Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 13.10.2023 15:17:46
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735c18b1d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
Факультет информационных технологий

УТВЕРЖДАЮ
Декан факультета
информационных технологий
/Д. Г. Демидов/
"28" апреля 2022 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Анализ и реверс инжиниринг программных средств»

Направление подготовки
10.03.01 «Информационная безопасность»

Профиль/специализация
«Безопасность компьютерных систем»

Квалификация
Бакалавр

Формы обучения
Очная

Москва, 2022 г.

Разработчик(и):
Преподаватель А.А. Недогарок

Согласовано:
И.о. заведующего кафедрой «Информационная безопасность», А.Ю. Гневшев

Руководитель образовательной программы, А.Ю. Гневшев

Содержание

1 Цели, задачи и планируемые результаты обучения по дисциплине 4
знать: 4
уметь: 5
2 Место дисциплины в структуре образовательной программы 5
3 Структура и содержание дисциплины 5
3.1 Виды учебной работы и трудоемкость 5
3.2 Тематический план изучения дисциплины 6
3.3 Содержание дисциплины 6
4 Учебно-методическое и информационное обеспечение 6
4.1 Основная литература 6
4.2 Дополнительная литература 6
4.3 Электронные образовательные ресурсы 6
5 Материально-техническое обеспечение 6
6 Методические рекомендации 7
6.1 Методические рекомендации для преподавателя по организации обучения 7
6.2 Методические указания для обучающихся по освоению дисциплины 7
7 Фонд оценочных средств 7
7.1 Методы контроля и оценивания результатов обучения 7
7.2 Шкала и критерии оценивания результатов обучения 7
7.3 Оценочные средства 11
7.3.1 Список вопросов для экзамена по дисциплине. 11

1 Цели, задачи и планируемые результаты обучения по дисциплине

К **основным целям** освоения дисциплины «Анализ и реверс инжиниринг программных средств» следует отнести:

- применять на практике знания, полученные в течение курса «Анализ и реверс инжиниринг программных средств»

К **основным задачам** освоения дисциплины «Анализ и реверс инжиниринг программных средств» следует отнести:

- формирование навыков разработки программного обеспечения;
- научиться исследовать и анализировать программный код;
- изучить алгоритм и стандарты написания документации

Обучение по дисциплине «Анализ и реверс инжиниринг программных средств» направлено на формирование у обучающихся следующих компетенций:

Код и наименование компетенций	Индикаторы достижения компетенции
ОПК-7. Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности	**знать:** • реверс инжиниринг программно-аппаратных решений; • разработка программного обеспечения; • исследование и анализ программного кода; • написание отчетной и технической документации о выполненных работах. **уметь:** • применять на практике знания, полученные в области реверс инжиниринга программного кода. **владеть:** • навыками исследования и анализа программного кода и написания отчётной, технической документации.

2 Место дисциплины в структуре образовательной программы

Дисциплина «Анализ и реверс инжиниринг программных средств» относится к числу профессиональных учебных дисциплин части формируемой участниками образовательных отношений (Б1.2) основной образовательной программы (Б1.2.04).

Изучение дисциплины опирается на знания, умения и навыки, приобретенные в предшествующих дисциплинах: Языки программирования, Технологии и методы программирования.

3 Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 4 зачетных единицы, т.е. **144** академических часов (лабораторные занятия – 72 час, самостоятельная работа - 72 часов, форма контроля – экзамен) в 7 семестре.

Структура и содержание дисциплины «Анализ и реверс инжиниринг программных средств» по срокам и видам работы отражены в пунктах 3.1 и 3.2.

3.1 Виды учебной работы и трудоемкость (по формам обучения)

3.1.1 Очная форма обучения

№ п/п	Вид учебной работы	Количество часов	Семестры	
			Семестр	Неделя семестра
1	**Аудиторные занятия**	**72**	7	1-18
	В том числе:			
1.1	Лекции			
1.2	Семинарские/практические занятия			
1.3	Лабораторные занятия	72	7	1-18
2	**Самостоятельная работа**	**72**	7	1-18
3	**Промежуточная аттестация**		7	19-21
	Экзамен			
	Итого:	**144**		

3.2 Тематический план изучения дисциплины (по формам обучения)

3.2.1 Очная форма обучения

№ п/п	Разделы/темы дисциплины	Трудоемкость, час					
		Всего	Аудиторная работа				Самостоятельная работа
			Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	
1	Системное программирование: определение предметной области.	8			4		4
2	Создание системных приложений средствами операционной системы. (базовые функции API)	8			4		4
3	Низкоуровневое программирование: начальные сведения.	8			4		4
4	Программирование на языке Ассемблера.	8			4		4
5	Обработка на низком уровне различных типов данных.	8			4		4

