МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ»
КАФЕДРА КІБЕРБЕЗПЕКИ

М. Д. Василенко, В. О. Рачук, В. М. Слатвінська

АНАЛІЗ ШКІДЛИВОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

НАВЧАЛЬНО-МЕТОДИЧНІ РЕКОМЕНДАЦІЇ

(в допомогу до самостійної роботи для здобувачів вищої освіти кваліфікації бакалавр факультету кібербезпеки та інформаційних технологій)

Одеса - 2020

УДК

Рекомендовано Навчально-методичною радою Національного університету «Одеська юридична академія» (протокол № 2 від 4 грудня 2020 р.)

Рецензенти:

Тупкало В.М. - доктор технічних наук, професор, завідувач кафедри менеджменту та інформаційних технологій Інституту інтелектуальної власності та права Національного університету «Одеська юридична академія», м. Київ

Логінова Н.І. - кандидат педагогічних наук, доцент, завідувачка кафедри інформаційних технологій Національного університету «Одеська юридична академія»

М.Д. Василенко, В. О. Рачук, В.М. Слатвінська

Аналіз шкідливого програмного забезпечення: навчально-методичні рекомендації (в допомогу до самостійної роботи для здобувачів вищої освіти кваліфікації бакалавр факультету кібербезпеки та інформаційних технологій). Одеса : Видавничий дім «Гельветика» 2020. 31 с.

Навчально-методичні рекомендації з курсу «Аналіз шкідливого програмного забезпечення» розроблено відповідно до навчального плану, вони складаються з навчальної програми курсу, методичних рекомендацій із проведення практичних занять, питань до самоконтролю, завдань для самостійної роботи, списку рекомендованої літератури. Вивчення дисципліни «Аналіз шкідливого програмного забезпечення» допоможе студентам розвинути навички навчання практичним методам ізоляції і аналізу шкідливого програмного забезпечення на низькому рівні, для чого необхідні знання та навички віртуалізації і аналізу роботи системи в машинних кодах і програмування в різних асемблерах. Матеріали призначено для студентів факультету кібербезпеки та інформаційних технологій Національного університету «Одеська юридична академія», які навчаються за спеціальністю «кібербезпека».

ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Навчальна дисципліна «Аналіз шкідливого програмного забезпечення» призначена для студентів, що навчаються за освітньо-кваліфікаційним рівнем «бакалавр».

Актуальність курсу зумовлена тим, що розповсюдження шкідливого програмного забезпечення (ШПЗ) є однією з найбільш небезпечних загроз, що впливає на безпеку даних у сучасному кіберсередовищі. Аналіз шкідливих програм і аналіз дампов пам'яті - це потужні методи аналізу і розслідування, які використовуються в реверс-інжинірингу, цифровий криміналістиці і при реагуванні на інциденти. Через те, що зловмисники стають все більш витонченими і здійснюють атаки з використанням складного шкідливого ПО на критичні інфраструктури, центри обробки даних і інші організації, виявлення і розслідування таких вторгнень, реагування на них мають вирішальне значення для професіоналів в області інформаційної безпеки.

Предметом вивчення навчальної дисципліни є низькорівневий аналіз програмного забезпечення, яке перешкоджає роботі комп'ютера, збирає конфіденційну інформацію або отримує доступ до приватних комп'ютерних систем.

В курсі використана віртуальна машина DOSBox, що дозволяє експлуатувати його на широкому спектрі апаратних і програмних систем, робить його платформонезалежним і ліцензійно-чистим з точки зору використовуваного програмного забезпечення.

Міждисциплінарні зв'язки: вивчення дисципліни «Аналіз шкідливого програмного забезпечення» базується на дисципліні «Операційні системи» та повинна передувати дисциплінам: "Кібербезпека", "Управління інформаційною безпекою", "Спеціалізовані комп'ютерні системи захисту інформації"

Оволодіння дисципліни «Аналіз шкідливого програмного забезпечення» передбачає читання лекцій, проведення практичних занять, консультацій, самостійної роботи та іспиту.

ЗАПЛАНОВАНІ РЕЗУЛЬТАТИ НАВЧАННЯ

Мета викладання навчальної дисципліни «Аналіз шкідливого програмного забезпечення» - навчання практичним методам ізоляції і аналізу ШПЗ на низькому рівні, для чого необхідні знання та навички віртуалізації і аналізу роботи системи в машинних кодах і програмування в різних асемблерах.

Основними завданнями вивчення дисципліни «Аналіз шкідливого програмного забезпечення» є вивчення класифікації та побудови типових ШПЗ вивчення принципів та систем ізоляції підозрюванного програмного забезпечення (ППЗ) за допомогою технологій віртуалізації різного рівня; вивчення складання та побудови програми з використанням мови асебмлера та машинного коду; вивчення аналізу ППЗ з використанням мови асебмлера, системної архитектури (ISA) та машинного коду.

Згідно з вимогами освітньо-професійної програми студенти повинні:

знати:

класифікацію та побудову основних ШПЗ

принципи, інструменти, засоби написання і аналізу програмного забезпечення на низькому рівні (машинний код / асемблер) **вміти:**

ізолювати підозрюване програмне забезпечення за допомогою технологій віртуалізації різного рівня

складати алгоритми і програми з використанням мови асебмлера та машинного коду

аналізувати підозрюване програмне забезпечення з використанням мови асебмлера та машинного коду

використовувати на практиці навички розуміння машинного коду **бути ознайомленими:**

з використанням програмних та програмно-апаратних комплексів засобів захисту інформації в інформаційно-телекомунікаційних (автоматизованих) системах

з відновленням функціонування інформаційних, інформаційно-телекомунікаційних (автоматизованих) систем після реалізації загроз, здійснення кібератак, збоїв та відмов різних класів та походження

СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

№ теми	Види занять	Лекційні заняття (год.)		Практичні заняття (год.)		Самостійна робота (год.)	
	Тема	очна	заочна	очна	заочна	очна	заочна
1	Віртуалізація. FreeDOS. DOSBox	5	2	2	0	10	10
2	Користувальницькі інтерфейси ОС	5	2	2	0	14	14
3	Низькорівневе програмування та аналіз ПЗ	5	0	0	0	14	14
4	Ансемблери	5	0	1	0	14	14
	Всього	20	6	5	2	52	52

ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Тема 1. Віртуалізація. FreeDOS. DOSBox

Віртуальна машина. Сфери використання віртуальних машин. Недоліки використання віртуальних машин. Переваги використання віртуальних машин. Види віртуальних машин. Операційна система як віртуальна машина. Методика застосування технологій віртуалізації Unix подібних операційних систем у підготовці бакалаврів інформатики. Поняття віртуалізації. Систематизація технологій віртуалізації. Команди DOS. Захист інформації і обмеження можливостей програм. DOSBox. Віртуалізація платформи. Віртуалізація прикладного програмного забезпечення. FreeDOS. DOSBox.

Тема 2. Користувальницькі інтерфейси ОС

Етапи розробки користувальницького інтерфейсу. Методи розробки користувальницького інтерфейсу. Графічний користувальницький інтерфейс. Структура і класифікація користувальницьких інтерфейсів. Основні принципи створення користувацького інтерфейсу. Управляючі кошти для користувача інтерфейсу. Якість інтерфейсу. Покоління інтерфейсу користувача. Класи та підкласи користувача інтерфейсу в інформаційних системах. Інтерфейс командного рядка. Консоль. Термінал. Формат команди. Параметри команд. Внутрішні команди. ПЗ командного рядку. Команди роботи з дисками. Команди роботи з файлами. Команди роботи з каталогами. Команди управління системою. Двохпанельні інтерфейси. Пакетні командні файли.

Тема 3. Низькорівневе програмування та аналіз ПЗ

Низькорівневе програмування. Низькорівневе програмування комп'ютерної графіки. Низькорівневе програмування контроллера клавіатури. Архітектура комп'ютера та низькорівневе програмування. Архітектура та низькорівневе програмування засобів Інтернету речей. Низькорівневе програмування мікроконтролерів. Низькорівневе програмування на JS. Пам'ять, регістри та адресація. Способи адресації пам'яті. Типи адресації. Методи адресації. Регістрова пам'ять. Регістрова адресація. Способи адресації операндів на основі операції зміщення. Індексна адресація. Непряма регістрова адресація. Види комп'ютерної пам'яті. Фізична пам'ять комп'ютера. Оперативна пам'ять комп'ютера. Зовнішня і внутрішня пам'ять комп'ютера.

Рівні пам'яті комп'ютера. Статична і динамічна пам'ять комп'ютера. Арифметика мікропроцесора I8086.

Тема 4. Асемблери

Життєвий цикл програм. Мова асемблера. Речення мови асемблера. Операнди команд. Директиви. Структури (бази) даних асемблера. Алгоритми роботи асемблерів. Двопрохідний асемблер - перший прохід. Деякі структури даних першого проходу. Структура таблиць асемблера. Двопрохідний асемблер - другий прохід. Деякі додаткові директиви. Одно- та багатопрохідний асемблер. Команди процесора на мові Асемблер. Операції зі строками в мові Асемблер. Робота з масивами в мові Асемблер. Макрозасоби мови Асемблер. Процедури в мові Асемблер. Символьні відладчики. C/C++ компілятор. C-SPY відладчик. Компілятори, компонувальники, відладчики. Поняття про початкові коди. Практична робота з турбоассемблером. Регістр прапорів. Управління виконанням програми. Підпрограми.

ТЕМИ ПРАКТИЧНИХ ЗАНЯТЬ

Тема 1. Віртуалізація. FreeDOS. DOSBox

Питання до підготовки:

1. Віртуальна машина. Захист інформації і обмеження можливостей програм. DOSBox.
2. Віртуалізація платформи.
3. Віртуалізація прикладного програмного забезпечення.
4. FreeDOS. DOSBox.

Література:
Основна: 1-15
Допоміжна: 1-23

Тема 2. Користувальницькі інтерфейси ОС

Питання до підготовки:

1. Інтерфейс командного рядка. Консоль. Термінал.
2. Формат команди. Параметри команд. Внутрішні команди. ПЗ командного рядку. Команди роботи з дисками. Команди роботи з файлами. Команди роботи з каталогами. Команди управління системою.
3. Двохпанельні інтерфейси.
4. Пакетні командні файли.

Література:
Основна: 1-15
Допоміжна: 1-23

Тема 3. Низькорівневе програмування та аналіз ПЗ

Питання до підготовки:

1. Низькорівневе програмування.
2. Пам'ять, регістри та адресація
3. Арифметика мікропроцесора I8086

Література:
Основна: 1-15
Допоміжна: 1-23

Тема 4. Асемблери

Питання до підготовки:

1. Життєвий цикл програм.
2. Компілятори, компонувальники, відладчики
3. Поняття про початкові коди. Практична робота з турбоассемблером
4. Регістр прапорів. Управління виконанням програми. Підпрограми

Література:

Основна: 1-15

Допоміжна: 1-23

ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

Тема 1. Віртуалізація. FreeDOS. DOSBox

1. Підготувати доповіді на теми:

- Сфери використання віртуальних машин
- Недоліки використання віртуальних машин
- Переваги використання віртуальних машин
- Види віртуальних машин
- Операційна система як віртуальна машина
- Методика застосування технологій віртуалізації Unix подібних операційних систем у підготовці бакалаврів інформатики
- Поняття віртуалізації
- Історія віртуалізації
- Систематизація технологій віртуалізації
- Команди DOS
- Технології захисту інформації
- Захист інформації в локальних мережах
- Комплексна система захисту інформації
- Способи захисту інформації на комп'ютері Mac
- Організаційні методи захисту інформації
- Програмні засоби захисту інформації
- Програми для захисту інформації

2. Ознайомитись з VirtualBox, VMware Workstation, Microsoft Virtual PC, Parallels Workstation, VMware Workstation, VMware Player, VMware Fusion, VMware Server, VMware Ace, VMware vSphere, VMware vSphere Hypervisor.
3. Завантажити операційну систему FreeDOS за посиланням: http: //www.freedos.org/download/.

