

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ

Ректор Белорусского
государственного университета
А.Д.Король

26 мая 2025 г.

Регистрационный № 3596/б.



ВВЕДЕНИЕ В ТЕХНОЛОГИЮ ПРОГРАММИРОВАНИЯ

Учебная программа учреждения образования по учебной дисциплине для
специальности:

6-05-0533-10 Информатика

Профилизация: Технологии разработки сложных информационных систем

2025 г.

Учебная программа составлена на основе ОСВО 6-05-0533-10-2023 и учебного плана № 6-5.3-58/04 от 15.05.2023.

СОСТАВИТЕЛЬ:

А.Н.Курбацкий, заведующий кафедрой технологий программирования факультета прикладной математики и информатики Белорусского государственного университета, доктор технических наук, профессор

РЕЦЕНЗЕНТ:

И.А.Король, кандидат физико-математических наук, доцент, ведущий научный сотрудник, заместитель директора Государственного предприятия «Центр систем идентификации» Национальной академии наук Беларусь

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой технологий программирования БГУ
(протокол № 17 от 15.05.2025);

Научно-методическим советом БГУ
(протокол № 10 от 22.05.2025)

Заведующий кафедрой



А.Н.Курбацкий

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Цели и задачи учебной дисциплины

Цель учебной дисциплины – изучение методологий, технологий программирования, принципов построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения, методов организации работы в коллективах разработчиков программного обеспечения. Особое внимание уделяется созданию программного обеспечения для сложных интегрированных систем.

Рассматриваются подходы к стандартизации в жизненном цикле программного обеспечения, сравнительному анализу технологических платформ разработки программного обеспечения. Рассматривается роль искусственного интеллекта в жизненном цикле программного обеспечения.

Задачи учебной дисциплины:

- дать студентам базу, необходимую для успешного усвоения материала дисциплин профилизации;
- дать студентам базу, необходимую для успешного освоения современных технологий программирования;
- приобретенные знания позволяют понять принципы организации работы коллективов, технологии разработки ПО различных компаний поставщиков;
- получить знания, необходимые им в дальнейшем для успешной работы в качестве разработчика программного обеспечения либо руководителя проектом в области ИКТ.

Место учебной дисциплины в системе подготовки специалиста с высшим образованием.

Учебная дисциплина относится к модулю дисциплинам профилизации «Технологии разработки сложных информационных систем» компонента учреждения образования.

Учебная программа по дисциплине профилизации «Введение в технологию программирования» разработана в соответствии с учебным планом и образовательным стандартом общего высшего образования по специальности 6-05-0533-10 Информатика.

Учебная дисциплина «Введение в технологию программирования» знакомит студентов с основными методологиями технологий программирования и принципами организации процесса разработки ПО.

Основой для изучения методологий, технологий программирования являются следующие дисциплины: «Промышленное программирование», «Дискретная математика и математическая логика», «Операционные системы».

Учебная дисциплина «Введение в технологию программирования» непосредственно связана с параллельно изучаемыми учебными дисциплинами: «Методы оптимизации» модуля «Математические модули принятия решений» компонента учреждения высшего образования, «Технологии

программирования» модуля «Программирование» государственного компонента.

Требования к компетенциям

Освоение учебной дисциплины «Введение в технологию программирования» должно обеспечить формирование следующих компетенций:

Универсальные компетенции:

Владеть основами исследовательской деятельности, осуществлять поиск, анализ и синтез информации;

Решать стандартные задачи профессиональной деятельности на основе применения информационно-коммуникационных технологий.

Базовые профессиональные компетенции:

Применять при проектировании приложений такие парадигмы программирования как структурное, объектно-ориентированное и функциональное программирование;

Разрабатывать программное обеспечение в интегрированных средах разработки.

Специализированные компетенции

Применять методологии и технологии программирования. Использовать принципы построения, приемы работы с инструментальными средствами для создания программного обеспечения. Владеть современными средствами телекоммуникации. Взаимодействовать со специалистами смежных профилей.

В результате освоения учебной дисциплины студент должен:

знать:

- основные стадии жизненного цикла;
- общепринятые технологии разработки ПО;
- классические и современные тенденции в разработке ПО;
- парадигмы программирования;
- языки моделирования.

