Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 26.06.2024 16:27:04
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735c18b1d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«Московский политехнический университет»

УТВЕРЖДЕНО

Декан факультета

Информационных технологий

_____ / Демидов Д.Г. /

«15» февраля 2024 г.

Рабочая программа дисциплины

«ПРОГРАММИРОВАНИЕ В СИСТЕМАХ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ»

Направление подготовки:

09.03.01 Информатика и вычислительная техника

Образовательная программа (профиль):

«Разработка инженерного программного обеспечения»

Год начала обучения:

2024

Уровень образования:

бакалавриат

Квалификация (степень) выпускника:

Бакалавр

Форма обучения:

очная

Москва, 2024

Разработчик(и):

к.т.н., доцент кафедры «СМАРТ технологии»  / А.В. Джунковский /

Согласовано:
Заведующий кафедрой «СМАРТ технологии»,
к.т.н., доцент  / Е.В. Петрунина /

1 Цели, задачи и планируемые результаты обучения по дисциплине

К **основным целям** освоения дисциплины относится:

- овладение общей методикой разработки прикладного программного обеспечения;
- получение знаний и умений разработки приложений для программного создания чертежей, деталей и сборок в Компас 3D и nanoCAD;
- закрепление получаемых в семестре знаний и навыков на практике;
- формирование взаимосвязей, получаемых в семестре знаний и навыков с изученными ранее и изучаемых параллельно с данной дисциплиной;
- подготовка студентов к деятельности в соответствии с квалификационной характеристикой бакалавра.

К **основным задачам** дисциплины относятся:

- овладение навыками и приемами прикладного программирования в САПР;
- изучение и освоение теоретического материала, как в процессе контактной, так и в ходе самостоятельной работы;
- выполнение предоставленных практических заданий различных форм, как в процессе контактной, так и в ходе самостоятельной работы;
- самостоятельная работа над тематикой дисциплины для формирования компетенций основной образовательной программы (далее, ООП).

Обучение по дисциплине «Программирование в системах автоматизированного проектирования» направлено на формирование у обучающихся следующих компетенций:

Код и наименование компетенций	**Индикаторы достижения компетенции**
ПК-1. Способен разрабатывать требования и проектировать программное обеспечение.	**ИПК-1.1 Знает:** ● Возможности существующей программно-технической архитектуры ● Возможности современных и перспективных средств разработки программных продуктов, технических средств ● Методологии разработки программного обеспечения и технологии программирования ● Методологии и технологии проектирования и использования баз данных ● Языки формализации функциональных спецификаций ● Методы и приемы формализации задач ● Методы и средства проектирования программного обеспечения ● Методы и средства проектирования программных интерфейсов ● Методы и средства проектирования баз данных ● Принципы построения и виды архитектуры компьютерного программного обеспечения; ● Типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения ● Нормативно-технические документы (стандарты), определяющие требования к технической документации на компьютерное программное обеспечение. **ИПК-1.2 Умеет:**

	● Проводить сбор и систематизацию требований к компьютерному программному обеспечению ● Выявлять взаимосвязи и документировать требования к компьютерному программному обеспечению; ● Проводить анализ исполнения требований к компьютерному программному обеспечению; ● вырабатывать варианты реализации требований к компьютерному программному обеспечению; ● проводить оценку и обоснование рекомендуемых решений; ● осуществлять коммуникации с заинтересованными; сторонами; ● выбирать средства реализации требований к компьютерному программному обеспечению; ● выбирать средства реализации требований к компьютерному программному обеспечению; ● использовать существующие типовые решения и шаблоны проектирования компьютерного программного обеспечения; ● применять методы и средства проектирования компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов; ● использовать командные средства разработки компьютерного программного обеспечения; ● применять существующие стандарты для разработки технической документации на компьютерное программное обеспечение. **ИПК-1.3 Владеет:** ● инструментами и технологиями разработки требований и проектирования программного обеспечения; ● инструментами и технологиями разработки программного кода.
ПК-5 Способен проектировать и разрабатывать программные решения в области систем автоматизированного проектирования и другого инженерного программного обеспечения.	**ИПК-5.1 Знает:** ● архитектуру и особенности разработки САПР, геометрических ядер и другого инженерного программного обеспечения; **ИПК-5.2 Умеет:** ● проектировать программные решения в области САПР, инженерного программного обеспечения.

2 Место дисциплины в структуре образовательной программы

Дисциплина относится к части, формируемой участниками образовательных отношений, модуль «Разработка инженерного программного обеспечения».