6	Использование системных вызовов. Программирование ввода и вывода.	8			4		4
7	Структурное программирование.	8			4		4
8	Динамические структуры данных.	8			4		4
9	Объектно-ориентированное программирование.	8			4		4
10	Работа в среде визуального программирования Delphi.	8			4		4
11	Основные принципы среды.	8			4		4
12	Использование стандартных элементов управления.	8			4		4
13	Организация управления приложением.	8			4		4
14	Простейшие графические изображения и мультипликации	8			4		4
15	Системные параметры, считываемые программой.	8			4		4
16	Работа с файлами на уровне операционной системы.	8			4		4
17	Динамически присоединяемые библиотеки.	8			4		4
18	Отладка кода программных модулей на языке Ассемблера	8			4		4
Итого		**144**			**72**		**72**

4 Учебно-методическое и информационное обеспечение

4.1 Основная литература

1. Абрамов, Е.С. Машинно-ориентированное программирование : учебное пособие / Е.С. Абрамов, И.Д. Сидоров ; Министерство образования и науки РФ, Южный федеральный университет, Инженерно-технологическая академия. – Таганрог : Издательство Южного федерального университета, 2016. – 88 с. : схем., табл., ил. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=492941 (дата обращения: 19.08.2019). – ISBN 978-5-9275-2065-7. – Текст : электронный.

2. Пильщиков, В.Н. Программирование на языке ассемблера IBM PC : учебное пособие / В.Н. Пильщиков. – Москва : Диалог-МИФИ, 2014. – 288 с. : ил. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=447687 (дата обращения: 19.08.2019). – Библиогр. в кн. – ISBN 5-86404-051-7. – Текст : электронный.

4.2 Дополнительная литература

1. Флоренсов, А.Н. Системное программное обеспечение : учебное пособие / А.Н. Флоренсов ; Минобрнауки России, Омский государственный технический университет. – Омск : Издательство ОмГТУ, 2017. – 139 с. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=493301 (дата обращения: 19.08.2019). – Библиогр. в кн. – ISBN 978-5-8149-2441-4. – Текст : электронный.

4.3 Электронные образовательные ресурсы

Электронный образовательный ресурс на разработке.

5 Материально-техническое обеспечение

Для проведения всех видов занятий необходимо презентационное оборудование (мультимедийный проектор, экран) – 1 комплект.

Для проведения лабораторных занятий необходимо наличие компьютерных классов оборудованных современной вычислительной техникой из расчета одно рабочее место на одного обучаемого.

Оборудование и аппаратура:

1. Операционная система Microsoft Windows.
2. IDE с поддержкой разработки на требуемом языке.

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.

6.2 Методические указания для обучающихся по освоению дисциплины

Изучение дисциплины осуществляется в строгом соответствии с целевой установкой в тесной взаимосвязи учебным планом. Основой теоретической подготовки студентов являются лекции.

В процессе самостоятельной работы студенты закрепляют и углубляют знания, полученные во время аудиторных занятий, готовятся к экзамену, а также самостоятельно изучают отдельные темы учебной программы.

7 Фонд оценочных средств

7.1 Методы контроля и оценивания результатов обучения

В процессе обучения используются следующие оценочные формы самостоятельной работы студентов, оценочные средства текущего контроля успеваемости и промежуточных аттестаций:

- экзамен.

7.2 Шкала и критерии оценивания результатов обучения

Показателем оценивания компетенций на различных этапах их формирования является достижение обучающимися планируемых результатов обучения по дисциплине (модулю):

ОПК-7 Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности				
Показатель	**Критерии оценивания**			
	2	**3**	**4**	**5**
знать: • реверс инжиниринг программно-аппаратных решений; • разработка программного обеспечения; • исследование и анализ программного кода; • написание отчетной и технической документации о выполненных работах.	Обучающийся демонстрирует полное отсутствие или недостаточное соответствие следующих знаний: • реверс инжиниринг программно-аппаратных решений; • разработка программного обеспечения; • исследование и анализ программного кода; • написани	Обучающийся демонстрирует неполное соответствие следующих знаний: • реверс инжиниринг программно-аппаратных решений; • разработка программного обеспечения; • исследование и анализ программного кода; • написание отчетной и технической документации о	Обучающийся демонстрирует частичное соответствие следующих знаний: • реверс инжиниринг программно-аппаратных решений; • разработка программного обеспечения; • исследование и анализ программного кода; • написание отчетной и технической	Обучающийся демонстрирует полное соответствие следующих знаний: • реверс инжиниринг программно-аппаратных решений; • разработка программного обеспечения; • исследование и анализ программного кода; • написание отчетной и технической