уметь:

- осуществлять разработку и поддержку ПО;
- определять и устранять возникающие в процессе разработки ПО проблемы;
- эффективно организовывать работу сотрудника сферы ИТ;
- быть в курсе технологий разработки ПО различных компаний поставщиков.

иметь навык:

- владения базовыми инструментами организации и планирования процесса разработки ПО;
- анализа проблемных ситуаций в сфере ИКТ.

Структура учебной дисциплины

Дисциплина изучается в 5 семестре. В соответствии с учебным планом всего на изучение учебной дисциплины «Введение в технологию программирования» отведено для очной формы получения высшего образования – 108 часов, в том числе 68 аудиторных часа: лекции – 34 часа, лабораторные занятия – 34 часа. Из них:

Лекции – 34 часа, лабораторные занятия – 30 часов, управляемая самостоятельная работа – 4 часа.

Трудоемкость учебной дисциплины составляет 3 зачетные единицы.

Форма промежуточной аттестации – экзамен.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Тема 1. Предмет и задачи дисциплины

Работа над проектом создания учебно-методического комплекса как основная методология изучения дисциплины. Связь с другими дисциплинами.

Тема 2. Жизненный цикл программного обеспечения

Основные стадии жизненного цикла. Модели жизненного цикла: поэтапная, каскадная, спиральная и т.д. Возможности формализации и автоматизации отдельных стадий и этапов жизненного цикла. Стандартизация в жизненном цикле программного обеспечения. Роли искусственного интеллекта.

Тема 3. Программное обеспечение для сложных интегрированных информационных систем

Сложные интегрированные информационные системы. Их специфика, эволюция развития. Некоторые критерии классификации. Их роль в цифровой трансформации.

Тема 4. Концептуальная фаза жизненного цикла разработки программного обеспечения (ПО)

Определение проекта, разработка его концепции, формирование бизнес-идей, постановка, сравнительная оценка альтернатив, сбор исходных данных. Мозговой штурм.

Тема 5. Методологии разработки ПО для сложных интегрированных информационных систем

Основные методологии. Методологии ООП. Специфика методологий для ПО сложных интегрированных информационных систем. Методологические стратегии. Методологии с использованием искусственного интеллекта.

Тема 6. Стандарты в области разработки программного обеспечения

Классификация стандартов. Модели разработки. Стандарты ISO, стандарты SW-CMM. Паттерны проектирования.

Тема 7. Разработка и производство ПО

Гибкие методологии. Экстремальное программирование. SCRUM. Автоматизация разработки и производства ПО. «Промышленное программирование». Работы в программировании. Искусственный интеллект в программировании.

Тема 8. Команда разработки ПО

Индивидуальная и командная разработка. Роли в команде разработчиков. Специфика команд при разработке сложных интегрированных

информационных систем. Дистанционное сотрудничество. Искусственный интеллект в команде.

Тема 9. Некоторые современные аспекты тестирования

Тестирование до разработки. Разработка через тестирование. Парное программирование. Системы контроля версий. Автоматизация тестирования. Самотестирующиеся и самокорректирующиеся программы.

Тема 10. Лицензирование

Понятие лицензий. Лицензирование. Проприетарное и открытое ПО. Свободное программное обеспечение. Возрастание его роли в современном мире. Специфика разработки.

Тема 11. Обеспечение информационной безопасности разрабатываемого ПО

Угрозы безопасности ПО. Модели угроз. Методы и средства анализа безопасности ПО. Методологии защиты ПО. Жизненный цикл ПО: технологическая и эксплуатационная безопасность ПО.

Тема 12. Сборочное программирование

Основы индустрии ПО. Методы сборочного построения сложных интегрированных информационных систем. Основы построения технологических линий разработки ПО.

Тема 13. Искусственный интеллект и его влияние на инженерию программирования

Влияние ИИ на этапы жизненного цикла разработки ПО. Изменение традиционного жизненного цикла разработки ПО. Специфика разработки ПО для сложных интегрированных информационных систем.

Тема 14. Парадигмы программирования и развитие языков программирования

Модульное программирование. Структурное программирование. Процедурное программирование. Функциональное программирование. Логическое программирование, Событийно-ориентированное программирование. Аспектно-ориентированное программирование. Сервисное программирование. Генерирующее программирование. Метапрограммирование. Развитие языков программирования.