Дисциплина взаимосвязана логически и содержательно-методически со следующими дисциплинами и практиками ООП:

- Основы программирования;
- Базы данных
- Численные методы и программирование
- Информационная безопасность

3 Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 8 зачетных единиц (288 часов).

3.1 Виды учебной работы и трудоемкость

3.1.1 Очная форма обучения

№ п/п	Вид учебной работы	Количество часов	Семестры	
			2	3
1	**Аудиторные занятия**	**126**	54	72
	В том числе:			
1.1	Лекции	18	6	12
1.2	Семинарские/практические занятия			
1.3	Лабораторные занятия	108	48	60
2	**Самостоятельная работа**	**162**	90	72
	В том числе:			
2.1	Подготовка к практическим занятиям	140	80	60
2.2	Тестирование	22	10	12
3	**Промежуточная аттестация**			
	Зачет/диф.зачет/экзамен		зачет	экзамен
	Итого:	**288/8**	144/4	144/4

3.2 Тематический план изучения дисциплины

№ п/п	Разделы/темы дисциплины	Трудоемкость, час					
		Всего	Аудиторная работа				Самостоятельная работа
			Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	
1	Создание проекта MFC Dialog Based Application. Классы диалога, приложения и их основные методы		2				8
2	Создание проекта MFC Dialog Based Application. Переменные контролов GUI. Метод UpdateData()						8
3	Создание проекта MFC Dialog Based Application. Основные контролы GUI и их использование.				2		8
4	Основы работы с API Компас- 3D. Виды API. Подключение к Компас-		2		6		8

	3D. Создание чертежа. Формат чертежа. Основная надпись. Виды. Текстовая надпись. Технические требования. Таблицы.						
5	Основы работы с API Компас- 3D. Графические примитивы.				6		8
6	Основы работы с API Компас- 3D. Размеры и штриховка.				4		8
7	Основы работы с API Компас- 3D. Обозначение баз. Шероховатость.				4		8
8	Основы работы с API Компас- 3D. Спецификации.				4		8
9	Основы работы с API nanoCAD. Виды API. Создание и загрузка приложения. Пользовательские команды.		2		8		10
10	Основы работы с API nanoCAD. Создание пользовательского графического примитива.				8		8
11	Основы работы с API nanoCAD. Создание реакторов.				6		8
	Итого во 2-м семестре		**6**		**48**		**90**
12	Создание проекта MFC SDI Application. Архитектура Документ – вид.		2		4		4
13	Создание проекта MFC SDI Application. Архитектура Документ – вид. Использование дополнительного вида для документа		2		4		8
14	Создание проекта MFC SDI Application. Архитектура Документ – вид. Работа с главным и контекстным меню		1		4		8
15	Создание проекта MFC SDI Application. Архитектура Документ – вид. Работа с модальными и немодальными диалогами		1		4		8
16	Основы работы с API Компас- 3D. Документ, содержащий 3D-модель (сборку). Иерархия интерфейсов в 3D.		2		4		8
17	Основы работы с API Компас- 3D. Эскиз. Вспомогательная геометрия. Твердотельные построения.				8		8

18	Основы работы с API Компас- 3D. Фаски, скругления, массивы				8		8
19	Основы работы с API Компас- 3D. Параметризованная 3D-модель.				4		8
20	Основы работы с API Компас- 3D. Создание сборки.		4		20		8
Итого в 3-м семестре			**12**		**60**		**72**
Итого			**18**		**108**		**162**

3.3 Содержание дисциплины

Раздел 1. Создание проекта MFC Dialog Based Application. Классы диалога, приложения и их основные методы

- изучение пользовательского интерфейса MS Visual Studio и возможностей его конфигурации
- изучение возможностей «мастера приложений» MFC MS Visual Studio.
- создание приложения Windows с Dialog Based интерфейсом на языке C++ с помощью «мастера приложений»
- изучение классов CWinApp и CDialog
- изучение возможностей работы с ресурсами приложения

Раздел 2. Создание проекта MFC Dialog Based Application. Переменные контролов GUI. Метод UpdateData()

- изучение методов взаимодействия с контролами GUI с использованием переменных и без них
- изучение работы метода CDialog::UpdateData()

Раздел 3. Создание проекта MFC Dialog Based Application. Основные контролы GUI и их использование.

- изучение применения основных контролов GUI: кнопки, radiobutton, checkbox, combobox, treectrl, listctrl, etditctrl и т.д.

Раздел 4. Основы работы с API Компас- 3D. Виды API. Подключение к Компас- 3D. Создание чертежа. Формат чертежа. Основная надпись. Виды. Текстовая надпись. Технические требования. Таблицы.