Інсталяційний файл повинен бути CD образ (файл з розширенням .ISO), зберегти його в папці користувача на комп'ютері.

ЗапуститьVirtualBox. Щоб створити нову віртуальну машину нажати New (Ctrl+ N).

Задати ім'я, тип і версію операційної системи і натиснути Next:

- Name: FreeDOS

-Type: Other

- Version: DOS5.

Задати кількість оперативної пам'яті (RAM) в мегабайтах виділяються віртуально машині (32 Мб достатньо).

Натисніть Next.

Вибрати create a virtual hard disk now (створити новий віртуальний жорсткий диск) і натиснути create.

Вибрати тип жорсткого диска VDI і натиснути кнопку Next. Вибрати dynamically allocated (динамічно виділений) жорсткий диск і натиснути next.

Задати ім'я створюваному жорсткому диску ггеескш розмір рівний 500 Мб, натиснути Create

4. Дайте відповіді на наступні питання:

1. Що таке віртуальна машина?
2. Що таке «хостова» ОС?
3. Що таке «гостьова» ОС?
4. Які основні схеми віртуалізації відомі?
5. Який принцип дії віртуальних машин з емуляцією API гостьової ОС?
6. Який принцип дії віртуальних машин з квазіемуляцією гостьовий ОС?
7. Який принцип дії віртуальних машин з повною емуляцією гостьовий ОС?
8. Які сфери застосування віртуальних машин відомі?

Тема 2. Користувальницькі інтерфейси ОС

1. Підготувати доповіді на теми:
   - Етапи розробки користувальницького інтерфейсу
   - Методи розробки користувальницького інтерфейсу
   - Графічний користувальницький інтерфейс
   - Структура і класифікація користувальницьких інтерфейсів
   - Основні принципи створення користувацького інтерфейсу
   - Управляючі кошти для користувача інтерфейсу
   - Якість інтерфейсу
   - Покоління інтерфейсу користувача
   - Класи та підкласи користувача інтерфейсу в інформаційних системах
   - Команди MS-DOS для роботи з каталогами
   - Формати архівів
   - Програми-оболонки
   - Способи подачі команди в Windows
   - Створення командних файлів
   - Пакетні файли в Windows
   - Термінал і консоль для Firefox
   - Термінал Windows
   - Консоль Windows

- Консоль terminal з HTML
- GPRS термінал
- Термінал в Mac (macOS)
- Термінал Linux
- Бібліотека функцій WinAPI
- Бібліотека класів VCL
- Бібліотека класів MFC
- Бібліотека. Net Framework (C #)
- Типові інтерфейси графічних редакторів
- Проектування інтерфейсу обміну даними між клієнтом і сервером
- Задачі і цілі тестування користувальницького інтерфейсу
- Функціональне тестування користувацьких інтерфейсів
- Тестування зручності використання користувальницьких інтерфейсів

2. Ознайомитись з Unreal Commander, Multi Commander, Altap Salamander
3. Ознайомитись з Cmder, Hyper, Terminus, Fluent Terminal, Alacritty
4. Поясніть структуру файлової системи в операційній системі MS DOS (Windows).
5. Поясніть основні правила побудови імен для файлів і каталогів.
6. Поясніть правила перенаправлення вводу-виводу для команд операційної системи.
7. Наведіть команди управління логічними дисками.
8. Наведіть команди управління каталогами.
9. Наведіть команди управління файлами.
10. Поясніть призначення командних файлів операційної системи.
11. Як здійснюється блокування виводу команд і повідомлень при виконанні командного файлу?
12. Які умови перевіряються командою if?
13. Поясніть організацію циклів в командних файлах?
14. Самостійна робота

Завдання 1

Налаштування вікна командної оболонки Відкрийте вікно Командний рядок. Клацніть в лівому верхньому куті вікна та виберіть команду Властивості. Виберіть вкладку Загальні. В області Запам'ятовування команд виберіть або введіть значення 999 в полі Розмір буфера, а потім виберіть або введіть значення 5 у полі Кількість буферів. В області Правка встановіть галочки Виділення мишею і Швидка вставка. Виберіть вкладку Розташування. В області Розмір буфера екрана введіть або виберіть значення 2500 у полі Висота. Виконайте одну з наступних дій на вибір. В області Розмір буфера

екрана збільшіть значення параметра Ширина. В області Розмір вікна збільшіть значення параметра Висота. В області Розмір вікна збільшіть значення параметра Ширина. Зніміть прапорець Автоматичний вибір, а потім в області Положення вікна змініть значення полів Лівий край і Верхній край.

15. Дайте відповідь на наступні питання:

1. Що таке командна оболонка.
2. Чи існує можливість вкладати командні оболонки в Cmd.exe.
3. Що визначають системні змінні середовища Cmd.exe.
4. Що визначають локальні змінні середовища Cmd.exe.
5. Що виконує команда Dir.
6. Що виконує команда Chdir або Cd.
7. Що виконує команда Mkdir або Md.
8. Що виконує команда Rmdir або Rd.
9. Що виконує команда Copy.
10. Що виконує команда Move.
11. Що виконує команда Del або Erase.
12. Що виконує команда Exit.
13. Що виконує команда Netstat.
14. Що виконує команда Ping.
15. Що виконує команда Pathping.
16. Що виконує команда Tracert.
17. Що виконує команда Ipconfig.
18. Що виконує команда Route.