Тема 15. Языки моделирования. Методологии и стандарты

Языки визуального моделирования. UML. IDEF. SADT. ARIS.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Очная форма получения высшего образования с применением
дистанционных образовательных технологий (ДОТ)

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Форма контроля знаний
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Иное	
1	2	3	4	5	6	7	8
1	Предмет и задачи дисциплины.	2					Устный опрос
2	Жизненный цикл программного обеспечения	2			2		Устный опрос
3	Программное обеспечение для сложных интегрированных информационных систем	2			2		Отчёт по лабораторной работе
4	Концептуальная фаза жизненного цикла разработки программного обеспечения (ПО)	2			2		Контрольная работа
5	Методологии разработки ПО для сложных интегрированных информационных систем	2			2		Отчёт по лабораторной работе
6	Стандарты в области разработки программного обеспечения	2			2		Отчёт по лабораторной работе
7	Разработка и производство ПО	4			2	2	Контрольная работа
8	Команда разработки ПО	2			2		Отчёт по лабораторной работе
9	Некоторые современные аспекты тестирования	2			2		Отчёт по лабораторной

							工作中
10	Лицензирование	2		2			Устный опрос
11	Обеспечение информационной безопасности разрабатываемого ПО	2		2		2	Контрольная работа
12	Сборочное программирование	2		2			Отчёт по лабораторной работе
13	Искусственный интеллект и его влияние на инженерию программирования	2		2			Проект
14	Парадигмы программирования и развитие языков программирования.	4		4			Контрольная работа
15	Языки моделирования. Методологии и стандарты	2		2			Отчёт по лабораторной работе
	Всего	34		30		4	

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Основная литература

1. Мартин, Р. Идеальная работа. Программирование без прикрас / Роберт Мартин ; [пер. с англ. И. Рузмайкиной]. - Санкт-Петербург ; Москва ; Минск : Питер, 2022. - 380 с.
2. Кравчук, А. С. Язык Java. Основы объектно ориентированного программирования : учебно-методическое пособие для студентов учреждений высшего образования / А. С. Кравчук, А. И. Кравчук, Е. В. Кремень ; БГУ. – Минск : БГУ, 2023. – URL: <https://elib.bsu.by/handle/123456789/308478>.
3. Лой, М. Программируем на Java / Марк Лой, Патрик Нимайер, Дэниэл Лук ; [пер. с англ. Е. Матвеев]. – 5-е междунар. изд. – Санкт-Петербург ; Москва ; Минск : Питер, 2023. – 540 с. – URL: <https://ibooks.ru/bookshelf/387726/reading>.
4. Юрча, И.А. Технологии программирования : пособие для учреждений высшего образования, / И. А. Юрча, Ж. И. Щербокович ; Академия управления при Президенте Республики Беларусь. – Минск : Академия управления при Президенте РБ, 2025. – 177 с.
5. Паньшин, Б.Н. Основы информационных технологий : учебно-методическое пособие для студентов учреждений высшего образования / Б. Н. Паньшин, И. А. Каракун, Т. А. Бронская ; БГУ. – Минск : БГУ, 2024. – 143 с.

Дополнительная литература

1. Берг, Д., Деоган, Д., Савано, Д. Безопасно by design / Д. Берг, Д. Деоган, Д. Савано. – М: Издательство Прогресс книга, 2021. – 432 с.
2. Дэвис, Д., Дэниелс, К. Философия DevOps. Искусство управления ИТ / Д. Дэвис, К. Дэниелс. – СПб.: Питер, 2017. – 416 с.
3. Казарин, О.В. Методология защиты программного обеспечения / О.В. Казарин. – М.: МЦНМО, 2009. – 464 с.
4. Вайсфельд, М. Объектно-ориентированный подход / М. Вайсфельд. – СПб.: Питер, 2020. – 256 с.
5. Павловская, Т.А. С/C++. Программирование на языке высокого уровня / Т.А. Павловская. – СПб.: Питер, 2020. – 464 с.
6. Меджуи, Мехди, Уайлд, Эрик, Ронни, Митра. Непрерывное развитие API. Правильные решения в изменчивом технологическом ландшафте / М. Меджуи, Э. Уайлд, М. Ронни. – СПб.: Питер, 2023. – 368 с.
7. Лаврищева, Е.М. Программная инженерия. Парадигмы, технологии и CASE-средства: учебник для вузов / Е.М. Лаврищева. – М.: Издательство Юрайт, 2016. – 280 с.
8. Орлов, С.А. Программная инженерия: технологии разработки программного обеспечения / С.А. Орлов. – СПб.: Питер, 2017. – 640 с.