- основы COM-технологии
- основные классы API Компас
- применение API для создания, конфигурирования и оформления чертежа.

Раздел 5. Основы работы с API Компас- 3D. Графические примитивы.

- использование API Компас для построения графических примитивов

Раздел 6. Основы работы с API Компас- 3D. Размеры и штриховка.

- использование API Компас для построения размеров и штриховок

Раздел 7. Основы работы с API Компас- 3D. Обозначение баз. Шероховатость.

- использование API Компас для обозначения баз и параметров шероховатостей.

Раздел 8. Основы работы с API Компас- 3D. Спецификации.

- использование API Компас для работы со спецификацией.

Раздел 9. Основы работы с API nanoCAD. Виды API. Создание и загрузка приложения. Пользовательские команды.

- виды API nanoCAD
- создание пользовательской команды

Раздел 10. Основы работы с API nanoCAD. Создание пользовательского графического примитива.

- класс NcDbEntity и наследные от него
- реализация пользовательского примитива CrossCircle

Раздел 11. Основы работы с API nanoCAD. Создание реакторов.

- виды реакторов
- реализация пользовательского реактора

Раздел 12. Создание проекта MFC SDI Application. Архитектура Документ – вид.

- архитектура Document-View основные понятия и методы

Раздел 13. Создание проекта MFC SDI Application. Архитектура Документ – вид. Использование дополнительного вида для документа

- использование CSplitterWnd
- реализация пользовательского вида для документа

Раздел 14. Создание проекта MFC SDI Application. Архитектура Документ – вид. Работа с главным и контекстным меню

- класс CMenu его использование для реализации разных видов ользлвателских меню

Раздел 15. Создание проекта MFC SDI Application. Архитектура Документ – вид. Работа с модальными и немодальными диалогами

- Понятие модальных и немодальных диалогов. Специфика их реализации

Раздел 16. Основы работы с API Компас- 3D. Документ, содержащий 3D-модель (сборку). Иерархия интерфейсов в 3D.

- создание трехмерных документов в API Компас-3D

Раздел 17. Основы работы с API Компас- 3D. Эскиз. Вспомогательная геометрия. Твердотельные построения.

- построение эскизов в координатных и произвольных плоскостях
- операции твердотельного моделирования

Раздел 18. Основы работы с API Компас- 3D. Фаски, скругления, массивы

- построение фасок и скруглений
- идентификация объектов для построений фасок и скруглений

Раздел 19. Основы работы с API Компас- 3D. Параметризованная 3D-модель.

- методы создания параметризированной 3D модели

Раздел 20. Основы работы с API Компас- 3D. Создание сборки.

- создание документа «сборка», загрузка деталей в него.
- управление положением деталей в сборке
- наложение ограничений на детали в сборке

3.4 Тематика семинарских/практических и лабораторных занятий

Лабораторные занятия

Лабораторная работа №1. Создание проекта MFC Dialog Based Application. Классы диалога, приложения и их основные методы

Лабораторная работа № 2. Создание проекта MFC Dialog Based Application. Переменные контролов GUI. Метод UpdateData()

Лабораторная работа № 3. Создание проекта MFC Dialog Based Application. Основные контролы GUI и их использование.

Лабораторная работа № 4. Основы работы с API Компас- 3D. Виды API. Подключение к Компас- 3D. Создание чертежа. Формат чертежа. Основная надпись. Виды. Текстовая надпись. Технические требования. Таблицы.

Лабораторная работа № 5. Основы работы с API Компас- 3D. Графические примитивы.

Лабораторная работа № 6. Основы работы с API Компас- 3D. Размеры и штриховка.

Лабораторная работа № 7. Основы работы с API Компас- 3D. Обозначение баз. Шероховатость.

Лабораторная работа № 8. Основы работы с API Компас- 3D. Спецификации.

Лабораторная работа № 9. Основы работы с API nanoCAD. Виды API. Создание и загрузка приложения. Пользовательские команды.

Лабораторная работа № 10. Основы работы с API nanoCAD. Создание пользовательского графического примитива.

Лабораторная работа № 11. Основы работы с API nanoCAD. Создание реакторов.

Лабораторная работа № 12. Создание проекта MFC SDI Application. Архитектура Документ – вид.

Лабораторная работа № 13. Создание проекта MFC SDI Application. Архитектура Документ – вид. Использование дополнительного вида для документа

Лабораторная работа № 14. Создание проекта MFC SDI Application. Архитектура Документ – вид. Работа с главным и контекстным меню

Лабораторная работа № 15. Создание проекта MFC SDI Application. Архитектура Документ – вид. Работа с модальными и немодальными диалогами

Лабораторная работа № 16. Основы работы с API Компас- 3D. Документ, содержащий 3D-модель (сборку). Иерархия интерфейсов в 3D.

