

**Вопросы к государственному экзамену**  
**Магистерская программа "Компиляторные технологии"**

1. Основные понятия дедуктивной верификации. Методы доказательства корректности программ.
2. Основные понятия дедуктивной верификации. Методы доказательства завершимости программ.

**Список рекомендованной литературы**

1. Буздалов, Корныхин, Панфёров, Петренко, Хорошилов. *Практикум по дедуктивной верификации программ: учебно-методическое пособие.* – М.: МАКС-Пресс, 2014.
  2. Б.Мейер. *Объектно-ориентированное конструирование программных систем* – М.: Русская Редакция, 2005.
- 
3. Основные сведения об объектном языке ограничений (OCL): состав OCL-выражения, навигация по ассоциациям, виды коллекций, операции с коллекциями, учёт наследования в выражениях и наследование ограничений. Примеры использования OCL.
  4. Способы объектно-реляционного отображения для классов и атрибутов, бинарных и N-арных ассоциаций, классов ассоциаций, иерархий наследования. Примеры применения этих способов. Моделирование схемы реляционной базы данных с помощью диаграммы классов.
  5. Образцы (паттерны) проектирования, их классификация и способ описания. Примеры образцов: структурного, поведенческого и порождающего.

**Список рекомендованной литературы**

1. Арлоу Д., Нейштадт А. *UML 2 и унифицированный процесс. Практический объектно-ориентированный анализ и проектирование.* - СПб.: Символ-Плюс. - 2008. Глава 25.
  2. Рамбо Дж., Блаха М. *UML 2.0. Объектно-ориентированное моделирование и разработка.* - СПб.: Питер. - 2007. Главы 3 и 19.
  3. Гамма Э. и др. *Приемы объектно-ориентированного проектирования. Паттерны проектирования.*: Пер. с англ. - СПб.: Питер, 2016.
- 
6. Основные понятия безопасности информации: конфиденциальность, целостность, доступность. Виды защиты информации. Модель Белла-Лападулы. Понятие ошибки, уязвимости в программном обеспечении, примеры.
  7. Ошибка типа «переполнение буфера». Выполнение произвольного кода на исполнимом стеке. Противодействие выполнению кода на стеке: «канарейка», DEP. Выполнение произвольного кода на неисполнимом стеке. Return-to-libc, return-oriented programming (ROP).
  8. Статический анализ исходного кода с целью поиска ошибок. Типы обнаруживаемых ошибок. Путь распространения ошибки: source, propagation, sink. Поточковая и контекстная чувствительность. Качество результата анализа: false/true positive/negative. Интерпретация результатов анализа.
  9. Применение отладки для оценки возможности эксплуатации уязвимостей. Технологии отладки. Отладка пользовательского кода. Полносистемная отладка в виртуальной машине. Статическое и динамическое инструментирование. Фаззинг. Разновидности фаззинга: черный ящик, белый ящик, серый ящик.
  10. Символьное выполнение: основные понятия. Схема работы системы символьного выполнения. Предикат пути, предикат безопасности. Проблема экспоненциального взрыва, стратегии выбора следующего состояния.

## Список рекомендованной литературы

1. Brian Chess, Jacob West. *Secure Programming with Static Analysis* / Addison-Wesley Professional, 2007.
  2. *Aleph One. Smashing the Stack for Fun and Profit*
  3. Edward J. Schwartz, Thanassis Avgerinos, and David Brumley. *Q: Exploit Hardening Made Easy*.
  4. Al Bessey, Ken Block, Ben Chelf, Andy Chou, Bryan Fulton, Seth Hallem, Charles Henri-Gros, Asya Kamsky, Scott McPeak, Dawson Engler. *A Few Billion Lines of Code Later: Using Static Analysis to Find Bugs in the Real World*.
  5. William R. Bush, Jonathan D. Pincus, David J. Sielaff. *A Static Analyzer for Finding Dynamic Programming Errors*.
  6. Eli Bendersky. Серия статей "How debuggers work".
  7. Chow J., Garfinkel T., Chen P. M. *Decoupling dynamic program analysis from execution in virtual environments* // USENIX 2008 Annual Technical Conference on Annual Technical Conference. – 2008. – С. 1-14
  8. Nethercote N., Seward J. *Valgrind: a framework for heavyweight dynamic binary instrumentation* // ACM Sigplan notices. – ACM, 2007. – Т. 42. – №. 6. – С. 89-100.
  9. Амини П., Самтон М., Грин А. *Fuzzing: исследование уязвимостей методом грубой силы*. — Символ-Плюс, 2009.
  10. Edward J. Schwartz, Thanassis Avgerinos, David Brumley. *All You Ever Wanted to Know about Dynamic Taint Analysis and Forward Symbolic Execution (but might have been afraid to ask)*, 2010
  11. C. Cadar, D. Dunbar, D. Engler. *KLEE: Unassisted and Automatic Generation of High-Coverage Tests for Complex Systems Programs*, 2008.
- 

11. Метод нумерации значений в пределах базового блока и в пределах процедуры. Реализация метода путем построения ориентированных ациклических графов и использования хеш-таблиц.
12. Исключение частично-избыточных выражений методом анализа потока данных.
13. Граф зависимостей программы: определение, построение, применение.
14. Проблемы статического анализа объектно-ориентированных языков (C++, Java). Поток управления в присутствии исключений. Вызовы по указателю и их анализ. Понятие о девиртуализации.
15. Инструментирование при динамическом анализе: инструментирование исходного кода программ при компиляции, динамическая двоичная трансляция.

## Список рекомендованной литературы

1. Альфред В. Ахо, Моника С. Лам, Рави Сети, Джефффри Д. Ульман. *Компиляторы: принципы, технологии и инструментарий*. Второе издание. Москва, Вильямс, 2008.
- 

16. Информационная безопасность. Шифрование данных. Криптографическая стойкость. Симметричная криптография. Блочный шифр (DES) и его режимы. Ассиметричные схемы (RSA и Диффи-Хеллмана). Код аутентификации (MAC). Цифровая подпись (DSA).
17. Понятие анонимности пользователя в сети. Идентификаторы пользователя в сети на разных уровнях (устройства, ОС, ПО). Подходы к деанонимизации и способы защиты. Концепция анонимных сетей (Mix и Tor). Луковая маршрутизация. Виды атак на анонимные сети.

## Список рекомендованной литературы

1. Эндрю Таненбаум, Дэвид Уэзеролл. *Компьютерные сети*. Пятое издание. — СПб.: Питер, 2012.
2. Jon Mark Allen. *OS and Application Fingerprinting Techniques*. – SANS Institute, 2007.
3. Grahm, K. J., Forss, T., & Pulkkis, G. *Anonymous Communication on the Internet*. *Proceedings of Informing Science & IT Education Conference (InSITE) 2014* (pp. 103-120).