

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

ЗАТВЕРДЖЕНО




Вченою радою ВНТУ

протокол № 12

від «30» березня 2017 р.

професор з наукової роботи

 С. В. Павлов

ПРОГРАМА

вступного іспиту до аспірантури за спеціальністю

121 – інженерія програмного забезпечення

галузь знань 12 – інформаційні технології

Розглянуто і схвалено

Секцією Науково-технічної ради ВНТУ

протокол № 5

від «22» лютого 2017 р.

голова секції НТР



О. В. Грушко

Вінниця 2017

Зміст програми.

1. Вступ.
2. Мета фахового вступного випробування.
3. Порядок складання випробування.
4. Перелік дисциплін.
5. Теми вступних фахових випробувань.
 - 5.1. Основи програмної інженерії.
 - 5.2. Архітектура комп'ютера.
 - 5.3. Архітектура програмного забезпечення
 - 5.4. Комп'ютерна дискретна математика.
 - 5.5. Об'єктно-орієнтоване програмування.
 - 5.6. Операційні системи.
 - 5.7. Системне програмування.
 - 5.8. Організація комп'ютерних мереж.
 - 5.9. Людино-машинна взаємодія.
 - 5.10. Моделювання та аналіз програмного забезпечення.
 - 5.11. Кодування та захист інформації.
 - 5.12. Бази даних.
6. Критерії оцінювання.
7. Список літератури.

1 ВСТУП

Програма фахового вступного випробування розроблена у відповідності з освітньо-професійною програмою «магістр» зі спеціальності 121 – Інженерія програмного забезпечення та навчальним планом і на основі загальних вимог до підготовки докторів філософії в аспірантурі Вінницького національного технічного університету за спеціальністю **121 - Інженерія програмного забезпечення.**

У програму фахового вступного випробування увійшли питання з фундаментальних та загально-інженерних дисциплін, які є базовими для навчання за освітньо-професійною програмою підготовки «доктор філософії» за спеціальністю **121 - Інженерія програмного забезпечення.** Кожен з екзаменаційних білетів включає теоретичні питання та практичні завдання, які охоплюють весь комплекс напрямків фундаментальної та загально-інженерної підготовки. При відповідях на них претендент повинен продемонструвати такий рівень фундаментальної підготовки, який дозволить йому успішно опанувати наступний рівень кваліфікації.

2 МЕТА ФАХОВОГО ВСТУПНОГО ВИПРОБУВАННЯ

Метою фахового вступного випробування є оцінка рівня знань по спеціалізації майбутніх докторів філософії зі спеціальності 121 – Інженерія програмного забезпечення, які повинні демонструвати теоретичні знання та практичні навички з напрямків:

- основні принципи та особливості створення програмних продуктів, вимоги до програмних продуктів;
- функціональні вузли комп'ютерів, організацію зв'язків між вузлами комп'ютерів;
- основні поняття алгебри логіки, теорії графів, алгоритми пошуку оптимального шляху;
- основні сучасні методи об'єктно-орієнтованого програмування та системного програмування, теоретичні та практичні питання, пов'язані з розробкою програм та їх експлуатацією;
- принципи організації та функціонування комп'ютерів, концепції утворення та функціонування операційних систем;
- основні поняття та принципи людино-машинної взаємодії, побудову інтерфейсів користувача;
- основи теорії кодування та захисту інформації;
- інформацію про структури даних, операції над ними, методи організації та управління базами даних.

3 ПОРЯДОК СКЛАДАННЯ ВИПРОБУВАННЯ

Фахові вступні випробування приймаються екзаменаційною комісією (ЕК), що складається з провідних викладачів кафедри програмного забезпечення.

Екзаменаційні білети містять теоретичні та практичні питання з перелічених нижче дисциплін (розділ 4).

При відповідях на теоретичні питання претендент повинен продемонструвати не тільки володіння науковим матеріалом, але й розуміння зв'язку теорії та практики.

4 ПЕРЕЛІК ДИСЦИПЛІН

1. Основи програмної інженерії.
2. Архітектура комп'ютера.
3. Комп'ютерна дискретна математика.
4. Об'єктно-орієнтоване програмування.
5. Операційні системи.
6. Системне програмування.
7. Організація комп'ютерних мереж.
8. Людино-машинна взаємодія.
9. Моделювання та аналіз програмного забезпечення.
10. Кодування та захист інформації.
11. Бази даних.

