (Зафиксировано)	Согласовывается (Определено)	Принимается (Корректируемо)
Ресурсы		
Время (график)		
Возможности (набор функций программы)		

Традиционная модель управления проектами подразумевает четкую формулировку требований на начальном этапе проекта, разработку на основании технического задания. Подход компромиссов основывается на принципе изменяющихся проектных условий. Разработчикам необходимо проявлять гибкость и в любой момент времени быть готовым к изменениям и рискам. Методология Microsoft Solution Framework предлагает вычленять возможные риски и анализировать их, чтобы ими можно было управлять. Риском называется проблема, которая еще не возникла. В свою очередь, проблемой называют риск, который уже материализовался. Причиной возникновения любых рисков являются неопределенности в программном проекте. Следовательно, нужно стремиться к выявлению рисков. Иногда это осуществляется методом мозгового штурма, опроса экспертов, метода Дельфи и др. или на опыте предыдущих проектов, опыте других разработчиков.

Риски делят на два вида:

- 1) известные – риски, которые определены, оценены, для них возможно планирование и анализ;
- 2) неизвестные – они не могут быть заранее идентифицированы или спрогнозированы.

Б. Бозм приводит список 10 наиболее распространенных рисков программного проекта [1]:

- 1) дефицит специалистов;
- 2) нереалистичные сроки и бюджет;
- 3) реализация несоответствующей функциональности;
- 4) разработка неправильного пользовательского интерфейса;
- 5) «золотая сервировка», перфекционизм, ненужная оптимизация и оттачивание деталей;

18

Приложение 2

Код индикатора	Индикатор достижения компетенции	Оценочные средства
Код и содержание компетенции		
ПК-3.1:	Оценивает формализованное описание решений поставленных задач в соответствии с требованиями технического	Примерные вопросы к экзамену: Перечислите способы формализации требований. Расскажите о способах и техниках "вытягивания" требований.

	задания или других принятых в организации нормативных документов	Перечислите разные виды документов, формализующих требования. Расскажите об отличии функциональных и нефункциональных требований. Расскажите о типовом <i>цикле работы</i> с требованиями. . Перечислите типовые ошибки при работе с требованиями. . Что такое риски проекта и критерии приемки. Что такое планирование проекта? В чем заключаются цель/назначение планирования проекта и каков его результат? Опишите, что такое сетевая диаграмма проекта. Как она составляется?
ПК-3.2:	Решает профессиональные задачи написания программного кода с использованием языков программирования, определения и манипулирования данными	Примерные задания к лабораторным работам: 1. Вычислить характеристики модульности и сложности программы для заданного варианта. 2. Разработать UML диаграмму классов для заданного варианта. Выполнение лабораторных работ по технологии функционального, логического и объектно-ориентированного программирования
ПК-3.3:	Осуществляет контроль за оформлением программного кода в соответствии с установленными требованиями	Примерный перечень вопросов к экзамену Расскажите о понятии <i>baseline</i> . Перечислите и кратко охарактеризуйте различные способы <i>контроля качества</i> ПО. . Дайте определение процессов верификации и аттестации. . Дайте определение тестирования и кратко прокомментируйте его. . Что означает в контексте тестирования ожидаемое поведение программы? . Что входит в искусственные, специально заданные условия воздействия на систему, которые имеются в виду в определении тестирования? . В чем важность концепции теста? . В чем преимущества <i>автоматического тестирования</i> перед "ручным"? . В чем трудности <i>автоматического тестирования</i> ? Приведите свои собственные примеры проблем с интерфейсами к тестируемым системам.

б) Порядок проведения промежуточной аттестации, показатели и критерии оценивания:

Промежуточная аттестация по дисциплине «Программная инженерия» включает теоретические вопросы и практические задания, позволяющие оценить степень сформированности компетенций, проводится в форме экзамена.

Показатели и критерии оценивания экзамена:

– 5 баллов оценка «отлично» – обучающийся набирает, если демонстрирует высокий уровень сформированности компетенций, всестороннее, систематическое и глубокое знание учебного материала, свободно выполняет практические задания, свободно оперирует знаниями, умениями, применяет их в ситуациях повышенной сложности.

– 4 балла оценка «хорошо» – обучающийся набирает, если демонстрирует средний уровень сформированности компетенций: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.

– 3 балла оценка «удовлетворительно» – обучающийся набирает, если демонстрирует пороговый уровень сформированности компетенций: в ходе контрольных мероприятий допускаются ошибки, проявляется отсутствие отдельных знаний, умений, навыков, обучающийся испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

– на оценку «не удовлетворительно» (1-2 балла) – обучающийся не может показать знания на уровне воспроизведения и объяснения информации, не может показать интеллектуальные навыки решения простых задач.