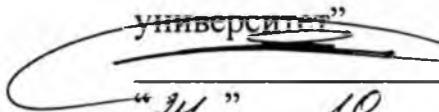


Учреждение образования  
«Белорусский государственный экономический университет»

УТВЕРЖДАЮ

Первый проректор учреждения  
образования «Белорусский  
государственный экономический  
университет»

 Е. Ф. Киреева

«24» 10 2022 г.

Регистрационный № УД 5324-28ч

**Технологии программирования**  
Учебная программа учреждения высшего образования  
по учебной дисциплине для специальности  
1-31 03 06 «Экономическая кибернетика» (по направлениям)

Учебная программа составлена на основе типового учебного плана учреждения высшего образования по специальности 1-31 03 06 «Экономическая кибернетика», дата утверждения 26.07.2013, регистрационный номер № G 31-1-038/пр.-тип.

**СОСТАВИТЕЛИ:**

*Тарасов С.А.*, доцент кафедры математических методов в экономике учреждения образования «Белорусский государственный экономический университет», кандидат технических наук, доцент

**РЕЦЕНЗЕНТЫ:**

*И.В. Кашиникова*, заведующий кафедрой микропроцессорных систем и сетей учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», кандидат физико-математических наук, доцент;

*В.В. Косьянчук* – заведующий кафедрой высшей математики учреждения образования «Белорусский государственный экономический университет», кандидат физико-математических наук, доцент

**РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:**

Кафедрой математических методов в экономике учреждения образования «Белорусский государственный экономический университет»  
(протокол № 2 от 28.09.2022)

Научно-методическим советом учреждения образования «Белорусский государственный экономический университет»  
(протокол № 1 от 19 10 2022)

## Пояснительная записка

С развитием и внедрением в повседневную жизнь информационно-коммуникационных технологий резко возрос интерес к программированию. Современным специалистам приходится иметь дело с огромным количеством разнообразных языков программирования различных уровней и назначений. Потребность в эффективных средствах разработки программного обеспечения привела к появлению объектно-ориентированных систем программирования, предназначенных для быстрой разработки программных приложений.

Дисциплина "Технологии программирования" направлена на формирование у студента понимания основ технологий программирования с использованием различных подходов и процессов.

Целью изучения дисциплины является практическое освоение общих принципов и современных методов технологии программирования, практическое применение современных информационных технологий при создании программных продуктов и программных комплексов различного назначения.

Задачи преподавания дисциплины:

- приобретение теоретических основ знаний, необходимых для разработки и проектирования программных продуктов;
- приобретение знаний о системе классификаций программ, в зависимости от их состава и разработанной к ним документации;
- формирование и развитие умений использовать полученные знания для создания макетов программных продуктов;
- формирование и развитие умений ставить задачу и разрабатывать алгоритм ее решения, использовать прикладные системы программирования, разрабатывать основные программные документы, анализировать производительность получаемых решений.

В результате изучения учебной дисциплины выпускник должен обладать следующими компетенциями:

- БПК-3. Способен анализировать требования и проектировать компоненты программного обеспечения
- Знает методологии разработки, методы и средства программного обеспечения и технологии программирования. Основные стандарты в области инфокоммуникационных систем и технологий, в том числе стандарты Единой системы программной документации
- Умеет разрабатывать варианты реализации требований к программному обеспечению, в том числе на основе готовых решений и

шаблонов. Ставить задачу и разрабатывать алгоритм ее решения, использовать прикладные системы программирования, разрабатывать основные программные документы. Работать с современными системами программирования, включая объектно-ориентированные

– Имеет навыки по оценке и согласованию требований к программному обеспечению на основе диалога с заказчиком, применения библиотек классов и инструментальных средств при проектировании и разработке программного обеспечения, анализа предметной области сложных программных систем на концептуальном, логическом, математическом и алгоритмическом уровнях.

Изучение дисциплины направлено на формирование способности решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности

В соответствии с учебным планом специальности 1-31 03 06 «Экономическая кибернетика» учебная программа рассчитана на 110 часов, из них аудиторных занятий 102 часов, в том числе: лекций – 34 часов, лабораторных занятий – 34 часов.

Форма получения высшего образования - очная.

Форма текущей аттестации – зачет.

## **Содержание учебного материала**

### **Тема 1. Базовые концепции технологии программирования**

Введение в технологию программирования, программную инженерию. Понятие программного средства. Технология программирования и основные этапы ее развития. Технологические процессы и порядок их прохождения - стадии (с использованием знаний, методов и средств). Классификация технологических подходов. Классификация технологических процессов. Классификация технологических стадий. Проблемы и перспективы развития технологий программирования.

