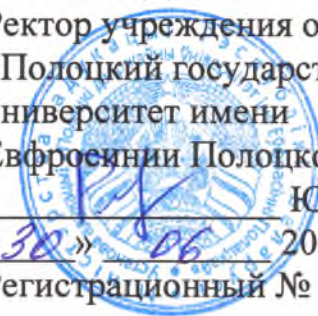


Учреждение образования
«Полоцкий государственный университет имени Евфросинии Полоцкой»

УТВЕРЖДАЮ

Ректор учреждения образования
«Полоцкий государственный
университет имени
Евфросинии Полоцкой»


Ю.Я. Романовский
« 30 » 06 2023 г.
Регистрационный № УД- 948/23 уч

МОДУЛЬ «ПРОГРАММИРОВАНИЕ-2»

ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ

Учебная программа учреждения образования
по учебной дисциплине для специальности
6-05-0533-12 «Кибербезопасность»

2023 г.

Учебная программа составлена на основе учебного плана специальности 6-05-0533-12 «Кибербезопасность». Регистрационный 14-23/уч. ФКНЭ от 04.04.2023 для дневной очной формы получения высшего образования

СОСТАВИТЕЛЬ:

Сергей Васильевич Кухта, старший преподаватель кафедры математики и компьютерной безопасности учреждения образования «Полоцкий государственный университет имени Евфросинии Полоцкой»

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой математики и компьютерной безопасности учреждения образования «Полоцкий государственный университет имени Евфросинии Полоцкой» (протокол № 6 от «30» 05 2023 г.).

Методической комиссией факультета компьютерных наук и электроники учреждения образования «Полоцкий государственный университет имени Евфросинии Полоцкой» (протокол № 10 от «22» 06 2023 г.).

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Учебная дисциплина «Технологии программирования» знакомит студентов с методами разработки программных продуктов с использованием различных инструментальных средств, включая интеграцию с CASE- системами. Особое внимание в учебной дисциплине уделяется методам проектирования программных систем.

В учебной дисциплине также рассматриваются основные этапы процесса создания программного продукта, в том числе анализ предъявляемых к нему требований, составление технического задания, проектирование программы в целом, а также её сопровождение после установки у заказчика. Отдельные темы посвящены определению типа процесса разработки и управлению проектом, что особенно важно при работе в команде.

При изложении учебной дисциплины «Технологии программирования» важно продемонстрировать комплексный подход к решению основных вопросов, возникающих в процессе разработки программных проектов.

Целью изучения дисциплины «Технологии программирования» является освоение основных понятий и методов современной теории и практики создания программных систем.

Образовательная цель: формирование составной части банка знаний, получаемых будущими специалистами в процессе учебы и необходимых им в дальнейшем для успешной работы.

Развивающая цель: формирование у студентов основ проектирования программных систем.

Задачи изучения дисциплины «Технологии программирования»:

- формирование у обучаемых понятий метод, методология, технология разработки программных систем;
- изучение методов и технологий разработки программного обеспечения (ПО) с целью использования при решении практических задач;
- использование современных инструментов для моделирования и проектирования программных продуктов.

В результате изучения дисциплины «Технологии программирования» обучаемый должен:

знать:

- классические и современные модели процесса разработки программных систем;
- принципы, способы и методы разработки и поддержки программных продуктов;
- методы организации и управления процессом разработки ПО;
- стандарты процесса разработки ПО;
- принципы работы с различными инструментами для поддержки процесса разработки программных систем;

уметь:

- моделировать программные системы;
- проводить анализ созданных моделей;
- использовать модели при разработке и поддержке программных систем;
- определять основные качественные характеристики программ;
- пользоваться современными CASE-средствами для поддержки процесса разработки и тестирования ПО;

владеть:

- основными методами анализа и проектирования ПО;
- методами оценки качества ПО;
- навыками самообразования и способами использования приемов и методов разработки программных продуктов.

Требования к уровню освоения содержания учебной дисциплины. При изучении дисциплины «Технологии программирования» у студентов специальности 6-05-0533-12 «Кибербезопасность» должен сформироваться набор компетенций, соответствующих присваиваемой по завершению высшего образования квалификации «Специалист по кибербезопасности», обеспечивающих выпускникам по указанной специальности успешность

применения полученных знаний и умений в дальнейшей профессиональной деятельности, он должен обладать следующими

специализированными компетенциями:

СК-32. Создавать программные, аппаратно-программные и технические средства и системы защиты информации. Владеть современными технологиями проектирования сложных систем и программных средств.