5 ТЕМИ ВСТУПНИХ ФАХОВИХ ВИПРОБОВУВАНЬ

1. Основи програмної інженерії

1. Модель життєвого циклу системи.
2. Методи і процеси стандартизації життєвого циклу.
3. Методи і засоби вдосконалення складних програмних систем
4. Профіль стандартів і основні цілі застосування.
5. Життєвий цикл профілів стандартів систем і програмних засобів.
6. Етапи розвитку і застосування профілів життєвого циклу.
7. Профіль життєвого циклу програмних засобів.
8. Управління проектами програмних засобів в системі - СММІ.
9. Моделі управління проектами програмних засобів
10. Стандарти, що регламентують процеси менеджменту проектів
11. Експертне техніко-економічне обґрунтування проектів програмних засобів.
12. Оцінка техніко-економічних показників проектів
13. Організація розробки вимог до складних програмних засобів.
14. Процеси розробки вимог до характеристик складних програмних засобів.
15. Структура основних документів, які відображають вимоги до програмних засобів
16. Організація планування життєвого циклу складних програмних засобів.
17. Завдання планів для забезпечення життєвого циклу складних програмних засобів.
18. Планування процесів управління якістю складних програмних засобів.

19. Завдання і особливості об'єктно-орієнтованого проектування програмних засобів.
20. Основні поняття і моделі об'єктно-орієнтованого проектування програмних засобів.

2. Архітектура комп'ютера.

1. Історичні аспекти розвитку комп'ютерів. Покоління ЕОМ. Їх характеристика.
2. Структура комп'ютера. Основні функціональні вузли комп'ютера. Основні характеристики функціональних вузлів комп'ютера.
3. Організація зв'язків між функціональними вузлами комп'ютера:
 - 3.1. Двошинна структура комп'ютера з обміном через процесор.
 - 3.2. Двошинна структура комп'ютера з обміном через пам'ять.
 - 3.3. Одношинна структура комп'ютера.
4. Архітектурні принципи Джона фон Неймана.
5. Ненейманівські архітектури комп'ютерів:
 - 5.1. Гарвардська архітектура.
 - 5.2. Дуальна пристонсько-гарвардська архітектура.
 - 5.3. Асоціативна машина.
 - 5.4. Машина потоків даних.
 - 5.5. Паралельні комп'ютерні системи.
6. Основні напрямки використання комп'ютера.
7. Класифікація комп'ютерів залежно від сфер застосування та технічних характеристик.
8. Ієрархічна природа комп'ютера. Багаторівнева організація комп'ютера. Характеристика семи рівнів організації комп'ютера за Е. Таненбаумом.
9. Поняття про системи числення. Двійкова, вісімкова, шіснадцяткова системи числення. Методи переведення чисел з однієї системи числення в іншу.
10. Подання інформації в ЕОМ із фіксованою та плаваючою комою. Кодування чисел у двійковій системі числення. Прямий, обернений та доповняльний коди числа.
11. Правила додавання двійкових чисел.
12. Класифікація методів додавання двійкових чисел.
13. Методи простого множення на суматорі прямого коду.
14. Методи простого множення на суматорі доповняльного коду.
15. Методи прискореного множення двійкових чисел.
16. Методи ділення двійкових чисел.
17. Прискорений метод ділення у двійковій системі числення.
18. Канонічний метод структурного синтезу автомата з жорсткою логікою за В.М.Глушковим.
19. Аналіз етапів синтезу автомата Мура (з жорсткою логікою).
20. Аналіз етапів синтезу автомата Мілі (з жорсткою логікою).

21. Принципи розробки мікропрограми для керуючого автомата з програмованою логікою з примусовою адресацією.
22. Принципи розробки мікропрограми для керуючого автомата з програмованою логікою з природною адресацією.
23. Структурна схема операційного пристрою.
24. Структура, призначення та особливості реалізації арифметико-логічного пристрою (АЛП).
25. Запам'ятовувальні пристрої ЕОМ.

3. Комп'ютерна дискретна математика.