### **Тема 2. Классический технологический процесс программной разработки**

Возникновение и исследование идеи продукта. Особенности управления программным проектом. Анализ требований и проектирование программного продукта. Процесс программирования программного продукта. Тестирование и отладка программного продукта. Ввод программы в действие. Эксплуатация и сопровождение программного продукта. Процесс завершения эксплуатации программного продукта.

### **Тема 3. Проблемы разработки сложных программных систем**

Понятие сложной программы и отличия сложных программ от простых. Признаки простой программы. Основные проблемы разработки сложных программ. Свойства сложной программы. Принципы работы со сложными программами. Организационные, инженерные, технические аспекты в создании сложного ПО. Практически полезная сложная программная система и понятие ее «правильности».

вводимых с клавиатуры в интерактивном режиме и выполнение файлов с исходным кодом в пакетном режиме. Вход в интерактивный режим работы. Пакетный режим работы.

### **Тема 4. Жизненный цикл программного обеспечения**

Жизненный цикл характеризует весь период существования программного обеспечения. Артефакты жизненного цикла. Роли сотрудников на различных этапах создания и эксплуатации ПО. Модели жизненного цикла ПО. Стандарты жизненного цикла. 3-х ступенчатая модель жизненного цикла. Каскадная модель жизненного цикла. Итеративные или инкрементальные модели. Спиральная модель жизненного цикла Бозма.

## **Тема 5. Унифицированный процесс разработки Rational (RUP ) и экстремальное программирование (XP)**

RUP как пример «тяжелого» процесса, детально описанного и предполагающего поддержку собственно разработки исходного кода ПО большим количеством вспомогательных действий. Иерархическое пошаговое детальное описание предпринимаемых действий при использовании RUP. Варианты использования (use cases). Четыре основные фазы жизненного цикла. Модель вариантов использования, модель анализа, модель проектирования, модель реализации, модель развертывания, модель тестирования RUP. Распределение работ между различными дисциплинами в проекте по RUP.

Экстремальное программирование (Extreme Programming, XP) как эволюционный метод разработки ПО «снизу вверх». XP еще известен как метод «живой» разработки (Agile Development Method). Группа «живых» методов помимо экстремального программирования включает методы SCRUM , DSDM (Dynamic Systems Development Method , метод разработки динамических систем) , Feature Driven Development (разработка, управляемая функциями системы) и др. Схема потока работ в XP.

## **Тема 6. Стандарты и методики, используемые при разработке программных средств**

Виды стандартов. Методики проектирования. Стандартизация жизненного цикла программного средства в системе государственных стандартов и стандартов ISO. Стандарт пользовательского интерфейса. Стандарт проектирования. Стандарт оформления документации. Стандартизация процесса разработки программ и программной документации.

Основы Программной Инженерии (по SWEBOOK). Основы построения профессионального интерфейса. Стандартные элементы интерфейса. Выбор шрифтов, цветов. Принципы построения интерфейсов: «Золотое сечение», «Кошелек Миллера», «Принцип группировки», «Бритва Оккама или KISS», «Видимость окружает полезность», «Умное заимствование». Эвристические правила Якоба Нильсена (правило обратной связи, информированность пользователя, средства обеспечения обратной связью, время оповещения, равенство между системой и реальным миром, свобода действий пользователя, последовательность и стандарты, предупреждение ошибок, понимание лучше, чем запоминание, гибкость и эффективность использования, эстетичный и

минималистический дизайн, распознавание и исправление ошибок, описание ошибки, описание решения проблемы).

## **Тема 7. Технологические средства разработки программного обеспечения**

Инструментарий технологии программирования. Состав и назначение инструментария технологий программирования. Инструменты разработки программных средств. Средства для создания приложений: локальные средства, интегрированные среды. Case-средства. Общая характеристика Case-средств. Классификация Case-средств.

Инструментальная среда разработки. Средства поддержки проекта. Отладчики.

Автоматизация процесса разработки программного обеспечения. Автоматическое документирование ПО и автоматическая кодогенерация. Анализ требований и проектирование спецификаций.

Понятие Case-средств. Определение потребностей в Case-средствах. Оценка и выбор Case-средств.

Характеристики Case-средств: ERWin, BPWin, PowerDesigner, Rational Rose. Характеристика пакета создания диаграмм Microsoft Visio.