Перечень дисциплин, в продолжение и на базе которых изучается дисциплина.

Изучение дисциплины «Технологии программирования» базируется на знании дисциплины «Программирование на C++».

Перечень дисциплин, которые изучаются на базе дисциплины.

Материалы дисциплины «Технологии программирования» используются при изучении дисциплин «Базы данных», «Технология разработки программного обеспечения».

В соответствии с учебным планом по специальности 6-05-0533-12 «Кибербезопасность» на изучение учебной дисциплины отводится:

Форма получения высшего образования	дневная
Курс (курсы)	1
Семестр	1
Всего часов по дисциплине	180
Всего аудиторных часов по дисциплине	72
В том числе:	
Лекций, часов	36
Лабораторные занятия, часов	36
Самостоятельная работа, часов	108
Форма промежуточной аттестации	зачет
Трудоемкость дисциплины, зачетные единицы	5

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

ВВЕДЕНИЕ В ДИСЦИПЛИНУ

Цели и задачи изучения дисциплины. Введение в дисциплину. Содержание и цели дисциплины. Основные понятия.

Раздел 1 ЖИЗНЕННЫЙ ЦИКЛ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Тема 1.1 Основные понятия технологии программирования. Этапы её развития.

Основные понятия технологии программирования: метод, методология, программа, программное обеспечение (ПО), программный продукт, программное изделие, модуль, компонента. Классификация ПО. Общие принципы разработки программных средств. Основные этапы развития технологии программирования.

Тема 1.2 Понятия и основные этапы жизненного цикла ПО.

Жизненный цикл ПО. Стандарт жизненного цикла ПО ISO/IEC 12207. Оценка трудозатрат по этапам жизненного цикла ПО.

Тема 1.3 Модели жизненного цикла ПО.

Понятие модели жизненного цикла ПО. Каскадная модель. V-образная модель. Процесс макетирования ПО. Инкрементная модель. Спиральная модель. Компонентная модель. Модель быстрой разработки приложений. Выбор модели для реализации проекта.

Тема 1.4 Управление требованиями к ПО.

Организация разработки требований к сложным программным средствам. Требования заказчика к ПО. Разработка D-требований к ПО. Разработка технического задания.

Раздел 2 КЛАССИЧЕСКИЕ МЕТОДЫ АНАЛИЗА И ПРОЕКТИРОВАНИЯ ПРОГРАММНЫХ СИСТЕМ

Тема 2.1 Структурный подход к разработке ПО.

«Хаотическое» и структурное программирование. Принципы структурного проектирования. Типовая структура программного комплекса. Сцепление и связность модулей программного средства. Восходящее и нисходящее проектирование ПО.

Тема 2.2 Функциональное моделирование.

CASE-технологии структурного анализа и проектирования ПО. Методология функционального моделирования. Синтаксис IDEF0-диаграмм. Декомпозиция и её стратегии при IDEF0-моделировании. Примеры IDEF0-модели.

Тема 2.3 Моделирование потоков данных.

Методология структурного анализа потоков данных DFD. Примеры построения DFD-модели.

Тема 2.4 Методология моделирования процессов IDEF3.

Понятие динамического моделирования. Методология IDEF3. Основные элементы динамической модели. Правила и особенности построения IDEF3-модели. Декомпозиция в IDEF3.

Раздел 3 ОСНОВЫ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРЕДСТАВЛЕНИЯ ПО

Тема 3.1 Объектно-ориентированный подход к разработке ПО.

Объектно-ориентированная методология разработки ПО. Принципы объектного подхода. Классы и объекты. Объектно-ориентированный анализ. Объектно-ориентированное проектирование.

Тема 3.2 Унифицированный язык моделирования UML.

История развития и назначение UML. Принципы моделирования на языке UML. Структура языка UML. Диаграммы UML.

Тема 3.3 Диаграмма вариантов использования.

Назначение диаграммы. Сущности диаграммы. Отношения на диаграмме. Примеры диаграмм.

Тема 3.4 Диаграмма классов.