Тема 3. Низькорівневе програмування та аналіз ПЗ

1. Підготувати доповіді на теми:

- Низькорівневе програмування комп'ютерної графіки
- Низькорівневе програмування контроллера клавіатури
- Архітектура комп'ютера та низькорівневе програмування
- Архітектура та низькорівневе програмування засобів Інтернету речей
- Низькорівневе програмування мікроконтролерів
- Низькорівневе програмування на JS
- Способи адресації пам'яті
- Вимоги до ПЗ
- Типи адресації
- Методи адресації
- Регістрова пам'ять

- Регістрова адресація
- Способи адресації операндів на основі операції зміщення
- Індексна адресація
- Непряма регістрова адресація
- Види комп'ютерної пам'яті
- Фізична пам'ять комп'ютера
- Оперативна пам'ять комп'ютера
- Зовнішня і внутрішня пам'ять комп'ютера
- Рівні пам'яті комп'ютера
- Статична і динамічна пам'ять комп'ютера
- intel 8086 характеристики
- Подання чисел у мікропроцесорі
- Двійкова арифметика і intel 8086
- Двійкова система числення у intel 8086
- Спеціальні функції регістрів мікропроцесора Intel 8086
- Мікроконтролери. Гарвардська архітектура.
- Ядро, постійна пам'ять, оперативна пам'ять, таймери, периферія.
- Машинні коди PIC16 RISC.
- Асемблер MPASM.
- Мікроконтролери PIC компанії Microchip.
- Технічний опис, архітектура та структура мікроконтролера PIC16F84A.
- Електронна схема включення мікроконтролерів.
- Засоби розробки програм для мікроконтролерів та симуляції роботи мікроконтролерів PIC (MpLabX, KtechLab, Proteus).
- Основи асемблера мікроконтролерів PIC.
- Структура програми мовою асемблера MPASM.
- Регістри спеціального призначення.
- Основи програмування мовою С.
- Програмування роботи цифрових портів вводу-виводу мікроконтролерів.
- Програмування таймерів мікроконтролерів, програмування часових затримок.
- Модуль послідовного зв'язку USART та EUSART.
- Програмна та апаратна реалізація асинхронного послідовного передавання та приймання інформації.
- Підключення вбудованого АЦП.
- Читання та запис даних у статичну пам'ять EEPROM.
- Архітектура та структура мікроконтролерів STM32.
- Ядро Cortex.
- Периферія, адресні шини та шини даних.

- Асемблер 32 розрядних мікроконтролерів.
- Інтегроване середовище розробки CooCox.
- Програмування мікроконтролерів мовою С.

2. Намалювати структурну схему мікропроцесора Intel 8086

3. Ознайомитись з низькорівневими мовами: Асемблер, Pcode, Байт-код Java, Perl, CIL

Тема 4. Асемблери

1. Підготувати доповіді на теми:
- Мова асемблера
- Речення мови асемблера
- Операнди команд
- Директиви
- Структури (бази) даних асемблера
- Алгоритми роботи асемблерів
- Двопрохідний асемблер - перший прохід
- Деякі структури даних першого проходу
- Структура таблиць асемблера
- Двопрохідний асемблер - другий прохід
- Деякі додаткові директиви
- Одно- та багатопрохідний асемблер
- Команди процесора на мові Асемблер.
- Операції зі строками в мові Асемблер
- Робота з масивами в мові Асемблер
- Макрозасоби мови Асемблер
- Процедури в мові Асемблер
- Символьні відладчики
- C/C++ компілятор
- C-SPY відладчик
2. Самостіна робота

Завдання № 1

На базі програми test01.asm створити програму z11.asm, що виконує очищення екрана (шляхом переустановки відеорежиму) і виводить на екран довільне повідомлення. Алгоритм роботи програми: 1) настроїти регістр DS на сегмент даних; 2) очистити екран; 3) вивести повідомлення; 4) завершити роботу.

Примітка. Для переустановки відеорежиму використовується функція 0 переривання 10h. Для її виклику потрібно за допомогою інструкції mov

помістити в регістри такі дані: у регістр AH: 0 (номер функції), у регістр AL: 3 (номер відеорежиму), потім викликати переривання інструкцією int.

Завдання № 2

На базі програми zll.asm створити програму z12.asm, що після очищення екрана виводить повідомлення в заданих координатах (номер рядка і номер стовпця).
Примітка. Для переміщення курсору в задану позицію екрана використовується функція 2 переривання 10h. Для її виклику потрібно за допомогою інструкції mov помістити в регістри такі дані: у регістр AH: 2 (номер функції), у регістр DH - номер рядка (від 0 до 24), у регістр DL -номер стовпця (від 0 до 79), у регістр BH 0 (номер відеосторінки), потім викликати переривання інструкцією int (у даному випадку int 10h).

Завдання № 3

На базі програми z12.asm створити програму z13.asm, у якій очищення екрана виконується не переустановкою відеорежиму, а прокручуванням вікна (при цьому вікно буде заповнено заданим атрибутом). *Примітка.* Для прокручування вікна використовується функція 6 переривання 10h. Для її виклику потрібно за допомогою інструкції mov помістити в регістри такі дані: у регістр AH: 6 (номер функції), у регістр CH 0 (номер рядка верхнього лівого кута вікна), у регістр CL 0 (номер стовпця верхнього лівого кута вікна), у регістр DH 24 (номер рядка правого нижнього кута вікна), у регістр DL 79 (номер стовпця правого нижнього кута), у регістр BH - атрибут заповнення (старший напівбайт атрибута -колір тла, молодший напівбайт - колір символів, наприклад, атрибут 07h -сірі символи на чорному тлі, 1Eh - жовті символи на синім тлі), потім викликати переривання інструкцією int (у даному випадку int 10h).