Перечень рекомендуемых средств диагностики и методика формирования итоговой отметки

Объектом диагностики компетенций студентов являются знания, умения, практический опыт, полученные ими в результате изучения учебной дисциплины. Выявление учебных достижений студентов осуществляется с помощью мероприятий текущего контроля и промежуточной аттестации.

Текущий контроль работы студента проходит в следующих формах:

- технические: лабораторные работы, выполняемые на компьютере. Они оцениваются исходя из читаемости и оптимизированности программного кода;

- устно-письменные: устная и/или письменная (в виде отчёта) защита лабораторных работ, оцениваемая на основе полноты и последовательности ответа (отчёта), полноты раскрытия содержания выполненного задания, понимания работы алгоритмов и методов, использованных при выполнении задания, проект; контрольная работа;

- устные: устные опросы, проводимые в целях первичного мониторинга усвоения материала студентами и оцениваемые исходя из полноты и последовательности ответа, понимания основных понятий, методов и алгоритмов, изложенных на лекционных или лабораторных занятиях.

Формой промежуточной аттестации по дисциплине «Введение в технологию программирования» предусмотрен экзамен.

Для формирования итоговой отметки по учебной дисциплине используется модульно-рейтинговая система оценки знаний студента, дающая возможность проследить и оценить динамику процесса достижения целей обучения. Рейтинговая система предусматривает использование весовых коэффициентов для текущей и промежуточной аттестации студентов по учебной дисциплине.

Формирование итоговой отметки в ходе проведения контрольных мероприятий текущей аттестации (примерные весовые коэффициенты, определяющие вклад текущей аттестации в отметку при прохождении промежуточной аттестации):

- устный опрос – 20%,
- отчет по лабораторным работам – 40%,
- проект – 10%,
- контрольная работа – 30%.

В случае успешной защиты отчётов по всем лабораторным работам, положительных результатов контрольной работы, устного опроса и успешной защиты проекта допускается определение результатов промежуточной аттестации по дисциплине без проведения дополнительного опроса на экзамене. При этом явка обучающегося на экзамен является обязательной.

Итоговая отметка по дисциплине рассчитывается на основе итоговой отметки текущей аттестации (модульно-рейтинговой системы оценки знаний) 40 % и экзаменационной отметки 60 %.

Примерная тематика лабораторных занятий

Лабораторная работа № 1. Жизненный цикл программного обеспечения.

Лабораторная работа № 2. Программное обеспечение для сложных интегрированных информационных систем.

Лабораторная работа № 3. Концептуальная фаза жизненного цикла разработки программного обеспечения (ПО).

Лабораторная работа № 4. Методологии разработки ПО для сложных интегрированных информационных систем.

Лабораторная работа № 5. Стандарты в области разработки программного обеспечения.

Лабораторная работа № 6. Разработка и производство ПО.

Лабораторная работа № 7. Команда разработки ПО.

Лабораторная работа № 8. Некоторые современные аспекты тестирования.

Лабораторная работа № 9. Лицензирование.

Лабораторная работа № 10. Обеспечение информационной безопасности разрабатываемого ПО.

Примерный перечень заданий для управляемой самостоятельной работы

Тема 7. Разработка и производство ПО. (2ч.)

Экстремальное программирование. Автоматизация разработки и производства ПО.

(Форма контроля – контрольная работа).

Тема 11. Обеспечение информационной безопасности разрабатываемого ПО. (2ч.)

Методы и средства анализа безопасности ПО. Методологии защиты ПО.

(Форма контроля – контрольная работа).