## **Тема 8. Методы проектирования структуры ПО**

Классификация методов проектирования. По степени автоматизации проектных работ: неавтоматизированное (традиционное) проектирование и автоматизированное проектирование (Case-технология и ее элементы). По принятой методологии процесса разработки: метод функциональной декомпозиции (структурное проектирование программных средств), информационное моделирование предметной области и связанных с ней приложений, объектно-ориентированное проектирование программных средств. В зависимости от объекта структурирования: функционально-ориентированные методы (ориентированные на обработку), методы структурирования данных (ориентированные на данные). Методы, ориентированные на обработку: модульное программирование, функциональная декомпозиция, проектирование с использованием потока данных, технология структурного анализа проекта. Методы проектирования, основанные на использовании структур данных: методология Джексона, методология Уорнера, метод иерархических диаграмм, объектно-ориентированная методология.

## **Тема 9. Структура программных продуктов**

Внутренняя структура. Функциональная структура приложения. Модульная структура программных продуктов.

Метод пошаговой детализации для проектирования структуры ПО. Декомпозиция системы методом пошаговой детализации. Структурная схема программы.

Визуальное моделирование программной системы. Понятие модель. Графические (визуальные) модели.

Структурный подход к разработке программных средств. Методы структурного анализа и проектирования ПО. Принципы инженерии ПО. Два базовых: принцип «разделяй и властвуй» и принцип иерархического упорядочивания. Принципы второстепенные: абстрагирования, формализации, «упрятывания», концептуальной общности, полноты, непротиворечивости, логической независимости, независимости данных, структурирования данных, доступа конечного пользователя.

Средства структурного подхода: диаграммы потоков данных, ER-диаграммы, диаграммы декомпозиции, структурные схемы (архитектура программного продукта в виде иерархии взаимосвязанных программных модулей с идентификацией связей между ними).

## **Тема 10. Характеристика последовательности этапов и используемых методов**

Метод функционального моделирования SADT. Общие сведения. Функциональная модель. Состав функциональной модели. Построение иерархии диаграмм (метод моделирования IDEF0-IDEF3). Типы связей между функциями.

Информационное моделирование предметной области. Моделирование потоков данных (процессов). Состав диаграмм потоков данных. Построение иерархии диаграмм потоков данных.

Объектно-ориентированный подход к проектированию программных средств. Методы объектно-ориентированного анализа и проектирования ПО. Сущность объектно-ориентированного подхода. Принципы объектно-ориентированного представления: абстрагирование, инкапсуляция, модульность, иерархия. Классы. Общая характеристика. Виды отношений между классами. Ассоциации классов. Объектная модель.

Унифицированный язык моделирования UML. Структурные модели: диаграммы классов, диаграммы компонентов, диаграммы размещения. Примеры диаграмм.

Модели поведения: диаграммы деятельности, диаграммы схем состояний, диаграммы последовательности, диаграммы сотрудничества (кооперации), диаграммы взаимодействия, диаграммы вариантов использования (Use Case). Примеры диаграмм.

## **Тема 11. Методы отладки и тестирования программного средства**

Введение в тестирование. Основные понятия. Типичные ошибки (синтаксические, логические, времени исполнения). Защита от ошибок. Методы защиты от ошибок, методы тестирования. Тест и тестирование. Этапы процесса тестирования. Стратегии тестирования (черный ящик, белый ящик). Подходы к тестированию: сверху вниз, снизу вверх. Отладка. Принципы и виды отладки. Заповеди отладки. Автономная отладка и тестирование программного модуля. Комплексная отладка и тестирование программного средства. Стандартная техника отладки.

## **Тема 12. Методы защиты программ и данных**

Защита программного обеспечения от несанкционированного приобретения, использования, распространения, модифицирования, исследования. Средства защиты: технические, юридические, организационные. Методы защиты ПО. Проверка оригинального носителя. Ввод серийного номера. Активация программного обеспечения. Использование электронных ключей. Подходы к защите программ от несанкционированного копирования. Криптографические методы защиты. Метод привязки к идентификатору. Методы, основанные на работа с переходами и стеком. Манипуляции с кодом программы. Методы противодействия динамическим способам снятия защиты программ от копирования. Уязвимости методов защиты.

### **Тема 13. Документирование и оценка качества программных продуктов**

Понятие качества программного обеспечения. Стандартизация характеристик качества. Критерии качества. Дерево характеристик качества. Выбор показателей качества.

Оценка качества программных продуктов. Методы контроля качества. Методы оценки качества. Методы обеспечения качества.

Надежность и функциональность программных средств. Обеспечение надежности и функциональности программного средства.