3. Відповісти на наступні питання:

1) Що являє собою асемблерна програма?
2) З яких сегментів складається програма?
3) Які існують основні види регістрів процесорів сімейства Intel x86?
4) Які сегментні регістри вказують на відповідні сегменти програми?
5) На які групи поділяються пропозиції асемблера?
6) Основні етапи створення асемблерної програми.
7) Які спеціальні програми використовуються для асемблювання і компонування?
8) Як виконати асемблювання і компонування програми зі збереженням налагоджувальної інформації?

9) Як завантажити програму в налагоджувач?

10) Як запустити програму на виконання в середовищі налагоджувача?

11) Яким образом виконується виконання програми в покроковому режимі?

12) Як установити або прибрати контрольну точку?

4. Проаналізувати наступні онлайн-компілятори: CodePad.io, C++shell, IdeOne, CodingGround, JDoodle, TryCode.

ТЕМИ РЕФЕРАТІВ

1. Поведінка шкідливих програм.
2. Програми для завантаження і запуску ПЗ.
3. Викрадення облікових даних.
4. Руткіти, що працюють в режимі користувача.
5. Прихований запуск шкідливого ПЗ.
6. Впровадження перехоплювачів.
7. Впровадження асинхронних процедур.
8. Кодування даних.
9. Нестандартне кодування.
10. Поширені криптографічні алгоритми.
11. Мережеві сигнатури, націлені на шкідливе ПЗ.
12. Мережеві контрзаходи.
13. Контрзаходи, засновані на трафіку мережі.
14. Поєднання динамічних і статичних методик аналізу.
15. Аналіз методів виявлення шкідливого програмного забезпечення в комп'ютерних системах.
16. Класифікація шкідливого програмного забезпечення.
17. Призначення асемблера.
18. Призначення мови асемблера.
19. Речення мови асемблера.
20. Команди асемблера.
21. Операнди команд асемблера.
22. Директиви асемблера.
23. Операнди директив асемблера.
24. Структури (бази) даних асемблера.
25. Структури даних першого проходу двопрохідного асемблера.
26. Алгоритм роботи першого проходу двопрохідного асемблера.
27. Структури даних другого проходу двопрохідного асемблера.
28. Алгоритм роботи другого проходу двопрохідного асемблера.
29. Однопрохідний асемблер.
30. Багатопрохідний асемблер.
31. Компілятори та інтерпретатори. Особливості. Відмінності.
32. Загальні принципи компонування.
33. Принципи роботи компонувальника.
34. Вбудований відладчик.
35. Програматор відладчик ST-LINK.
36. Відладчик 1С: Підприємство.

37. Класифікація загрозливого програмного абезпечення.
38. Способи виявлення загроз.
39. Підходи до виявлення загроз на інформаційні ресурси.
40. Класифікація загрозливих програм за способами виконання шкідливих файлів.
41. Оцінка актуальності загрози запуску ШПЗ.
42. Моделювання підходу до захисту від ШПЗ на основі контролю доступу до ресурсів.
43. Аналіз методів виявлення шкідливого програмного забезпечення в комп'ютерних системах.
44. Основні види шкідливого програмного забезпечення на комп'ютері.
45. Методи пошуку вірусів за допомогою антивірусних програм.
46. Еволюція шкідливого програмного забезпечення.
47. Типи шкідливого програмного забезпечення.

ПИТАННЯ ДО ЗАЛІКУ

1. Віртуальна машина.
2. Віртуалізація платформи.
3. Віртуалізація прикладного програмного забезпечення.
4. FreeDOS. DOSBox.
5. Інтерфейс командного рядка.
6. Формат команди. Параметри команд. Внутрішні команди.
7. Двохпанельні інтерфейси.
8. Пакетні командні файли.
9. Низькорівневе програмування.
10. Пам'ять, регістри та адресація
11. Арифметика мікропроцесора Ї8086.
12. Життєвий цикл програм.
13. Компілятори, компонувальники, відладчики.
14. Поняття про початкові коди.
15. Регістр прапорів.
16. Команди управління системою.
17. Команди роботи з дисками.
18. Команди роботи з каталогами.
19. Команди роботи з файлами.
20. Захист інформації і обмеження можливостей програм.
21. Типи адресації.
22. Методи адресації.
23. Регістрова пам'ять.
24. Регістрова адресація.
25. Управління виконанням програми.
26. Підпрограми.
27. ПЗ командного рядку.
28. Консоль. Термінал.
29. Низькорівневе програмування комп'ютерної графіки.
30. Низькорівневе програмування контроллера клавіатури.
31. Архітектура комп'ютера та низькорівневе програмування.
32. Символьні відладчики.
33. C/C++ компілятор.
34. C-SPY відладчик
35. Етапи розробки користувальницького інтерфейсу.
36. Методи розробки користувальницького інтерфейсу.
37. Графічний користувальницький інтерфейс.