Назначение диаграммы. Сущности диаграммы. Отношения на диаграмме. Примеры диаграмм.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ «ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ»
Дневная форма получения высшего образования

Номер раздела, темы, занятия	Название раздела, темы, занятия; перечень изучаемых вопросов	Количество аудиторных часов				Материальное обеспечение занятия (наглядные, методические пособия и др.)	Литература	Формы контроля знаний
		Лекции	Лабораторные занятия	Практические занятия	Управляемая (контролируемая) самостоятельная работа студента			
1	2	3	4	5	6	7	8	9
	Раздел 1 Жизненный цикл программного обеспечения	8	6					
<i>Тема 1.1</i>	<i>Введение в дисциплину</i> Цели и задачи изучения дисциплины. Введение в дисциплину. Содержание и цели дисциплины. Основные понятия. <i>Основные понятия технологии программирования. Этапы её развития.</i> Основные понятия технологии программирования: метод, методология, программа, программное обеспечение (ПО), программный продукт, программное изделие, модуль, компонента. Классификация ПО. Общие принципы разработки программных средств. Основные этапы развития технологии программирования.	2				Презентация	Осн. лит.: [1], [2], [3]. Доп. лит.: [1], [2].	*Контрольное тестирование №1

1	2	3	4	5	6	7	8	9
Тема 1.2	Понятия и основные этапы жизненного цикла ПО. Жизненный цикл ПО. Стандарт жизненного цикла ПО ISO/IEC 12207. Оценка трудозатрат по этапам жизненного цикла ПО.	2				Презентация	Осн. лит.: [1], [3]. Доп. лит.: [1], [2].	Блиц-опрос
Тема 1.3	Модели жизненного цикла ПО. Понятие модели жизненного цикла ПО. Каскадная модель. V-образная модель. Процесс макетирования ПО. Инкрементная модель. Спиральная модель. Компонентная модель. Модель быстрой разработки приложений. Выбор модели для реализации проекта.	2				Презентация	Осн. лит.: [1]. Доп. лит.: [2].	*Контрольное тестирование №2
Тема 1.4	Управление требованиями к ПО. Организация разработки требований к сложным программным средствам. Требования заказчика к ПО. Разработка D-требований к ПО. Разработка технического задания.	2				Презентация	Осн. лит.: [1]. Доп. лит.: [2].	*Контрольное тестирование №3
	Лабораторная работа №1 Изучение предметной области.		2			Методические указания к лабораторной работе		Защита отчета по лабораторной работе
	Лабораторная работа №2 Систематизация и формализация требований к ПО.		2			Методические указания к лабораторной работе		Защита отчета по лабораторной работе
	Лабораторная работа №2 Систематизация и формализация требований к ПО.		2					
	Раздел 2 Классические методы анализа и проектирования программных систем	14	20					
Тема 2.1	Структурный подход к разработке ПО. «Хаотическое» и структурное программирование. Принципы структурного проектирования. Типовая структура программного комплекса.	2				Презентация	Осн. лит.: [1], [2]. Доп. лит.: [2].	*Контрольное тестирование №4
	Сцепление и связность модулей программного средства. Восходящее и нисходящее проектирование ПО.	2				Презентация	Осн. лит.: [1], [2]. Доп. лит.: [2].	Блиц-опрос

1	2	3	4	5	6	7	8	9
Тема 2.2	<i>Функциональное моделирование.</i> CASE-технологии структурного анализа и проектирования ПО. Методология функционального моделирования.	2				Презентация	Осн. лит.: [1], [2], [3]. Доп. лит.: [2].	Блиц-опрос
	Синтаксис IDEF0-диаграмм. Декомпозиция и её стратегии при IDEF0-моделировании. Примеры IDEF0-модели.	2				Презентация	Осн. лит.: [1], [2], [3]. Доп. лит.: [2].	Блиц-опрос
	Лабораторная работа №3 Построение функциональной модели системы.		2			Методические указания к лабораторной работе		Защита отчета по лабораторной работе
	Лабораторная работа №3 Построение функциональной модели системы.		2					
	Лабораторная работа №3 Построение функциональной модели системы.		2					
	Лабораторная работа №3 Построение функциональной модели системы.		2					
Тема 2.3	<i>Моделирование потоков данных.</i> Методология структурного анализа потоков данных DFD. Примеры построения DFD-модели.	2				Презентация	Осн. лит.: [1], [2]. Доп. лит.: [2].	Блиц-опрос
	Лабораторная работа №4 Построение диаграммы потоков данных.		2			Методические указания к лабораторной работе		Защита отчета по лабораторной работе
	Лабораторная работа №4 Построение диаграммы потоков данных.		2					
	Лабораторная работа №4 Построение диаграммы потоков данных.		2					
Тема 2.4	<i>Методология моделирования процессов IDEF3.</i> Понятие динамического моделирования. Методология IDEF3. Основные элементы динамической модели.	2				Презентация	Осн. лит.: [3]. Доп. лит.: [1], [2].	Блиц-опрос
	Правила и особенности построения IDEF3-модели. Декомпозиция в IDEF3.	2				Презентация	Осн. лит.: [3]. Доп. лит.: [1], [2].	Блиц-опрос
	Лабораторная работа №5 Построение диаграммы IDEF3.		2			Методические указания к лабораторной работе		Защита отчета по лабораторной работе