	е отчетной и технической документации о выполненных работах..	выполненных работах. Допускаются значительные ошибки, проявляется недостаточность знаний, по ряду показателей, обучающийся испытывает значительные затруднения при оперировании знаниями при их переносе на новые ситуации.	документации о выполненных работах. , но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях.	документации о выполненных работах. , свободно оперирует приобретенными знаниями.
уметь: применять на практике знания, полученные в области реверс инжиниринга программного кода.	Обучающийся не умеет или в недостаточной степени умеет применять на практике знания, полученные в области реверс инжиниринга программного кода.	Обучающийся демонстрирует неполное соответствие следующих умений: применять на практике знания, полученные в области реверс инжиниринга программного кода. Допускаются значительные ошибки, проявляется недостаточность умений.	Обучающийся демонстрирует частичное соответствие следующих умений: применять на практике знания, полученные в области реверс инжиниринга программного кода. Умения освоены, но допускаются незначительные ошибки, неточности.	Обучающийся демонстрирует полное соответствие следующих умений: применять на практике знания, полученные в области реверс инжиниринга программного кода. Свободно оперирует приобретенными умениями, применяет их в ситуациях повышенной сложности.

владеть: навыками исследования и анализа программного кода и написания отчётной, технической документации.	Обучающийся не владеет или в недостаточной степени владеет навыками исследования и анализа программного кода и написания отчётной, технической документации.	Обучающийся владеет навыками исследования и анализа программного кода и написания отчётной, технической документации, но допускаются значительные ошибки, проявляется недостаточность владения	Обучающийся частично владеет навыками исследования и анализа программного кода и написания отчётной, технической документации, навыки освоены, но допускаются незначительные ошибки, неточности, затруднения.	Обучающийся в полном объеме владеет навыками исследования и анализа программного кода и написания отчётной, технической документации, свободно применяет полученные навыки в ситуациях повышенной сложности.

Шкалы оценивания результатов промежуточной аттестации и их описание:

Форма промежуточной аттестации: экзамен.

Промежуточная аттестация обучающихся в форме экзамена 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 (модулю) проводится преподавателем, ведущим занятия по дисциплине (модулю) методом экспертной оценки. По итогам промежуточной аттестации по дисциплине (модулю) выставляется оценка «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно».

Шкала оценивания	**Описание**
Отлично	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях,

	переносе знаний и умений на новые, нестандартные ситуации.
Хорошо	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует неполное, правильное соответствие знаний, умений, навыков приведенным в таблицах показателей, либо если при этом были допущены 2-3 несущественные ошибки.
Удовлетворительно	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, в котором освещена основная, наиболее важная часть материала, но при этом допущена одна значительная ошибка или неточность.
Неудовлетворительно	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

7.3 Оценочные средства

7.3.1 Список вопросов для экзамена

1. Программирование приложений ОС с помощью интерфейса API операционных систем.
2. Вычисление арифметических выражений на языке Ассемблера
3. Команды перехода. Программирование разветвленных алгоритмов на Ассемблере.
4. Программирование циклических алгоритмов на Ассемблере
5. Использование логических команд Ассемблера
6. Обработка строк на ассемблере
7. Программирование ввода-вывода на Ассемблере. Использование системных прерываний.
8. Использование системных прерываний для работы с дисковой памятью в ассемблерных программах
9. Способы передачи данных в подпрограмму.
10. Палитры компонент Standard, Additional, Win32, Win3.1, Dialogs, Samples, System. Обзор компонентов.

11. Применение диалоговых окон
12. Обработка исключительных ситуаций. Виды ошибок.
13. Сравнительный анализ достоинств и недостатков компонентов работы со строковой информацией.
14. Создание собственного файла настройки шаблонов маски ввода
15. Сравнительный анализ достоинств и недостатков компонентов для создания панелей инструментов.
16. Ознакомительный анализ компонента ImageList – списка изображений. Основные настроечные свойства. Достоинства применения
17. Сравнительный анализ достоинств и недостатков компонентов для управления числовыми величинами.
18. Методы рисования простейшей графики. Методы вывода теста. Методы вывода растровых изображений.
19. Звук. Типы звуковых и мультимедийных файлов. Процедуры воспроизведения звуков. Воспроизведение немых видео клипов.
20. Нестандартные манипуляции с текстом. Структура LOGFONT. Вывод текста под наклоном и придание других эффектов.
21. Создание мультипликации с текстовым сообщение.
22. Способы передачи данных в подпрограмму.
23. Обработка исключительных ситуаций. Виды ошибок.
24. Сравнительный анализ достоинств и недостатков компонентов для создания панелей инструментов.
25. Методы рисования простейшей графики. Методы вывода теста. Методы вывода растровых изображений.