Документирование программных средств. Внешнее описание программного средства. Спецификация качества программного средства. Функциональная спецификация программного средства.

### **Тема 14. Организация процесса проектирования ПС**

Понятие проекта. Представление проекта в виде «черного ящика». Классификация проектов. Класс проекта. Тип проекта. Масштаб проекта. Участники процесса разработки и их роль в разработке проекта.

Окружение проекта: структура организации-исполнителя проекта, организационная культура, заинтересованные в проекте лица. Виды деятельности, входящие в управление проектом: управление содержанием проекта и качеством, метрики ПО, управление ресурсами, специфика управления персоналом, управление рисками, управление коммуникациями и информационным обеспечением.

Технологии коллективной разработки (технические командные роли, психологические командные роли, типы совместной деятельности). Авторская разработка. Общинная модель разработки.

### **Тема 15. Архитектура ПО**

Каркасы и образцы проектирования. Классификация архитектур. Типы архитектур и их модели. Качество и выбор архитектуры. Диаграммы компонентов. Интерфейсы. Компоновка системы. Разновидности компонентов. Моделирование реализации системы.

### **Тема 16. Основы компонентной объектной модели**

COM-объекты, COM-интерфейсы. Технологии OLE и ActiveX.

Повторное использование COM-объектов. Связывание объектов различного происхождения. Технология COM и объектно-ориентированное программирование. Реализация интерфейсов.

Технология DCOM. Структуры диалога; поддержка пользователя; многооконные интерфейсы; примеры реализации интерфейсов с пользователем с использованием графических пакетов. «Заглушки». «Маленькие хитрости» в программировании. Статические, полустатические и динамические типы данных. Простые и составные типы данных, операция квалификации. Технологии распределенных вычислений: RPC, RMI, Corba, DCOM.

### **Тема 17. Сцепление и связность модулей**

Оценка схем иерархии в структурных методах разработки (критерии Майерса). Особенности сцепления и связности в объектно-ориентированном ПО (закон Деметера, видимость на уровне классов и объектов).

### **Требования к курсовой работе**

Курсовой работы учебным планом не предусмотрено.

**Учебно-методическая карта учебной дисциплины «Технологии программирования»  
для дневной формы получения высшего образования  
для специальности  
1-31 03 06 «Экономическая кибернетика» (по направлениям)**

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов						Иное*	Форма контроля знаний	
		Лекции	Практические занятия	Семинарские	Лабораторные занятия	Количество часов УСР				
						Л	Пз			Лаб
1	Базовые концепции технологии программирования	2			2			[1,3]	Тест	
2	Классический технологический процесс программной разработки	2			2			[2]	Тест	
3	Проблемы разработки сложных программных систем	2			2			[1,4]	Проверка работоспособности устройства и готовности к выполнению им программ	
4	Жизненный цикл программного обеспечения	2			2			[1,5]	Тест	
5	Унифицированный процесс разработки Rational (RUP) и экстремальное программирование (XP)	2			2			[2,3]	Контрольный запуск заданной программы	
6	Стандарты и методики, используемые при разработке программных средств	2			2			[2,4]	Тест + контрольный запуск заданной программы	
7	Технологические средства разработки программного обеспечения	2			2			[1, 4]	Тест + контрольный запуск заданной программы	
8	Методы проектирования структуры ПО	2			2			[1, 3]	Тест + контрольный запуск заданной программы	
9	Структура программных продуктов	2			2			[2, 3]	Тест	
10	Характеристика последовательности этапов и используемых методов	2			2			[3, 5]	Тест + контрольный запуск заданной	

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов						Иное*	Форма контроля знаний	
		Лекции	Практические занятия	Семинарские	Лабораторные занятия	Количество часов УСП				
						Л	Пз			Лаб
									программы	
11	Методы отладки и тестирования программного средства	2			2				[3д] Тест + контрольный запуск заданной программы	
12	Методы защиты программ и данных	2			2				[1д] Тест + контрольный запуск заданной программы	
13	Документирование и оценка качества программных продуктов	2			2				[1,4] Тест + контрольный запуск заданной программы	
14	Организация процесса проектирования ПС	2			2				[4, 5] Тест	
15	Архитектура ПО	2			2				[1, 4] Тест + контрольный запуск заданной программы	
16	Основы компонентной объектной модели	2			2				[5] Тест	
17	Сцепление и связность модулей	2			2				[4, 5] Тест + контрольный запуск заданной программы	
	<b>Всего часов</b>	<b>34</b>			<b>34</b>					

## ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

### *Методические рекомендации по организации самостоятельной работы студентов по учебной дисциплине «Технологии программирования»*

В овладении знаниями учебной дисциплины важным этапом является самостоятельная работа студентов. Рекомендуется бюджет времени для самостоятельной работы в среднем 1,5-2 часа на 2-х часовое аудиторное занятие.