1	2	3	4	5	6	7	8	9
Тема 2.4	Лабораторная работа №5 Построение диаграммы IDEF3.		2			Методические указания к лабораторной работе		Защита отчета по лабораторной работе
	Лабораторная работа №5 Построение диаграммы IDEF3.		2					
	Раздел 3 Основы объектно-ориентированного представления ПО	14	10					
Тема 3.1	Объектно-ориентированный подход к разработке ПО. Объектно-ориентированная методология разработки ПО. Принципы объектного подхода. Классы и объекты.	2				Презентация	Осн. лит.: [1], [2], [3]. Доп. лит.: [2].	*Контрольное тестирование №5
	Объектно-ориентированный анализ. Объектно-ориентированное проектирование. Объектно-ориентированное программирование.	2				Презентация	Осн. лит.: [1], [2], [3]. Доп. лит.: [2].	Блиц-опрос
Тема 3.2	Унифицированный язык моделирования UML. История развития и назначение UML. Принципы моделирования на языке UML.	2				Презентация	Осн. лит.: [3]. Доп. лит.: [2].	Блиц-опрос
	Структура языка UML. Диаграммы UML.	2				Презентация	Осн. лит.: [3]. Доп. лит.: [2].	Блиц-опрос
Тема 3.3	Диаграмма вариантов использования. Назначение диаграммы. Сущности диаграммы. Отношения на диаграмме. Примеры диаграмм.	2				Презентация	Осн. лит.: [1], [3]. Доп. лит.: [2].	Блиц-опрос
	Лабораторная работа №6 Построение диаграммы вариантов использования.		2			Методические указания к лабораторной работе		Защита отчета по лабораторной работе
	Лабораторная работа №6 Построение диаграммы вариантов использования.		2					

1	2	3	4	5	6	7	8	9
Тема 3.4	Диаграмма классов. Назначение диаграммы. Сущности диаграммы.	2				Презентация	Осн. лит.: [3]. Доп. лит.: [2].	Блиц-опрос
	Отношения на диаграмме. Примеры диаграмм.	2				Презентация	Осн. лит.: [3]. Доп. лит.: [2].	*Контрольное тестирование №6
	Лабораторная работа №7 Построение диаграммы классов.		2			Методические указания к лабораторной работе		Защита отчета по лабораторной работе
	Лабораторная работа №7 Построение диаграммы классов.		2					
	Лабораторная работа №7 Построение диаграммы классов.		2					
	Всего (72 часа)	36	36					

* КОНТРОЛЬНЫЕ ТОЧКИ

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

ЛИТЕРАТУРА

Основная:

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, 2023. – 432 с.
2. Волк, В.К. Практическое введение в программную инженерию: учебное пособие для вузов – СПб.: Лань, 2022. – 100 с. – Текст: электронный // Лань : электронно-библиотечная система. URL: <https://e.lanbook.com/book/249848> (дата обращения: 19.09.2023).
3. Заботина, Н. Н. Методы и средства проектирования информационных систем: учебное пособие / Н.Н. Заботина. – Москва : ИНФРА-М, 2023. – 331 с. – - Текст : электронный. - URL: <https://znanium.ru/catalog/product/1902833> (дата обращения: 19.09.2023).