1. Скінченна множина. Об'єкти множин. Потужність множини. Способи задання множини.
2. Геометрична інтерпретація множин. Діаграми Ейлера-Венна.
3. Операції на множинах: перетин, об'єднання, доповнення (заперечення), різниця. Властивості операцій над множинами. Пріоритет операцій.
4. Симетрична різниця двох множин.
5. Декартовий добуток множин. Декартовий степінь множини.
6. Бінарні відношення. Способи задання бінарних відношень: списком, матричним способом, графом.
7. Операції над відношеннями: обернене відношення; композиція відношень; степінь відношення; переріз відношення; фактор множина.
8. Властивості бінарних відношень (приклади зображення таких відношень списком, графічно та матрицею): рефлексивність; антирефлексивність;
9. симетричність; асиметричність; антисиметричність; транзитивність; антитранзитивність.
10. Відношення еквівалентності. Відношення толерантності.
11. Відношення порядку: відношення часткового (нестрогого) порядку; відношення строгого порядку.
12. Функціональні відношення (навести приклади).
13. Область визначення та область значень відношення.
14. Види відображень: сюр'єктивне відображення; ін'єктивне відображення; бієктивне відображення.
15. Основи алгебри логіки. Таблиці істинності найпростіших булевих функцій. Логічні функції двох змінних.
16. Аксиоми та закони алгебри логіки. Мінімізація логічних функцій за законами алгебри логіки.
17. Особливості приведення логічних функцій до базисів І-НІ та АБО-НІ. Синтез комбінаційних схем.
18. ДДНФ та ДКНФ логічних функцій.
19. Комбінаторика. Правило суми. Правило добутку.
20. Комбінаторика. Розміщення і перестановка без повторень і з повтореннями.
21. Комбінаторика. Сполучення без повторень і з повтореннями.

22. Терміни теорії графів: Граф. Плоский (планарний) граф. Суміжні вершини і ребра. Інцидентні вершини і ребра. Ізольована вершина. Петля. Паралельні (кратні) ребра. Маршрут. Замкнений маршрут. Ланцюг. Простий ланцюг. Цикл. Простий цикл. Зв'язаний граф. Підграф.
23. Терміни теорії графів: Компонента зв'язності. Точка зчленування графа. Степінь вершини. Порожній граф. Простий граф. Повний граф. Регулярний (однорідний) граф. Дерево графа. Відстань між вершинами графа. Ексцентриситет. Діаметр графа. Радіус графа. Центральна вершина графа. Центр графа.
24. Матриця суміжності орієнтованого і неорієнтованого графа. Матриця інцидентій графа.
25. Алгоритм Дейкстри пошуку оптимального шляху за графом.
26. Алгоритм Флойда пошуку оптимального шляху за графом.

4. Об'єктно-орієнтоване програмування.

1. Як пов'язані мова програмування Сі та С++? Чим якісно відрізняється мова програмування С++ від Сі? Чому використання мови програмування С++ дозволяє створювати набагато більші програми в порівнянні з мовою програмування Сі?
2. Що таке об'єкт? Що таке стан об'єкту? Якими засобами мови С++ він реалізується? Що таке поведінка об'єкта. Якими засобами мови С++ вона реалізується? Проілюструйте співвідношення "клас-об'єкт" у С++.
3. Як оголошується клас у мові програмування С++ ? Як визначати методи класу на С++ ПОЗА оголошенням класу і для чого це потрібно? Як визначити об'єкт певного класу? Як створити локальний об'єкт, глобальний і статичний? Де зберігаються об'єкти кожного типу, де їхня область видимості, який час життя? Як робиться доступ до елементів (змінних та функцій) класу? Як визначаються функції - елементи класу (по іншому - методи) ? Різні типи доступу до членів класу: `private`, `public`.
4. Використання оператора `::`. Як розв'язати конфлікт імен – між аргументом функції, змінною-елементом класу та глобальною змінною – тобто як здійснювати доступ до них у випадку, коли їх імена співпадають? Вказівник `this` – для чого потрібно, звідки береться? Наведіть приклад коду програми.
5. Конструктори та деструктори. Оператори `new` и `delete`. Статичне та динамічне створення об'єктів. Цикл життєдіяльності статично створеного об'єкта. Цикл життєдіяльності динамічно створеного об'єкта.
6. Аргументи функції по замовчуванню. Конструктор по замовчуванню.
7. Що означає термін "поверхнєве копіювання"? Що означає термін "глибинне копіювання"? Що означає термін "копіюючий конструктор"? Три випадки виводу? Як перевантажити оператор `=`?
8. Як створюється динамічний масив об'єктів першого типу? Як знищується? Як створюється динамічний масив об'єктів другого типу? Як знищується?