Лабораторная работа № 17. Основы работы с API Компас- 3D. Эскиз. Вспомогательная геометрия. Твердотельные построения.

Лабораторная работа № 18. Основы работы с API Компас- 3D. Фаски, скругления, массивы

Лабораторная работа № 19. Основы работы с API Компас- 3D. Параметризованная 3D-модель.

Лабораторная работа № 20. Основы работы с API Компас- 3D. Создание сборки.

3.5 Тематика курсовых работ

1. Разработать программный конфигуратор в среде Компас 3D для изделия «Корпус большой»
2. Разработать программный конфигуратор в среде Компас 3D для изделия «Корпус малый»
3. Разработать программный конфигуратор в среде Компас 3D для изделия «Шестерня косозубая»
4. Разработать программный конфигуратор в среде Компас 3D для изделия «Петля»
5. Разработать программный конфигуратор в среде Компас 3D для изделия «Крышка масляного бака»
6. Разработать программный конфигуратор в среде Компас 3D для изделия «Поворотный стол»
7. Разработать программный конфигуратор в среде Компас 3D для изделия «Крепление большое»
8. Разработать программный конфигуратор в среде Компас 3D для изделия «Радиатор»
9. Разработать программный конфигуратор в среде Компас 3D для изделия «Шестерня большая»
10. Разработать программный конфигуратор в среде Компас 3D для изделия «Шестерня малая»
11. Разработать программный конфигуратор в среде Компас 3D для изделия «Крепление большое»
12. Разработать программный конфигуратор в среде Компас 3D для изделия «Крепление малое»
13. Разработать программный конфигуратор в среде Компас 3D для изделия «Крепление фасонное»
14. Разработать программный конфигуратор в среде Компас 3D для изделия «Хомут»
15. Разработать программный конфигуратор в среде Компас 3D для изделия «Ручка большая»
16. Разработать программный конфигуратор в среде Компас 3D для изделия «Ручка малая»
17. Разработать программный конфигуратор в среде Компас 3D для изделия «Червячное колесо»
18. Разработать программный конфигуратор в среде Компас 3D для изделия «Червячный вал»
19. Разработать программный конфигуратор в среде Компас 3D для изделия «Насосная камера»
20. В Разработать программный конфигуратор в среде Компас 3D для изделия «Вал-червяк»
21. Разработать программный конфигуратор в среде Компас 3D для изделия «Корпус привода»
22. Разработать программный конфигуратор в среде Компас 3D для изделия «Корпус узла»
23. Разработать программный конфигуратор в среде Компас 3D для изделия «Уголок»
24. Разработать программный конфигуратор в среде Компас 3D для изделия «Шестерня двойная»
25. Разработать программный конфигуратор в среде Компас 3D для изделия «Вставка в катушку»
26. Разработать программный конфигуратор в среде Компас 3D для изделия «Защелка»
27. Разработать программный конфигуратор в среде Компас 3D для изделия «Заглушка»
28. Разработать программный конфигуратор в среде Компас 3D для изделия «Втулка»

29. Разработать программный конфигуратор в среде Компас 3D для изделия «Крепление для кулеров»
30. Разработать программный конфигуратор в среде Компас 3D для изделия «Крепление малое»