Дополнительная:

1. Вейцман, В.М. Проектирование информационных систем: учебное пособие для вузов – СПб.: Лань, 2022. – 316 с. – Текст: электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/208946> (дата обращения: 19.09.2023).
2. Проектирование информационных систем: учебник и практикум для вузов / Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук; под общей редакцией Д. В. Чистова. – Москва : Издательство Юрайт, 2023. – 293 с.

Елена Лаврищева Е. В.

ПЕРЕЧЕНЬ ЛАБОРАТОРНЫХ ЗАНЯТИЙ

Лабораторная работа №1 Изучение предметной области.

Лабораторная работа №2 Систематизация и формализация требований к ПО.

Лабораторная работа №3 Построение функциональной модели системы.

Лабораторная работа №4 Построение диаграммы потоков данных.

Лабораторная работа №5 Построение диаграммы IDEF3.

Лабораторная работа №6 Построение диаграммы вариантов использования.

Лабораторная работа №7 Построение диаграммы классов.

ПЕРЕЧЕНЬ ВОПРОСОВ ДЛЯ ПРОВЕДЕНИЯ ЗАЧЕТА

1. Технология программирования и основные этапы ее развития
2. Жизненный цикл ПО. Процессы жизненного цикла
3. Характеристика процесса разработки ПО
4. Модели жизненного цикла ПО. Выбор модели для реализации проекта
5. Каскадная модель жизненного цикла ПО: описание, критерии применения
6. Каскадная модель жизненного цикла ПО: преимущества и недостатки
7. V-образная модель жизненного цикла ПО: описание, критерии применения
8. V-образная модель жизненного цикла ПО: преимущества и недостатки
9. Модель макетирования ПО: её содержание, критерии применения
10. Модель макетирования ПО: преимущества и недостатки
11. Инкрементная модель жизненного цикла ПО: описание, критерии применения
12. Инкрементная модель жизненного цикла ПО: преимущества и недостатки
13. Спиральная модель жизненного цикла ПО: описание, критерии применения
14. Спиральная модель жизненного цикла ПО: преимущества и недостатки
15. Модель быстрой разработки приложений: описание, критерии применения
16. Модель быстрой разработки приложений: преимущества и недостатки
17. Организация разработки требований к сложным программным средствам
18. Методы выявления требований к сложным программным средствам
19. Процессы и этапы разработки требований к характеристикам сложных программных средств
20. Основные эксплуатационные требования к программным средствам
21. Принципы структурного подхода к проектированию ПО. Сцепление и связность модулей. Нисходящее и восходящее проектирование
22. Методология функционального моделирования. Синтаксис IDEF0-диаграмм. Декомпозиция и её стратегии при IDEF0-моделировании
23. Принципы объектного подхода к проектированию программных средств
24. Классы и объекты в объектно-ориентированном проектировании
25. История и назначение UML. Принципы моделирования
26. Сущности языка UML
27. Характеристика отношений языка UML
28. Классификация диаграмм языка UML
29. Диаграмма вариантов использования: назначение, сущности, отношения
30. Диаграмма классов: назначение, сущности, отношения

ОРГАНИЗАЦИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Обучение дисциплине «Технологии программирования» предполагает реализацию следующие формы самостоятельной работы студентов:

- изучение печатных источников по теме дисциплины;
- изучение профессиональных электронных ресурсов по теме дисциплины;
- подготовку к аудиторному выполнению лабораторных работ (предварительное знакомство с методическими указаниями, программным обеспечением, вариантом индивидуального задания по работе);
- подготовку к защите лабораторных работ (оформление отчета по индивидуальному варианту задания, защита результатов работы и демонстрации степени освоения навыков и умений по конкретной теме);
- изучение основной, дополнительной литературы в процессе подготовки к анализу и решению проблемных задач;
- углубленное изучение отдельных тем учебной дисциплины для подготовки к устным вопросам;
- подготовку к контрольному тестированию;
- систематизация полученных знаний при подготовке к зачету.

Условия для самостоятельной работы студентов, в частности, для развития навыков самоконтроля, способствующих интенсификации образовательного процесса, обеспечиваются:

- наличием и использованием в образовательном процессе учебного курса по дисциплине «Технологии программирования» в системе дистанционного обучения Moodle для доступа студентов к электронным вариантам курса лекций и учебно-методических указаний по основным разделам дисциплины, для организации учебной деятельности студентов и контроля ее результативности.