9. Як перевантажити унарний оператор (два способи)? Як перевантажити бінарний оператор(два способи)?
10. Чому `operator<< (ostream &ost, ...)` перевантажується виключно зовнішньою функцією?
11. Що таке ініціалізатор конструктора? Які два вида елементів класів можуть бути проініціалізовані ТІЛЬКИ у ініціалізаторі конструктора?
12. Елементи класу, які створюються динамічно – і код який створює (наприклад у конструкторі) їх та знищує (наприклад у деструкторі)
13. Friend-функції та класи. Статичні елементи класу – змінні та функції. Перевантаження функцій у мові C++. Чим відрізняється оголошення від визначення у мові C++ (продемонструйте на прикладі)
14. Як працює директива `extern "C" {};`
15. Що таке ієрархія. Магічне число 7+-2. Наслідування класів.
16. Віртуальні функції. Поліморфізм. Чим принципово відрізняється поліморфізм від перезавантаження функцій. Посилання (&)
17. Навіщо потрібні віртуальні деструктори? Для чого використовується `dynamic_cast<>?`
18. Як при наслідуванні здійснити виклик конструктора класа-предка. Як здійснити при потребі виклик до метода класа предка (особливо якщо в класі нащадку є метод з таким самим ім'ям та аргуменами)
19. Що таке абстрактні класи? Що таке чисто віртуальна функція?
20. Що таке інтерфейс (з точки зору ООП)?
21. Властивості типу доступу `protected`.
22. UML. Співвідношення між класами: успадкування. Проілюструйте невеличкою програмою.
23. UML. Співвідношення між класами: агрегація. Проілюструйте невеличкою програмою.
24. UML. Співвідношення між класами: композиція. Проілюструйте невеличкою програмою.
25. UML. Співвідношення між класами: асоціація. Проілюструйте невеличкою програмою.
26. UML. Діаграми кооперації. Проілюструйте невеличкою програмою.
27. UML. Діаграми послідовності. Проілюструйте невеличкою програмою.
28. UML. Діаграми стану. Проілюструйте невеличкою програмою.
29. Пояснити відмінність механізму `overloading` від механізму `щмуккшвштп`
30. Статичний та динамічний поліморфізм
31. Принцип ООП: `Single responsibility principle (SRP)`
32. Принцип ООП: `Open/closed principle (OCP)`
33. Принцип ООП: `Liskov substitution principle (LSP)`
34. Принцип ООП: `Interface segregation principle (ISP)`
35. Принцип ООП: `Dependency inversion principle (DIP)`

5. Операційні системи.

1. Структура і класифікація комп'ютерних систем
2. Комп'ютерні архітектури і їх класифікація
3. Операційні системи і їх класифікація.
4. Поняття "Адресний простір", процес розподілу.
5. Сегментна структура програм.
6. Коротка характеристика способів адресації.
7. Структура і образ пам'яті програми .COM.
8. Взаємодія між процесами.
9. Взаємне виключення з активним очікуванням.
10. Вирішення проблеми переповненого буфера за допомогою семафора
11. Основні поняття планування процесів
12. Структура багато -потокowych застосувань.
13. Обробка повідомлень у багато - потокowych застосуваннях.
14. Структура призначеного для користувача інтерфейсу.
15. Особливості і компоненти графічного інтерфейсу
16. Основні аспекти розробки інтерфейсу
17. Оцінка і поліпшення програмного інтерфейсу
18. Проектування засобів підтримки користувача
19. Проектування довідкової системи
20. Документація користувача.

6. Системне програмування.

1. Предмет дисципліни «Системне програмування». Системна і прикладна програма. Компоненти системного програмного забезпечення. Загальні поняття про мови системного програмування. Класифікація і покоління операційних систем (ОС). Склад і загальна схема функціонування ОС. Режими роботи.
2. Системне програмування мовою асемблер в режимі командного рядка Windows. Однопрограмна ОС MS DOS та режим командного рядка Windows. Модель пам'яті в режимі командного рядка. Внутрішні і зовнішні команди в режимі командного рядка. Командні файли і файли конфігурації. Команди керування виконанням *.BAT файлів та конфігураційні команди.
3. Модель мікропроцесора для програміста. Сегментація пам'яті. Сегмент даних. Сегмент команд. Сегмент стека. Формування фізичної адреси. Режими адресації процесора: регістрова, безпосередня, пряма, непряма, базова, індексна, відносна. Система команд процесора та мова Асемблер.
4. Оператори та директиви асемблера. Система команд. Завдання простих змінних. Структура програм. Складання програми лінійної структури. Команди передачі даних та роботи зі стеком. Команди арифметичних та логічних операцій. Команди роботи з ланцюжками. Команди умовних, безумовних переходів та виклику процедур.