4 Учебно-методическое и информационное обеспечение

4.1 Основная литература

1. Лянг, В. Ф. Программирование в САПР. Пространственное моделирование аппарата воздушного охлаждения в среде Autodesk Inventor : учебное пособие / В.Ф. Лянг. — Москва : ИНФРА-М, 2022. — 476 с. — (Высшее образование: Бакалавриат). — DOI 10.12737/991757. - ISBN 978-5-16-014572-3. - Текст : электронный. - URL: https://znanium.com/catalog/product/991757 (дата обращения: 28.06.2023). – Режим доступа: по подписке.
2. Лянг, В. Ф. Программирование в САПР: пространственное моделирование колонного аппарата в среде Autodesk Inventor : учебное пособие / В.Ф. Лянг. — Москва : ИНФРА-М, 2021. — 249 с. + Доп. материалы [Электронный ресурс]. — (Высшее образование: Бакалавриат). — DOI 10.12737/991773. - ISBN 978-5-16-014573-0. - Текст : электронный. - URL: https://znanium.com/catalog/product/991773 (дата обращения: 28.06.2023). – Режим доступа: по подписке.
3. Затонский, А. В. Программирование и основы алгоритмизации. Теоретические основы и примеры реализации численных методов: учебное пособие / А.В. Затонский, Н.В. Бильфельд. — 2-е изд. — Москва : РИОР : ИНФРА-М, 2020. — 167 с. — (Высшее образование). — DOI: https: //www.dx.doi.org/10.12737/20468. - ISBN 978-5-369-01195-9. - Текст : электронный. - URL: https://znanium.com/catalog/product/1077389 (дата обращения: 28.06.2023). – Режим доступа: по подписке.
4. Павловская, Т. А. C/C++. Структурное и объектно-ориентированное программирование : практикум / Т. А. Павловская, Ю. А. Щупак. - Санкт-Петербург : Питер, 2021. - 352 с. - (Серия «Учебное пособие»). - ISBN 978-5-4461-9799-6. - Текст : электронный. - URL: https://znanium.com/catalog/product/1857042 (дата обращения: 28.06.2023). – Режим доступа: по подписке.
5. Немцова, Т. И. Программирование на языке высокого уровня. Программирование на языке C++ : учебное пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев ; под ред. Л.Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2021. — 512 с. + Доп. материалы [Электронный ресурс]. — (Среднее профессиональное образование). - ISBN 978-5-8199-0699-6. - Текст : электронный. - URL: https://znanium.com/catalog/product/1172261 (дата обращения: 28.06.2023). – Режим доступа: по подписке.

4.2 Дополнительная литература

6. С.Н. Норсеев. Разработка приложений под компас в DELPHI. https://norseev.ru/download/develop-kompas-applications-on-delphi/?wpdmdl=1994&refresh=627d60e4181c41652383972
7. Диков, А.В. Математическое моделирование и численные методы : учебное пособие / А.В. Диков, С.В. Степанова ; ред. Г.В. Сугробов. – Пенза : Пензенский государственный педагогический университет (ПГПУ), 2000. – 162 с. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=96973 (дата обращения: 28.06.2023). – Текст : электронный.
8. Дэвид Дж. Круглински, Скотт Уингоу, Дж. Шеферд. Программирование на Microsoft Visual C++ 6.0 для профессионалов. М.: Русская редакция, 2004 г.-861 с.
9. Диков, А.В. Математическое моделирование и численные методы : учебное пособие / А.В. Диков, С.В. Степанова ; ред. Г.В. Сугробов. – Пенза : Пензенский

государственный педагогический университет (ПГПУ), 2000. – 162 с. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=96973 (дата обращения: 28.06.2023). – Текст : электронный.
10. Дэвид Дж. Круглински, Скотт Уингоу, Дж. Шеферд. Программирование на Microsoft Visual C++ 6.0 для профессионалов. М.: Русская редакция, 2004 г.-861 с.
11. Александров, Э.Э. Программирование на языке C в Microsoft Visual Studio 2010 : учебное пособие / Э.Э. Александров, В.В. Афонин ; Национальный Открытый Университет "ИНТУИТ". – Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), 2010. – 500 с. : ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=233564 (дата обращения: 28.06.2023). – Текст : электронный.
12. Owen R. Inventor Programming in C++ [Электронный ресурс]: – Электрон. учеб. – 2019. – режим доступа к учеб.: http://www.ransensoftware.com/Inventor-CPP/

Интернет ресурсы

13. https://docs.microsoft.com/ru-ru/cpp/?view=vs-2019
14. http://cppstudio.com/

4.3 Электронные образовательные ресурсы

«Программирование в САПР » https://online.mospolytech.ru/course/view.php?id=1204

«Программирование в САПР-2» https://online.mospolytech.ru/course/view.php?id=8645

4.4 Лицензионное и свободно распространяемое программное обеспечение

Microsoft Visual Studio Community Edition с установленным компилятором C++ и установленной библиотекой MFC

САПР Компас-3D версии 21 и выше

САПР nanoCAD версии 23 и выше

4.5 Современные профессиональные базы данных и информационные справочные системы

1. Справочно-правовая системы «КонсультантПлюс: Некоммерческая интернет-версия»
https://www.consultant.ru/online/
2. Официальный интернет-портал правовой информации http://pravo.gov.ru
3. Российская национальная библиотека http://www.nlr.ru
4. ЭБС «Университетская библиотека онлайн» https://biblioclub.ru/index.php
5. Единое окно доступа к образовательным ресурсам Федеральный портал http://window.edu.ru
6. Научная электронная библиотека http://www.elibrary.ru
7. Российская государственная библиотека http://www.rsl.ru

5 Материально-техническое обеспечение

1. Компьютерные классы с оснащением: столы, стулья, аудиторная доска, использование переносного мультимедийного комплекса (переносной проектор, персональный ноутбук).
1. Персональные компьютеры, мониторы, мышки, клавиатуры. Рабочее место преподавателя: стол, стул.
2. Аудитория для самостоятельной работы.
3. Библиотека, читальный зал.