38. Структура і класифікація користувальницьких інтерфейсів.
39. Основні принципи створення користувацького інтерфейсу.
40. Якість інтерфейсу.
41. Покоління інтерфейсу користувача.
42. Класи та підкласи користувача інтерфейсу в інформаційних системах.
43. Команди MS-DOS для роботи з каталогами.
44. Формати архівів.
45. Програми-оболонки.
46. Способи подачі команди в Windows.
47. Створення командних файлів.
48. Пакетні файли в Windows.
49. Термінал і консоль для Firefox.
50. Термінал Windows.
51. Консоль Windows.
52. Консоль terminal з HTML.
53. GPRS термінал.
54. Термінал в Mac (macOS).
55. Термінал Linux.
56. Поняття віртуалізації.
57. Історія віртуалізації.
58. Систематизація технологій віртуалізації.
59. Команди DOS.
60. Технології захисту інформації.
61. Захист інформації в локальних мережах.
62. Комплексна система захисту інформації.
63. Способи захисту інформації на комп'ютері Mac.
64. Організаційні методи захисту інформації.
65. Програмні засоби захисту інформації.
66. Програми для захисту інформації.
67. Сфери використання віртуальних машин.
68. Недоліки використання віртуальних машин.
69. Переваги використання віртуальних машин.
70. Види віртуальних машин.
71. Архітектура системи віртуальних машин.
72. Віртуальні машини з емуляцією API гостьової ОС.
73. Віртуальні машини з повною емуляцією гостьовий ОС.
74. Віртуальні машини з квазіемуляцією гостьовий ОС.

ФОРМИ ПІДСУМКОВОГО КОНТРОЛЮ УСПІШНОСТІ СТУДЕНТІВ

Поточний контроль здійснюється під час проведення практичних занять і має на меті перевірку рівня підготовленості студента до виконання конкретної роботи. При поточному контролі оцінці підлягають: рівень теоретичних знань та вміння працювати з науковою літературою, знання матеріалу, продемонстрованого у виконаних (як правило) індивідуальних завданнях; обґрунтованість висновків, а також самостійність та повнота вирішення практичних завдань та аналізу матеріалів; активність та систематичність роботи на заняттях; результати виконання домашніх завдань, тестів, експрес-опитувань тощо.

Форми проведення поточного контролю: усне опитування студентів, вирішення практичних завдань, тестові завдання (для зацікавленості у навчанні, розвитку здібностей студента може бути передбачено виконання інших, індивідуальних для кожного студента завдань).

Проміжний контроль проводиться після вивчення відповідних тем або блоку тем з метою з'ясування ступеню засвоюваності студентами відповідного об'єму опрацьованого та вивченого матеріалу та подальшої оцінки рівня отриманих знань. Форми проведення проміжного контролю: контрольна робота, колоквіум експрес-контроль на лекціях, тестове опитування, співбесіда (усне спілкування).

Підсумковий контроль у формі іспиту.

КРИТЕРІЇ ОЦІНЮВАННЯ

При використанні форми контролю у вигляді заліку враховується поточна, зокрема самостійна робота, наукова діяльність студента. Крім того, студент має надати відповідь на залікове запитання. Оцінка рівня знань виконується за принципом "відповідь вірна" або "відповідь невірна". При вірній відповіді виставляється оцінка "зараховано", при невірній, неповній відповіді виставляється оцінка "не зараховано".

СХЕМА НАРАХУВАННЯ БАЛІВ (СКЛАДОВІ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ)

Пункт оцінки	% підсумкової оцінки або максимальна оцінка в балах	Групове чи індивідуальне оцінювання
Поточний контроль, разом, у т.ч.:	**50**	
доповіді та повідомлення на семінарах	25	Групове та індивідуальне
виконання письмових індивідуальних завдань, рефератів, контрольних робіт	15	Індивідуальне
опитування на семінарських заняттях	10	Групове та індивідуальне
Підсумковий контроль, разом, у т.ч.:	**50**	
письмова компонента	25	Індивідуальне
усна компонента	25	Індивідуальне

ШКАЛА ЗА ECTS

<table>
<tr><th rowspan="2">Сума балів</th><th rowspan="2">Оцінка за 7-бальною шкалою</th><th colspan="2">Оцінка за 4-бальною шкалою</th></tr>
<tr><th>екзамен</th><th>залік</th></tr>
<tr><td>A</td><td>90-100</td><td>Відмінно</td><td rowspan="5">зараховано</td></tr>
<tr><td>B</td><td>82-89</td><td rowspan="2">Добре</td></tr>
<tr><td>C</td><td>75-81</td></tr>
<tr><td>D</td><td>67-74</td><td rowspan="2">Задовільно (достатньо)</td></tr>
<tr><td>E</td><td>60-66</td></tr>
<tr><td>Fx</td><td>35-59</td><td>Незадовільно з можливістю повторного складання</td><td>не зараховано</td></tr>
</table>

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

ОСНОВНА

1. Касперски К. Искусство дизассемблирования / К. Касперски, Е. Рокко. — СПб.: БХВ-Петербург, 2008. — 896 с.: ил. (В подлиннике) ISBN 978-5-9775-0082-1
2. Касперски К. Компьютерные вирусы изнутри и снаружи //СПб.: Питер. - 2006. - Т. 400.
3. Крис К. Техника отладки программ без исходных текстов. - БХВ-Петербург, 2005.
4. Касперски К. Техника и философия хакерских атак. - Солон 1999.
5. Касперский Е. Компьютерные вирусы: что это такое и как с ними бороться. - М.: СК Пресс, 1998.
6. Зубков С. В. Assembler для DOS, Windows и UNIX Серия: Для программистов Издательство: ДМК, 2000 г.
7. Джордейн Р. Справочник программиста персональных компьютеров IBM PC, XT и AT./пер.с англ. - М.: Финансы и статистика, 1992. - 544 с.
8. Столяров А.В. Программирование на языке ассемблера NASM для ОС Unix / Учебное пособие. 2-е изд. М.: МАКС Пресс, 2011, 188 с.
9. Жуков А.В., Авдюхин А.А.Ассемблер. СПб: БХВ-Петербург, 2003. 448 с.
10. Марек Р. Ассемблер на примерах. Базовый курс. / Р.Марек - СПб.:Наука
11. и Техника, 2005. - 240 с.
12.
10.