5. Макрозасоби мови асемблер, компанування та завантаження програм. Особливості зв'язку багатомодульних програм, написаних мовою Асемблера. Функції компоновщика та завантажувача. Типи завантажувачів. Розбиття програми на зовнішні процедури. Асемблювання і компіляція програм.
6. Порівняльна характеристика модулів типу .COM та .EXE, префікс програмного сегменту (PSP). Апаратні і програмні переривання. Таблиця векторів переривань. Структура програми обробки переривання. Програмні переривання BIOS і DOS.
7. Переривання від системного таймера – int 08h, int 1ch. Керування клавіатурою і дисплеєм – переривання int 10h, int 16h, int 21h. Введення-виведення через послідовний і паралельний порти – переривання int 14h, int 17h.
8. Файлова система FAT. Керування дисковою пам'яттю та робота з файлами – переривання int 13h, int 21h, int 25h, 26h.
9. Системне програмування в багатозадачних ОС. Керування пам'яттю в багатопрограмних ОС. Види організації віртуальної пам'яті.
10. Керування процесами. Синхронізація паралельних процесів. Керування процесорами та зовнішньою пам'яттю в багатозадачних ОС.
11. Керування пристроями в багатозадачних ОС. Захист інформації від несанкціонованого доступу в операційних системах.
12. Захищений режим процесорів i486 і старших. Керування пам'яттю в процесорах i486 і старших. Керування пам'яттю в Windows.
13. Багатозадачність та багатопотоковість в Windows. Синхронізація потоків в Windows. Керування пристроями в Windows. Файлові систем Windows - FAT 32, NTFS.
14. Системне програмування в середовищі Windows мовою C++. Принципи програмування в Windows. Структурне та об'єктно-орієнтоване програмування (ООП). DLL бібліотеки, API функції.
15. Найпростіший консольний і графічний додаток в Windows мовою C++. Засоби Windows (API-функції) для роботи з файлами, каталогами і реєстром.
16. API для керування пам'яттю в Windows, відображення файлів в адресний простір процесу і бібліотеки DLL.
17. API для керування процесами і потоками в середовищі Windows. Обробка виключень в Windows.
18. Мережеве програмування в Windows. Сокети.
19. Служби Windows Services. API для створення servісу Windows
20. Програмування в середовищі Win64. Перенесення існуючого коду на платформу Win64.
21. UNIX-подібні операційні системи та особливості системного програмування. Огляд архітектури UNIX. Керування пам'яттю і процесами в UNIX. Файлова система UNIX (LINUX). Співставлення функцій Windows, UNIX і бібліотеки C.

7. Організація комп'ютерних мереж.

1. Режими передачі даних: симплексний, напівдуплексний та повний дуплексний. Що означає термін мережева операційна система? Які топології комп'ютерних мереж існують. Наведіть їх плюси та мінуси.
2. Характеристика мережі "Відмовостійкість". Як проявляється, за рахунок чого досягається? Характеристика мережі "Масштабованість". Як проявляється, за рахунок чого досягається? Характеристика мережі "Якість обслуговування" (Quality of service). Як проявляється, за рахунок чого досягається? Характеристика мережі "Безпека". Як проявляється, за рахунок чого досягається?
3. Локальні мережі. Основні характеристики. Глобальні мережі. Основні характеристики. Взаємодія в однорангових мережах (peer-to-peer). Взаємодія в клієнт-серверних мережах. Що означають терміни "Маршрутизація" та "Коммутація"? Що означають терміни "Мультиплексування" та "Демультиплексування"?
4. Характеристики з'єднання шляхом комутації каналів. Переваги та недоліки. Характеристики з'єднання шляхом комутації пакетів. Переваги та недоліки. У чому полягає датаграмний спосіб передачі?
5. Семирівнева модель взаємодії відкритих систем OSI. Загальна характеристика. Які рівні входять до складу? Прикладний рівень семирівневої моделі взаємодії відкритих систем OSI. Основні функції. Рівень представлення семирівневої моделі взаємодії відкритих систем OSI. Основні функції.
6. Сеансовий рівень семирівневої моделі взаємодії відкритих систем OSI. Основні функції. Транспортний рівень семирівневої моделі взаємодії відкритих систем OSI. Основні функції. Мережевий рівень семирівневої моделі взаємодії відкритих систем OSI. Основні функції.
7. Канальний рівень семирівневої моделі взаємодії відкритих систем OSI. Основні функції. Фізичний рівень семирівневої моделі взаємодії відкритих систем OSI. Основні функції.
8. Стек протоколів TCP/IP. З чого складається, які функції? Як розшифровується і для чого використовується DNS? Як розшифровується і для чого використовується Telnet? Як розшифровується і для чого використовується SMTP? Як розшифровується і для чого використовується IMAP?
9. Як розшифровується і для чого використовується POP3? Як розшифровується і для чого використовується DHCP? Як розшифровується і для чого використовується HTTP? Як розшифровується і для чого використовується FTP?
10. TCP Протокол. Основні характеристики. Формат сегменту. Транспортний рівень. Розподіл портів. Як розглядається приналежність портів до каналу обміну інформацією. TCP Протокол в клієнт-серверній моделі. Встановлення з'єднання (three-way handshake). Завершення з'єднання. Підтвердження даних та питання "ковзаючого вікна" (window)
11. UDP Протокол. Формат пакету.