6 Методические рекомендации

6.1 Методические рекомендации для преподавателя по организации обучения

1. При подготовке к занятиям следует предварительно проработать материал занятия, предусмотрев его подачу точно в отведенное для этого время занятия. Следует подготовить необходимые материалы – теоретические сведения, задачи и др. При проведении занятия следует контролировать подачу материала и решение заданий с учетом учебного времени, отведенного для занятия.

2. При проверке работ и отчетов следует учитывать не только правильность выполнения заданий, но и оптимальность выбранных методов решения, правильность выполнения всех его шагов.

3. При организации и проведения экзаменов в практико-ориентированной форме следует использовать утвержденные кафедрой Методические рекомендации.

6.2 Методические указания для обучающихся по освоению дисциплины

Получение углубленных знаний по дисциплине достигается за счет активной самостоятельной работы обучающихся. Выделяемые часы целесообразно использовать для знакомства с учебной и научной литературой по проблемам дисциплины, анализа научных концепций. Промежуточная аттестация обучающихся в форме зачёта проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине (модулю), при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине проводится преподавателем, ведущим занятия по дисциплине методом экспертной оценки. По итогам промежуточной аттестации по дисциплине выставляется оценка «отлично», «хорошо», «удовлетворительно» или «не удовлетворительно». К промежуточной аттестации допускаются только обучающиеся, выполнившие все виды учебной работы, предусмотренные рабочей программой по дисциплине «Программирование в САПР».

7 Фонд оценочных средств

Код и наименование компетенций	Наименование компетенций	Индикаторы достижения компетенции	Форма контроля	Этапы формирования (разделы дисциплины)

ПК-1	ПК-1. Способен разрабатывать требования и проектировать программное обеспечение.	**ИПК-1.1 Знает:** ● Возможности существующей программно-технической архитектуры ● Возможности современных и перспективных средств разработки программных продуктов, технических средств ● Методологии разработки программного обеспечения и технологии программирования ● Методологии и технологии проектирования и использования баз данных ● Языки формализации функциональных спецификаций ● Методы и приемы формализации задач ● Методы и средства проектирования программного обеспечения ● Методы и средства проектирования программных интерфейсов ● Методы и средства проектирования баз данных ● Принципы построения и виды архитектуры компьютерного программного обеспечения; ● Типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения ● Нормативно-технические документы (стандарты), определяющие требования к технической документации на компьютерное программное обеспечение.	Промежуточный контроль: экзамен Текущий контроль: устная защита лабораторных работ, тестирование.	Разделы 1-11

		ИПК-1.2 Умеет: ● Проводить сбор и систематизацию требований к компьютерному программному обеспечению ● Выявлять взаимосвязи и документировать требования к компьютерному программному обеспечению; ● Проводить анализ исполнения требований к компьютерному программному обеспечению; ● вырабатывать варианты реализации требований к компьютерному программному обеспечению; ● проводить оценку и обоснование рекомендуемых решений; ● осуществлять коммуникации с заинтересованными; сторонами; ● выбирать средства реализации требований к компьютерному программному обеспечению; ● выбирать средства реализации требований к компьютерному программному обеспечению; ● использовать существующие типовые решения и шаблоны проектирования компьютерного программного обеспечения; ● применять методы и средства проектирования		

		компьютерного программного обеспечения, структур данных, баз данных, программных интерфейсов; ● использовать командные средства разработки компьютерного программного обеспечения; ● применять существующие стандарты для разработки технической документации на компьютерное программное обеспечение. **ИПК-1.3 Владеет:** ● инструментами и технологиями разработки требований и проектирования программного обеспечения; ● инструментами и технологиями разработки программного кода.		
ПК-5	Способен проектировать и разрабатывать программные решения в области систем автоматизированного проектирования и другого инженерного программного обеспечения.	**ИПК-5.1 Знает:** ● архитектуру и особенности разработки САПР, геометрических ядер и другого инженерного программного обеспечения; **ИПК-5.2 Умеет:** ● проектировать программные решения в области САПР, инженерного программного обеспечения.	Промежуточный контроль: экзамен Текущий контроль: устная защита лабораторных работ, тестирование.	Разделы 12-20

7.1 Методы контроля и оценивания результатов обучения

7.2 Шкала и критерии оценивания результатов обучения

Критерии оценки ответа на зачете

«зачтено»: обучающийся демонстрирует системные теоретические знания, практические навыки, владеет терминами, делает аргументированные выводы и обобщения, приводит примеры, показывает свободное владение монологической речью и способность быстро реагировать на уточняющие вопросы.