Дополнительное учебно-методическое обеспечение самостоятельной работы студентов очной формы обучения

1. Учебный курс по дисциплине «Технологии программирования» в системе дистанционного обучения Moodle (ссылка <https://moodle.psu.by/course/view.php?id=262>).

2. Методические указания к выполнению лабораторных работ по дисциплине «Технологии программирования» для студентов специальности 6-05-0533-12 «Кибербезопасность».

Содержание самостоятельной работы студентов

Вид самостоятельной работы	Тематическое содержание и используемые источники	Количество часов
1	2	3
Углубленное изучение отдельных тем для подготовки к блиц-опросам	<i>Тема 1.1 Основные понятия технологии программирования. Этапы её развития.</i> Осн. лит.: [1], [2], [3]. Доп. лит.: [1], [2].	3
	<i>Тема 1.2 Понятия и основные этапы жизненного цикла ПО.</i> Осн. лит.: [1], [3]. Доп. лит.: [1], [2].	3
	<i>Тема 1.3 Модели жизненного цикла ПО.</i> Осн. лит.: [1]. Доп. лит.: [2].	3
	<i>Тема 1.4 Управление требованиями к ПО.</i> Осн. лит.: [1]. Доп. лит.: [2].	3
	<i>Тема 2.1 Структурный подход к разработке ПО.</i> Осн. лит.: [1], [2]. Доп. лит.: [2].	3
	<i>Тема 2.2 Функциональное моделирование.</i> Осн. лит.: [1], [2], [3]. Доп. лит.: [2].	3
	<i>Тема 2.3 Моделирование потоков данных.</i> Осн. лит.: [1], [2]. Доп. лит.: [2].	3
	<i>Тема 2.4 Методология моделирования процессов IDEF3.</i> Осн. лит.: [3]. Доп. лит.: [1], [2].	3
	<i>Тема 3.1 Объектно-ориентированный подход к разработке ПО.</i> Осн. лит.: [1], [2], [3]. Доп. лит.: [1], [2].	3
	<i>Тема 3.2 Унифицированный язык моделирования UML.</i> Осн. лит.: [1], [3]. Доп. лит.: [2].	3
	<i>Тема 3.3 Диаграмма вариантов использования.</i> Осн. лит.: [1], [3]. Доп. лит.: [2].	3
	<i>Тема 3.4 Диаграмма классов.</i> Осн. лит.: [1], [3]. Доп. лит.: [2].	3
Подготовка к защите отчетов по лабораторным работам	Лабораторная работа №1 Изучение предметной области.	6
	Лабораторная работа №2 Систематизация и формализация требований к ПО.	6
	Лабораторная работа №3 Построение функциональной модели системы.	8
	Лабораторная работа №4 Построение диаграммы потоков данных.	6
	Лабораторная работа №5 Построение диаграммы IDEF3.	8
	Лабораторная работа №6 Построение диаграммы вариантов использования.	6
	Лабораторная работа №7 Построение диаграммы классов.	8
Подготовка к контрольному тестированию №1		4
Подготовка к контрольному тестированию №2		4
Подготовка к контрольному тестированию №3		4
Подготовка к контрольному тестированию №4		4
Подготовка к контрольному тестированию №5		4
Подготовка к контрольному тестированию №6		4
		108

КОНТРОЛЬ КАЧЕСТВА УСВОЕНИЯ ЗНАНИЙ

Контроль качества усвоения знаний проводится в соответствии с Положением о рейтинговой системе оценки знаний и компетенций студентов (приказ ректора УО ПГУ № 294 от 06.06.2014 (в редакции, утверждённой приказом № 605 от 17.11.2014) в форме текущего контроля и промежуточной аттестации.

Мероприятия текущего контроля проводятся в течение семестра и включают в себя следующие формы контроля:

- устная форма (блиц-опрос на лекциях);
- письменная форма (тесты, письменные отчёты по лабораторным работам);
- устно-письменная форма (отчёты по лабораторным работам с их устной защитой);
- техническая форма (электронные тесты).

Лабораторные работы предполагают выполнение и защиту. При их выполнении выдаётся индивидуальное задание. Отчёт по лабораторной работе представляется в электронном виде. Содержание отчёта: название работы, вариант задания, анализ задания, ход выполнения работы, основные и промежуточные результаты, выводы по работе. Защита работ проводится индивидуально и оценивается в соответствии с установленными правилами.