12. Протокол IPv4. Основні характеристики. Формат пакету. Що означає "Протокол IPv4 – протокол без встановлення з'єднання"? Що означає "Протокол IPv4 – протокол першої найкращої спроби (best effort)"? Що означає "Протокол IPv4 – протокол, що не залежить від середовища передачі даних"? Протокол IPv4: розбиття мережі на під-мережі. Чотири принципа розподілу
13. Як роутери дозволяють підвищити продуктивність мережі шляхом сегментації broadcast-доменів. Розбиття комп'ютерів на окремі мережі – з точки зору безпеки. Розбиття комп'ютерів на окремі мережі – з точки зору керування адресацією.
14. Шлюзи – вихід у глобальну мережу. Що собою представляє шлях у мережі. Таблиця маршрутизації на хості. Таблиця маршрутизації. Шлях по замовчуванню.
15. Протокол IPv4. Три випадки прийняття рішення роутером. Протоколи маршрутизації – обмін інформацією про мережі. Таблиці маршрутизації. Статична маршрутизація. Динамічна маршрутизація.
16. Протокол IPv4. Що собою представляє адреса? Обчислення адреси мережі, комп'ютерів та broadcast повідомлень. Що означають терміни unicast, multicast, broadcast? Зарезервовані адреси. Public та private адреси. Спеціальні адреси.
17. Історичні класи мереж. Планування присвоєння адрес у організації. Статичні та динамічні адреси на кінцевих комп'ютерах.
18. Протокол IPv4. Маска підмережі. Основи створення підмереж. IPv4. Визначення правильного розміру підмереж. Як визначити мережеву адресу?
19. Як визначити кількість хостів? Як визначити коректні адреси для хостів? Визначення з'єднання з 1)шлюзом 2)віддаленим комп'ютером?
20. Протокол ICMP v4 – для тестування та передачі повідомлень.
21. Навіщо потрібні програми - ping, tracer, Packet Tracer, netstat.
22. Протокол IPv6. Основні характеристики. Формат пакету. Фрагментація IP пакетів. Яка функція дозволяє визначити ip-адресу на комп'ютері, на якому запущено програму?
23. Як задається адреса приймача у протоколі TCP/IP? Як створити на сервері серверний сокет, який використовується для прослухування мережі і отримування запитів на з'єднання від нових клієнтів? Як на сервері організовується сокет для обміну даними? Як здійснюється обмін даними через створені сокети TCP/IP?
24. Для чого використовується функція socket? Для чого використовується функція bind? Для чого використовується функція listen? Для чого використовується функція connect?
25. Для чого використовується функція accept? Для чого використовується функція send? Для чого використовується функція recv? Для чого використовується функція sendto? Для чого використовується функція recvfrom?
26. Для чого використовується функція shutdown? Для чого використовується функція htons? Для чого використовується функція htonl? Для чого використовується функція ntohs? Для чого використовується функція ntohl?

27. Для чого використовується функція `ioctlsocket`? Для чого використовується функція `inet_ntoa`? Для чого використовується структура `SOCKADDR_IN`?
28. Для чого використовується функція `gethostbyname`? Для чого використовується функція `gethostname`? Для чого використовується функція `inet_pton`?
29. Як зробити сокет блокуючим? Як зробити сокет НЕблокуючим? Як здійснювати обмін інформацією через блокуючий сокет? Як здійснювати обмін інформацією через НЕблокуючий сокет?
30. Що означають терміни `little-endian` `big-endian`?

8. Людино-машинна взаємодія.