обучающийся демонстрирует прочные теоретические знания, практические навыки, владеет терминами, делает аргументированные выводы и обобщения, приводит примеры, показывает свободное владение монологической речью, но при этом делает несущественные ошибки, которые быстро исправляет самостоятельно или при незначительной коррекции преподавателем.

обучающийся демонстрирует неглубокие теоретические знания, проявляет слабо сформированные навыки анализа явлений и процессов, недостаточное умение делать аргументированные выводы и приводить примеры, показывает не достаточно свободное владение монологической речью, терминами, логичностью и последовательностью изложения, делает ошибки, которые может исправить только при коррекции преподавателем.

«не зачтено»: обучающийся демонстрирует незнание теоретических основ предмета, отсутствие практических навыков, не умеет делать аргументированные выводы и приводить примеры, показывает слабое владение монологической речью, не владеет терминами, проявляет отсутствие логичности и последовательности изложения, делает ошибки, которые не может исправить даже при коррекции преподавателем, отказывается отвечать на дополнительные вопросы

Критерии оценки ответа на экзамене

«отлично»: обучающийся демонстрирует системные теоретические знания, практические навыки, владеет терминами, делает аргументированные выводы и обобщения, приводит примеры, показывает свободное владение монологической речью и способность быстро реагировать на уточняющие вопросы.

«хорошо»: обучающийся демонстрирует прочные теоретические знания, практические навыки, владеет терминами, делает аргументированные выводы и обобщения, приводит примеры, показывает свободное владение монологической речью, но при этом делает несущественные ошибки, которые быстро исправляет самостоятельно или при незначительной коррекции преподавателем.

«удовлетворительно»: обучающийся демонстрирует неглубокие теоретические знания, проявляет слабо сформированные навыки анализа явлений и процессов, недостаточное умение делать аргументированные выводы и приводить примеры, показывает не достаточно свободное владение монологической речью, терминами, логичностью и последовательностью изложения, делает ошибки, которые может исправить только при коррекции преподавателем.

«не удовлетворительно»: обучающийся демонстрирует незнание теоретических основ предмета, отсутствие практических навыков, не умеет делать аргументированные выводы и приводить примеры, показывает слабое владение монологической речью, не владеет терминами, проявляет отсутствие логичности и последовательности изложения, делает ошибки, которые не может исправить даже при коррекции преподавателем, отказывается отвечать на дополнительные вопросы

Критерии оценки работы обучающегося на лабораторных занятиях

«5» (отлично): выполнены все практические задания, предусмотренные практическими занятиями, обучающийся четко и без ошибок ответил на все контрольные вопросы, активно работал на практических занятиях.

«4» (хорошо): выполнены все практические задания, предусмотренные практическими занятиями, обучающийся с корректирующими замечаниями преподавателя ответил на все контрольные вопросы, достаточно активно работал на практических занятиях.

«3» (удовлетворительно): выполнены все практические задания, предусмотренные практическими занятиями с замечаниями преподавателя; обучающийся ответил на все контрольные вопросы с замечаниями.

«2» (неудовлетворительно): обучающийся не выполнил или выполнил неправильно практические задания, предусмотренные практическими занятиями; студент ответил на контрольные вопросы с ошибками или не ответил на контрольные вопросы.

Критерии оценки тестирования

Тестирование оценивается в соответствии с процентом правильных ответов, данных студентом на вопросы теста. Стандартная шкала соответствия результатов тестирования выставляемой балльной оценке:

− «отлично» - свыше 85% правильных ответов;

− «хорошо» - от 70,1% до 85% правильных ответов;

− «удовлетворительно» - от 55,1% до 70% правильных ответов;

− от 0 до 55% правильных ответов – «неудовлетворительно»

7.3 Оценочные средства

7.3.1 Текущий контроль на лабораторных занятиях

Пример задания текущего контроля:

::В каком классе содержатся экземпляры панели инструментов, полосы статуса::В каком классе содержатся экземпляры панели инструментов, полосы статуса{

=MainFrm

~Doc,App

~Box

~AppFrame

}

Для фиксации тулбара исользуется метод::Для фиксации тулбара исользуется метод{

=EnableDocking

~EnableFly

~EnableDock

~Dock

}

Как называется класс рписывающий контекст устройства::Как называется класс рписывающий контекст устройства{