Результат текущего контроля за семестр оценивается отметкой в баллах по десятибалльной шкале и выводится, исходя из отметок, выставленных в ходе проведения мероприятий текущего контроля в течение семестра по следующей формуле:

$$T = \frac{(KT_1 + \dots + KT_{n1}) + (LP_1 + \dots + LP_{n2}) + (YO_1 + \dots + YO_{n3})}{(n1 + n2 + n3)},$$

где $KT_1, \dots, KT_{n1}$ – отметки, выставленные по результатам контрольного тестирования; $n1$ – количество тестов; $LP_1, \dots, LP_{n2}$ – отметки, выставленные по результатам защит лабораторных работ; $n2$ – количество работ; $YO_1, \dots, YO_{n3}$ – отметки, выставленные по результатам устных опросов на лекциях; $n3$ – количество устных опросов.

Результат текущего контроля рассчитывается как округлённое среднее значение. Результат может быть увеличен в соответствии с п.п. 6.8 и 6.9 Положения.

Промежуточная аттестация проводится в форме зачёта в первом семестре.

Зачёт проводится согласно Положению.

Заключение о зачёте формируется по формуле:

$$Z = k \cdot T,$$

где k – весовой коэффициент текущего контроля; T – результат текущего контроля за семестр. Весовой коэффициент k принимается равным 1.

Если полученная отметка $Z < 4$ баллов, то проводится устный зачёт отдельно по представленным в программе вопросам.

Перевод отметки по зачету осуществляется по следующим правилам: отметка «зачтено» выставляется студентам, получившим от 4 до 10 баллов, отметка «не зачтено» выставляется студентам, получившим от 1 до 3 баллов.

ХАРАКТЕРИСТИКА ИННОВАЦИОННЫХ ПОДХОДОВ К ПРЕПОДАВАНИЮ УЧЕБНОЙ ДИСЦИПЛИНЫ

Основные методы (технологии) обучения, отвечающие целям и задачам учебной дисциплины:

- проблемное обучение (проблемное изложение, вариативное изложение), реализуемое на лекционных занятиях;
- учебно-исследовательская деятельность, реализация творческого подхода, реализуемые на лабораторных занятиях;
- проектные технологии.

Используемые технологии обучения и диагностики компетенций в преподавании дисциплины «Технологии программирования» реализуют подход, основанный на максимально возможном использовании внутренней и учебной мотивации студента, проявляющейся в четком понимании им значимости всех видов выполняемых работ, как с точки зрения важности для профессиональной подготовки, так и с точки зрения оценивания. Подход предполагает использование элементов проблемного обучения и элементов исследовательской деятельности студентов в процессе аудиторной работы, а также при выполнении самостоятельных работ при постоянном рейтинговом контроле.

На лекционных занятиях по дисциплине «Технологии программирования» возможно использование элементов проблемного обучения: проблемное изложение некоторых аспектов, использование частично-поискового метода.

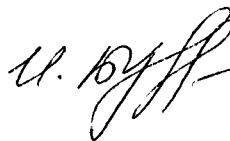
Изучение учебной дисциплины осуществляется на лекционных и лабораторных занятиях. На лекционных занятиях студенты овладевают знаниями и навыками в области проектирования и разработки ПО. В ходе лекционного изложения теоретических сведений используются традиционные словесные приемы и методы, которые активизируются постановкой проблемных вопросов и заданий, организацией учебных дискуссий в опоре на имеющуюся начальную подготовку студентов и их политехнический кругозор, использованием интерактивных методов обучения.

На лабораторных занятиях развиваются и формируются необходимые практические умения и навыки по разработке ПО средствами CASE-систем. Во время проведения лабораторных работ особое внимание уделяется формированию у студентов умения планировать работу, определять эффективную последовательность ее выполнения.

**ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ
С ДРУГИМИ УЧЕБНЫМИ ДИСЦИПЛИНАМИ СПЕЦИАЛЬНОСТИ**

Название дисциплины, по которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы по изучаемой учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу
Базы данных	МиКБ	Предложено и согласовано	
Технология разработки программного обеспечения	МиКБ	Предложено и согласовано	

Заведующий кафедрой математики и
компьютерной безопасности, к.т.н., доцент



И. Б. Бураченко