1. Основні поняття про людино-машинну взаємодію. Різновиди людино-машинних систем. Роль та місце людини в контурі людино-машинних систем. Підхід програмної інженерії до людинно-машинної взаємодії.
2. Значення інтерфейсу користувач-комп'ютер. Інтерфейс користувач-комп'ютер та його зручність. Стратегії розробки інтерфейсу користувач-комп'ютер. Критерії оцінки якості діалогу. Поняття про юзабіліті.
3. Елементи когнітивної психології. Когнетіка. Когнітивне свідоме і когнітивне несвідоме. Свідомість і моделі людського розуму. Локус уваги. Одночасне виконання завдань.
4. Компоненти інтерфейсу користувач-комп'ютер. Процеси введення. Процеси діалогу. Класифікація повідомлень в діалозі. Вхідні повідомлення. Вихідні повідомлення.
5. Структури діалогу. Структура діалогу типу запитання-відповідь. Структура діалогу типу "меню". Структура діалогу на основі екранних форм. Структура командного типу діалогу. Інтелектуальні інтерфейси.
6. Основи проектування інтерфейсу користувача. Підтримка користувача. Розміщення інформації на екрані, шаблони. Час відповіді системи. Види адаптації в діалогових системах. Опис діалогу: макети екрану, мережі переходів, продукційні системи.
7. Уніфікована мова моделювання UML та її використання при розробці інтерфейсу людина-комп'ютер. Базові діаграми UML.
8. Багатовіконні WIMP-інтерфейси. Основні аналоги. Панелі і вікна. Уніфіковані дії діалогу. Допоміжні функції. Переваги і недоліки WIMP-інтерфейсів. Критерії проектування WIMP-інтерфейсів. Стандарт CUA фірми ІВМ.
9. WEB-орієнтований інтерфейс. Технічні особливості. Структура WEB-додатку. WEB 2.0 та WEB 3.0. Технологія Ajax.
10. «Metro» - новий підхід фірми Microsoft до проектування інтерфейсів користувача десктопних та WEB-додатків.

11. Міжнародні стандарти, що відносяться до людино-машинної взаємодії та юзабіліті. ISO 9241, ISO 13407, ISO 18529, ISO 14915, ISO 16071, ISO 16982, ISO 20282.
12. Засоби розробки інтерфейсів десктопних додатків. Проектування багатовіконного графічного інтерфейсу в середовищі Microsoft Visual Studio та Borland. Проект Windows Forms Application.
13. Засоби розробки WEB-інтерфейсів. Розробка інтерфейсів клієнтської частини WEB-додатку. Мова розмітки тексту HTML. Мова програмування JavaScript, аплети Java, флеш-технології. Серверні засоби: PHP, Java-технології, технологія ASP.NET.
14. Квантифікація. Кількісний аналіз інтерфейсу. Інформаційна продуктивність інтерфейсу. Модель швидкості друку GOMS. Часові інтервали в інтерфейсі. Розрахунки за моделлю GOMS. Закон Фітса і закон Хіка.
15. Проблеми і тенденції розвитку людино-машинного інтерфейсу. Візуалізація даних: візуальний інтерфейс для систем підтримки прийняття рішень, OLAP-технології. Системи підтримки роботи в групі: групова робота в локальних і глобальних мережах, системи семінарів, робота з фреймами і мультидоступ, питання синхронізації групової взаємодії.
16. Мультимедіа середовища і мультисенсорні системи: мовний інтерфейс, звукові сигнали, розпізнавання текстів, анімація і відеофрагменти, Мультимедіа середовища і мультисенсорні системи: мовний інтерфейс, звукові сигнали, розпізнавання текстів, анімація і відеофрагменти, розпізнавання жестів, комп'ютерний зір. Технології віртуальної реальності. Керування за допомогою думки.

9. Моделювання та аналіз програмного забезпечення.

1. Створення та використання шаблонів функцій. Створення та використання шаблонів класів. Що означає “конструктор по замовчуванню”?
2. Як здійснити поверхнєве копіювання об'єктів. Як здійснити глибинне копіювання об'єктів. Копіюючий конструктор: три випадки виклику. Конструктор перетворення типу: три випадки виклику. Перевантаження оператора =, який реалізує глибинне копіювання.
3. Перевантажений оператор << для виведення на ostream. Перевантаження унарних операторів: як функція-елемент класу та як зовнішня функція. Перевантаження бінарних операторів: як функція-елемент класу та як зовнішня функція. Використання оператора перетворення типу. Використання індексатору (перевантажений оператор [])
4. Що таке функтор? Що таке предикат (звичайний та предикат – арифметична функція)? Що таке предикат, створений користувачем?
5. Використання послідовного контейнера vector : створення, додання елементів (push_back), перегляд елементів (з допомогою ітератора та з допомогою індексатора), визначення розміру (size), резервування місця, зміна розміру контейнеру (resize) , сортування (sort) .