Описание инновационных подходов и методов к преподаванию учебной дисциплины

При организации образовательного процесса используются следующие инновационные подходы:

практико-ориентированный подход, который предполагает:

- освоение содержания образования через решения практических задач;
- приобретение навыков эффективного выполнения разных видов профессиональной деятельности;
- ориентацию на генерирование идей, реализацию групповых студенческих проектов, развитие предпринимательской культуры;
- использованию процедур, способов оценивания, фиксирующих сформированность профессиональных компетенций.

метод проектного обучения, который предполагает:

- способ организации учебной деятельности студентов, развивающий актуальные для учебной и профессиональной деятельности навыки

планирования, самоорганизации, сотрудничества и предполагающий создание собственного продукта;

- приобретение навыков для решения исследовательских, творческих, социальных, предпринимательских и коммуникационных задач.

Методические рекомендации по организации самостоятельной работы обучающихся

Самостоятельная работа с целью изучения материала учебной дисциплины предполагает работу с рекомендованной учебной литературой и Интернет-ресурсами. Теоретические сведения закрепляются выполнением лабораторных заданий, при выполнении которых следует руководствоваться методическими разработками, размещенными в электронной библиотеке университета и на образовательном портале. Также могут быть предложены дополнительные задания (тесты, задания для самостоятельного выполнения) для самооценки и более глубокого усвоения полученного материала.

Примерный перечень вопросов к экзамену

1. Основные стадии жизненного цикла ПО. Модели жизненного цикла: поэтапная, каскадная, спиральная и т.д.
2. Модульное программирование. Структурное программирование. Процедурное программирование.
3. Возможности формализации и автоматизации отдельных стадий и этапов жизненного цикла.
4. Функциональное программирование. Логическое программирование.
5. Стандартизация в жизненном цикле программного обеспечения.
6. Событийно-ориентированное программирование. Аспектно-ориентированное программирование. Сервисное программирование.
7. Угрозы безопасности ПО. Модели угроз. Методы и средства анализа безопасности ПО.
8. Генерирующее программирование. Метапрограммирование.
9. Специфика разработки ПО для сложных интегрированных информационных систем.
10. Методологии защиты ПО.
11. Языки визуального моделирования ПО.
12. Искусственный интеллект и его влияние на инженерию программирования.
13. Жизненный цикл ПО: технологическая и эксплуатационная безопасность ПО.
14. Развитие языков программирования.
15. Лицензирование ПО.
16. Сборочное программирование: основы индустрии ПО.
17. Проприетарное и открытое ПО.

18. Методы сборочного построения сложных интегрированных информационных систем.
19. Свободное программное обеспечение. Возрастание его роли в современном мире. Специфика разработки.
20. Основы построения технологических линий разработки ПО.
21. Классификация стандартов в области разработки программного обеспечения. Модели разработки. Стандарты ISO, стандарты SW-CMM.
22. «Промышленное программирование». Работы в программировании.
23. Паттерны проектирования.
24. Гибкие методологии разработки ПО. Экстремальное программирование. SCRUM.
25. Концептуальная фаза жизненного цикла разработки программного обеспечения (ПО)
26. Специфика команд при разработке сложных интегрированных информационных систем. Дистанционное сотрудничество.
27. Основные методологии разработки ПО. Методологии ООП.
28. Автоматизация тестирования. Самотестирующиеся и самокорректирующиеся программы.
29. Тестирование до разработки. Разработка через тестирование.
30. Основные методологии разработки ПО. Методологии ООП.
31. Автоматизация тестирования. Самотестирующиеся и самокорректирующиеся программы.
32. Специфика методологий разработки ПО сложных интегрированных информационных систем.
33. Методологические стратегии.
34. Парное программирование. Системы контроля версий.
35. Индивидуальная и командная разработка ПО. Роли в команде разработчиков.

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ УО

Название учебной дисциплины, с 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)
Формальная семантика и верификация программ	Кафедра технологий программирования	Предложения отсутствуют	Рекомендовать к утверждению учебную программу (протокол № 17 от 15.05.2025)

Заведующий кафедрой технологий программирования, д.т.н., профессор

А.Н.Курбацкий

15.05.2025

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ УО
на _____ / _____ учебный год

№ п/п	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры
_____ (протокол № _____ от _____ 20_ г.)

Заведующий кафедрой

УТВЕРЖДАЮ
Декан факультета