Основными направлениями самостоятельной работы студента являются:

- первоначально подробное ознакомление с программой учебной дисциплины;
- ознакомление со списком рекомендуемой литературы по учебной дисциплине в целом и ее разделам, наличие ее в библиотеке и других доступных источниках, изучение необходимой литературы по теме, подбор дополнительной литературы;
- изучение и расширение лекционного материала преподавателя за счет специальной литературы, консультаций;
- подготовка к лабораторным занятиям по специально разработанным планам с изучением основной и дополнительной литературы;
- подготовка к выполнению диагностических форм контроля (контрольные задания на компьютере);
- подготовка к зачету.

*Примерный перечень тем управляемой самостоятельной работы обучающихся:*

УСРС в данном курсе не предусмотрена учебным планом;

*Средства диагностики результатов учебной деятельности:*

- краткий опрос по теории проектирования программ по материалам прошедших лекций и лабораторным работам перед началом занятий в подгруппах;
- краткий опрос по программным средствам, способным реализовать проектирование ПО из материалов прошедших лекций и лабораторным работам перед началом занятий в подгруппах.

## ЛИТЕРАТУРА

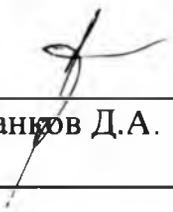
### *Основная:*

1. Жоголев, Е. А. Технология программирования – М.: Научный мир, 2004. – 216 с.
2. Смирнов, А. А. Технологии программирования : учебно-практическое пособие / А. А. Смирнов. - М.: Евразийский открытый институт, 2011 – 192 с.
3. Камаев, В. А. Технологии программирования / В. А. Камаев, В. В. Костерин. - М.: Высшая школа, 2005 - 359с.
4. Рудаков, А. В. Технология разработки программных продуктов : учебник / А. В. Рудакова. - 6-е изд., стереотип.- М.: Академия, 2011 – 208 с.
5. Светлов, Н. М. Информационные технологии управления проектами: учебное пособие / Н. М. Светлов, Г. Н. Светлова. – 2-е изд., перераб.и доп. – М.: ИНФРА-М, 2011. – 232 с.
6. Гниденко, И. Г. Технологии и методы программирования : учебное пособие для студентов высших учебных заведений, обучающихся по инженерно-техническим направлениям / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров ; [Санкт-Петербургский гос. экон. ун-т]. - Москва : Юрайт, 2021. - 233, [2] с.
7. Федотова, Е. Л. Информационные технологии и системы : учебное пособие для студентов высших учебных заведений, обучающихся по направлению подготовки 09.03.03 "Прикладная информатика" / Е. Л. Федотова. - М. : ФОРУМ : ИНФРА-М, 2021. - 351 с.

### *Дополнительная литература:*

1. Гвоздева, Т.В. Проектирование информационных систем: учебное пособие / Т.В. Гвоздева, Б.А. Баллод. – Ростов-н/Д: Феникс, 2009 – 508с.
2. Иванова, Г.С. Технология программирования: Учебник / Г.С. Иванова. - М.: КноРус, 2011 – 336с.
3. Котляров, В.П. Основы тестирования программного обеспечения: Учебное пособие / В.П. Котляров. – М.: Интернет-Университет Информационных Технологий, 2009 – 285с
4. Солдатов, В.П. Управление программными проектами / В.П. Солдатов. - М.: БИНОМ, 2010 - 384с

## ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ УВО

Название учебной дисциплины, которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)
Бухгалтерский учёт и экономический анализ	Кафедра бухгалтерского учета, анализа и аудита в отраслях народного хозяйства	 Панков Д.А.	протокол № 1 23/08.22

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ УВО  
на \_\_\_\_ / \_\_\_\_ учебный год

№ п/п	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры экономической информатики (протокол № \_\_\_\_ от \_\_\_\_\_ 201\_ г.)

Заведующий кафедрой  
д.э.н., профессор

\_\_\_\_\_  
(подпись) Г.О.Читая  
(И.О.Фамилия)

УТВЕРЖДАЮ  
Декан факультета  
к.э.н., доцент

\_\_\_\_\_  
(подпись) (И.О.Фамилия)