=CDC

~pDC

~DC

~DeviceContext

}

Как расшифровывается аббревиатура SDI::Как расшифровывается аббревиатура SDI{

=Single Document Interface

~Single Direct Interface

~Separate Document Interface

~Single Document Interaction

}

Какие классы создаются при создании приложения SDI::Какие классы создаются при создании приложения SDI{

=App,Doc,View,MainFrame

~App,Doc,View,MainApp

~App,View,MainApp

~App,Doc,TreeView,MainApp

}

Каким оператором можно загрузить данных документа::Каким оператором можно загрузить данных документа{
=>>
~<<
~->
~<-
}
Какой из создаваемых классов отвечает за визуализацию данных::Какой из создаваемых классов отвечает за визуализацию данных{
~Doc
~Doc,View
=View
~App
}
Какой из создаваемых классов отвечает за хранение данных::Какой из создаваемых классов отвечает за хранение данных{
=Doc
~Doc,View
~View
~App
}
Какой из создаваемых классов является визуальным контейнером для View::Какой из создаваемых классов является визуальным контейнером для View{
=MainFrm
~Doc,App
~Box
~App
}
Какой класс отвечает за передачу араметров командной строки в пприложение::Какой класс отвечает за передачу араметров командной строки в пприложение{
=CCommandLineInfo
~CCmdLineInfo
~CCommandStringInfo
~CCommandString
}
Какой метод вызывается при создании нового документа::Какой метод вызывается при создании нового документа{
=OnNewDocument
~CreateDocument
~OnCreateDocument
~NewDocument
}
Какой метод исользуется для задания имени ветки реестра::Какой метод исользуется для задания имени ветки реестра{
=SetRegistryKey
~SetRegistriKey
~WriteRegistryKey
~PutRegistryKey
}
Какой метод класса View отвечает за отрисовку::Какой метод класса View отвечает за отрисовку{
=OnDraw
~OnBeginPainting
~OnPainting

~DrawView
}
Какой метод отвечает за загрузку данных документа::Какой метод отвечает за загрузку данных документа{
=Serialize
~Open
~Load
~OnLoad
}
Какой метод отвечает за сохранение данных документа::Какой метод отвечает за сохранение данных документа{
=Serialize
~Save
~SaveAs
~OnSerialize
}

7.3.2 Промежуточная аттестация (экзамен)

Какой метод позволяет запросить интерфейс через зная его имя::Какой метод позволяет запросить интерфейс через зная его имя{
=QueryInterface
~FindInterface
~GetInterface
}
Какой метод позволяет начать редактирование эскиза::Какой метод позволяет начать редактирование эскиза{
=BeginEdit
~Edit
~StartEdit
~Change
~Update
}
Какой метод позволяет построить отрезок в эскизе::Какой метод позволяет построить отрезок в эскизе{
=ksLineSeg
~ksLineSegment
~ksLine
~ksLineStraight
}
Какой метод позволяет создать документ-деталь::Какой метод позволяет создать документ-деталь{
=Document3D()
~PartDocument3D()
~PartDocument()
}
Какой метод позволяет узнать есть ли активный экземпляр Компас::Какой метод позволяет узнать есть ли активный экземпляр Компас{
=GetActiveObject
~GetActiveCOMServer
~GetCOMServer
~SearchCOMObject
}

Какой метод позволяет установить имя грани::Какой метод позволяет установить имя грани{
=Putname
~Setname
~name
}
Какой метод позволяет установить параметры операции вращения::Какой метод позволяет установить параметры операции вращения{
=SetSideParam
~SetRotParam
~SetRotationParam
~SetParam
}
Какой метод позволяет установить параметры фаски::Какой метод позволяет установить параметры фаски{
=SetChamferParam
~SetChamferParameter
~SetParam
~SetParams
}
Какой метод позволяет установить плоскость эскиза::Какой метод позволяет установить плоскость эскиза{
=SetPlane
~PutPlane
~SelectPlane
~ToPlane
}
Какой программный идентификатор у Компас для API5::Какой программный идентификатор у Компас для API5{
=Kompas.Application.5
~Kompas.App.5
~Kompas.Ascon.5
~Kompas.Application
}
Объект какого типа может хранить набор сечений для операции лофт::Объект какого типа может хранить набор сечений для операции лофт{
=ksEntityCollectionPtr
~ksObjectCollectionPtr
~ksLoftCollectionPtr
~ksSketchCollectionPtr
}