13. Магда Ю.С. Ассемблер для процессоров Intel Pentium. / Ю.С. Магда - СПб.: Питер, 2006. - 410 с. 12.Калашников О.А. Ассемблер? Это просто! Учимся программировать. / О.А. Калашников - СПб.: БХВ-Петербург, 2006. - 384 с.: ил. 13.Василенко М.Д. Безпека комп'ютерних систем в контексті законодавства та запобігання кіберзагроз // Юридичний вісник. О. : ВД «Гельветика». 2019. № 2. С. 70-76 (в співавторстві). 14.Василенко М.Д. Кібербезпека: кіберзагрози та захищеність технічних (інформаційних) систем. Кібербезпека в сучасному світі: матеріали всеукраїнської науково-практичної конференції (м. Одеса, 29 листопада 2019 р.) / за ред. О. В. Дикого. - Одеса: Видавничий дім «Гельветика», 2019. - С. 86-88 (в співавторстві). 15.Василенко М.Д. Multi-level protection of digital information using alternative data streams Кібербезпека в сучасному світі: матеріали всеукраїнської науково-практичної конференції (м. Одеса, 29 листопада 2019 р.) / за ред. О. В. Дикого. - Одеса: Видавничий дім «Гельветика», 2019.- С. 64 - 65 (в співавторстві).

ДОПОМІЖНА

1. И. И. Шагурин, Е. М. Бердышев Процессоры семейства INTEL P6. Pentium II, Pentium III, Celeron и др. Архитектура, программирование, интерфейс Издательство: Горячая Линия - Телеком, 2000 г.
2. Х. Исида. Программирование для микрокомпьютеров. Москва., Мир, 1988.
3. Майко Г.В. Ассемблер для IBM PC: - М.:"Бизнес-Информ", "Сирин" 1999 г. - 212 с.
4. Донован Дж. Системное программирование. - М.: "Мир", 1975.
5. Бек А. Введение в системное программирование. - М.: "Мир", 1988.
6. Вишняков В.А., Петровский А.А. Системное обеспечение микроЭВМ. -Минск: "Вышэйшая школа", 1990.
7. Абель П. Язык Ассемблера для IBM PC и программирования/пер.с англ. - М.: Высш.шк., 1992. - 447 с.
8. Смирнов Н.Н. Программные средства персональных ЭВМ. - Л.: Машиностроение, 1990. - 271 с.
9. Максимов Ю.Я., Осипов С.В., Симоненков О.С. Практическая работа на компьютерах семейства IBM PC в операционной среде MS-DOS 4.01: Учебное пособие. - М.: ДИАЛОГ-МИФИ, 1991. - 160 с.
10. Нортон П. Программно-аппаратная организация IBM PC./пер.с англ. - М.: Радио и связь, 1991. - 328 с. 11. Нортон П. Персональный компьютер IBM PC и операционная система MS DOS./пер.с англ. - М.: Радио и связь, 1992. - 416 с. 12.Нортон П., Джордейн Р. Работа с жестким диском IBM PC./пер.с англ. - М.: Мир, 1992. - 560 с. 13.Фролов А.В., Фролов Г.В. Аппаратное обеспечение IBM PC: ч.1. - М.: ДИАЛОГ-МИФИ, 1992. - 208 с.
14. Фролов А.В., Фролов Г.В. Аппаратное обеспечение IBM PC: ч.2. - М.: ДИАЛОГ-МИФИ, 1992. - 234 с.
15. Фролов А.В., Фролов Г.В. Операционная система MS DOS: кн.1,2. - М.: ДИАЛОГ-МИФИ, 1992. - 238 с.
16. Фролов А.В., Фролов Г.В. Операционная система MS DOS: кн.3. - М.: ДИАЛОГ-МИФИ, 1992. - 224 с.
17. Справочное руководство по IBM PC: ч.1: Методические материалы - М.: ТПП "Сфера", 1991. - 174 с.
18. Справочное руководство по IBM PC: ч.2: Аппаратные средства ПЭВМ -М.: ТПП "Сфера", 1991. - 302 с.
19. Майко Г.В. Ассемблер для IBM PC: - М.: "Бизнес-Информ", "Сирин" 1999 г. - 212 с.
20. Соловьев Г.Н., Никитин В.Д. Операционные системы ЭВМ., М., Выш. Школа., 1989.
21. Краковяк С. Основы организации и функционирования ОС ЭВМ. М., Мир., 1989.

22.Столингс В. Операционные системы., М., Санк-Пет., К. 2002.
23.Дейтел Т. Введение с операционные системы. М.. Мир., 1987.
24. Шевчук Р.П. Опорний конспект лекцій з дисципліни „Сучасні інструментальні засоби розробки користувацького інтерфейсу", для студентів за спеціальностями: 7.05010301 "Програмне забезпечення систем", 8.05010301- «Програмне забезпечення систем» Тернопіль. 2012. 103 с.

ІНТЕРНЕТ РЕСУРСИ

1. Guide to x86 Assembly [Електронний ресурс]. - Режим доступу : http://www.cs.virginia.edu/~evans/cs216/guides/x86.html (дата звернення: 2019-10-27). - Назва з екрана.
2. IntroX86 [Електронний ресурс]. - Режим доступу : http://opensecuritytraining.info/IntroX86.html (дата звернення: 2019-10-27). -Назва з екрана.
3. Ассемблер | сайт языка ассемблера - assembler.com.ua [Електронний ресурс]. -Режим доступу : http://assembler.com.ua/ (дата звернення: 2019-10-27). - Назва з екрана.
4. flat assembler [Електронний ресурс]. - Режим доступу : https://flatassembler.net/docs.php (дата звернення: 2019-10-27). - Назва з екрана.
5. NASM [Електронний ресурс]. - Режим доступу : https://www.nasm.us/ (дата звернення: 2019-10-27). - Назва з екрана.
6. The Yasm Modular Assembler Project [Електронний ресурс]. - Режим доступу : https://yasm.tortall.net/ (дата звернення: 2019-10-27). - Назва з екрана.
7. GitHub - x64dbg/XEDParse: XEDParse: A MASM-like, single-line plaintext assembler [Електронний ресурс]. - Режим доступу : https://github.com/x64dbg/XEDParse (дата звернення: 2019-10-27). - Назва з екрана.
8. GitHub - Maratyszcza/PeachPy: x86-64 assembler embedded in Python [Електронний ресурс]. - Режим доступу: https://github.com/Maratyszcza/PeachPy (дата звернення: 2019-10-27). - Назва з екрана.
9. no|wrap.de - Flasm [Електронний ресурс]. - Режим доступу : http://www.nowrap.de/flasm (дата звернення: 2019-10-27). - Назва з екрана.
10. x64dbg [Електронний ресурс]. - Режим доступу : https://x64dbg.com (дата звернення: 2019-10-27). - Назва з екрана.
11. OllyDbg v1.10 [Електронний ресурс]. - Режим доступу : http://www.ollydbg.de/ (дата звернення: 2019-10-27). - Назва з екрана.
12. WinDbg [Електронний ресурс]. - Режим доступу : http://www.windbg.org/ (дата звернення: 2019-10-27). - Назва з екрана.

13. x86 Assembly - Wikibooks, open books for an open world [Електронний ресурс]. - Режим доступу : https://en.wikibooks.org/wiki/X86_Assembly (дата звернення: 2019-10-27). - Назва з екрана.
14. x86 Disassembly - Wikibooks, open books for an open world [Електронний ресурс]. - Режим доступу : https://en.wikibooks.org/wiki/X86_Disassembly (дата звернення: 2019-10-27). - Назва з екрана.
15. GitHub - cirosantilli/x86-bare-metal-examples [Електронний ресурс]. - Режим доступу : https://github.com/cirosantilli/x86-bare-metal-examples (дата звернення: 2019-10-27). - Назва з екрана.
16. PC Assembly Book [Електронний ресурс]. - Режим доступу : https://pacman128.github.io/pcasm/ (дата звернення: 2019-10-27). - Назва з екрана.
17. Paul Hsieh's Assembly Lab [Електронний ресурс]. - Режим доступу : http://www.azillionmonkeys.com/qed/asmexample.html (дата звернення: 2019-10-27). - Назва з екрана.
18. Tutorial [Електронний ресурс]. - Режим доступу : http://www.easycode.cat/English/Tutorial.htm (дата звернення: 2019-10-27). -Назва з екрана.
19. Assembly Programming Tutorial - Tutorialspoint [Електронний ресурс]. - Режим доступу : https://www.tutorialspoint.com/assembly_programming/index.htm (дата звернення: 2019-10-27). - Назва з екрана.
20. int80h.org -- Unix Assembly Language Programming [Електронний ресурс]. -Режим доступу : http://www.int80h.org/ (дата звернення: 2019-10-27). - Назва з екрана.
21. Assembly Optimization Tips [Електронний ресурс]. - Режим доступу : http://mark.masmcode.com/ (дата звернення: 2019-10-27). - Назва з екрана.
22. Art of Assembly Language Programming and HLA by Randall Hyde [Електронний ресурс]. - Режим доступу : http://www.plantation-productions.com/Webster/www.artofasm.com/index.html (дата звернення: 201910-27). - Назва з екрана.
23. Programming from the Ground Up [Електронний ресурс]. - Режим доступу : http://programminggroundup.blogspot.com/ (дата звернення: 2019-10-27). -Назва з екрана.
24. Programming from the Ground Up [Електронний ресурс]. - Режим доступу : https://web.archive.org/web/20110708061040/http://programminggroundup.blogsp ot.com/ (дата звернення: 2019-10-27). - Назва з екрана.
25. Microchip Technology Inc Home. - www.microchip.com
26. MPLAB® Integrated Development Environment http://www.microchip.com/pagehandler/en-us/devtools/mplab/home.html
27. PIC Microcontrollers - Programming in C. - Milan Verle http://learn.mikroe.com/ebooks/ piccprogramming/
28. «Мікропроцесорна техніка»: конспект лекцій [Електронний ресурс]: навч. посіб. для студ. спеціальності 171 «Електроніка», спеціалізації «Електронні

пристрої і системи» / КПІ ім. Ігоря Сікорського ; уклад.: Т. О. Терещенко, О.В. Хоменко - Електронні текстові данні (1 файл: 3880 кбайт). - Київ : КПІ ім. Ігоря Сікорського, 2017. - 165с.

29. Операційні системи: [Електронний ресурс]: навч. посіб. для студ. спеціальності 123 «Комп'ютерна інженерія» / В. Г. Зайцев, І. П. Дробязко; КІЛ ім. Ігоря Сікорського. - Електронні текстові дані (1 файл: 3 Мбайт). - Київ: КПІ ім. Ігоря Сікорського, 2019. - 240 с.
30. Тео Мандел. Разработка пользовательского интерфейса -http: // citforum.ru / book/interf/interf_c. shtml
31. Бібліотека офіційної технічної документації для розробників під ОС Microsoft Windows -www.msdn.com

ЗМІСТ

Навчальне видання

Микола Дмитрович Василенко

Валерій Олександрович Рачук

Валерія Миколаївна Слатвінська

АНАЛІЗ ШКІДЛИВОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Навчально-методичні рекомендації

(для студентів факультету кібербезпеки та інформаційних технологій)

Українською мовою