6. Використання послідовного контейнера `list` : створення, додавання елементів (`push_back`, `insert`), перегляд елементів (з допомогою ітератора), визначення розміру (`size`), видалення елементів (`remove`, `erase`, `clear`), перенесення елементів (`splice`, `merge`), сортування (`sort`), забезпечення унікальності (`unique`).
7. Порівняти контейнери `list` та `vector`.
8. Використання асоціативного контейнеру `map`: створення, додавання елементів (через `[]` та через `insert`), перегляд елементів з допомогою ітератора, пошук ключа (`find`), видалення пари по ключу(`erase`). Використання асоціативного контейнеру `multimap`: створення, додавання елементів (`insert`), перегляд елементів з допомогою ітератора, пошук ключа (`find`, `equal_range`, `lower_bound`, `upper_bound`), видалення пари по ключу(`erase`).
9. Використання асоціативного контейнеру `set`: додавання елементів, перегляд, вилучення елементів. Використання асоціативного контейнеру `multiset`: додавання елементів, перегляд, вилучення елементів.
10. Використання асоціативного контейнеру `queue`: додавання елементів, перегляд, вилучення елементів. Використання асоціативного контейнеру `priority_queue`: додавання елементів, перегляд, вилучення елементів.
11. Використання асоціативного контейнеру `dequeue`: додавання елементів, перегляд, вилучення елементів. Використання асоціативного контейнеру `stack`: додавання елементів, перегляд, вилучення елементів. Як функціонує блок `try-catch`?
12. Як сгенерувати один з системних винятків? Як сгенерувати свій власний виняток? Як зробити перезапущ винятку?
13. Для чого потрібен `find`? Для чого потрібен `find_if`? Для чого потрібен `for_each`? Для чого потрібен `mismatch`? Для чого потрібен `equal`? Для чого потрібен `count`? Для чого потрібен `count_if`?
14. Для чого потрібен `transform` над однією послідовністю? Для чого потрібен `transform` над двома послідовностями? Для чого потрібен `copy`? Для чого потрібен метод `resize` контейнера `vector`? Для чого потрібен `copy_backward`?
15. Для чого потрібен `greater<int>`? Як відсортувати `list`? Як створити `vector` одразу з декількома значеннями? Для чого потрібен `sort`? Для чого потрібен `distance`? Для чого потрібен `unique`?
16. Для чого потрібен `istream_iterator<>`? Для чого потрібен `istreamstringstream`? Для чого потрібен `front_inserter`? Для чого потрібен `back_inserter`? Для чого потрібен `insert_iterator`? Для чого потрібен `set_symmetric_difference`? Для чого потрібні `make_heap` `push_heap` `pop_heap`?
17. C++11 : які переваги надає використання `nullptr` ? Які переваги надає використання `auto` ? Навіщо потрібне ключове слово `shared_ptr`? Наведіть приклади використання. Навіщо потрібне ключове слово `unique_ptr`? Наведіть приклади використання.
18. C++11 : продемонструйте використання цикла `for` заснованого на діапазоні. Використання лямбда-вираза в якості функтора.
19. Хвостовий специфікатор типа повертаємого значення (suffix return type syntax). Як вказати тип результату лямбда-виразу? Як лямбда-вираз може

використати змінні з зовнішнього контекста (за значенням)? Як лямбда-вираз може використати змінні з зовнішнього контекста (за значенням) і змінювати їх? Як лямбда-вираз може використати змінні з зовнішнього контекста (за посиланням)

20. Детально шаблон проектування Одинак (Singleton) – призначення, реалізація у C++.
21. Детально шаблон проектування Фабричний Метод (Factory Method) – призначення, реалізація у C++.
22. Детально шаблон проектування Абстрактна Фабрика (Abstract Factory) – призначення, реалізація у C++.
23. Детально шаблон проектування Будівельник (Builder) – призначення, реалізація у C++. Детально шаблон проектування Легковаговик (Flyweight) – призначення, реалізація у C++.
24. Коротко про шаблон проектування Прототип (Prototype). Коротко про шаблон проектування Adapter.
25. Коротко про шаблон проектування Bridge. Коротко про шаблон проектування Composite. Коротко про шаблон проектування Decorator
26. Коротко про шаблон проектування Facade. Коротко про шаблон проектування Proxy. Коротко про шаблон проектування Command. Коротко про шаблон проектування Chain of Responsibility.
27. Коротко про шаблон проектування Interpreter. Коротко про шаблон проектування Iterator.
28. Коротко про шаблон проектування Mediator. Коротко про шаблон проектування Memento. Коротко про шаблон проектування Observer
29. Коротко про шаблон проектування State. Коротко про шаблон проектування Strategy.
30. Коротко про шаблон проектування Template Method. Коротко про шаблон проектування Visitor

10. Кодування та захист інформації.

1. Інформація та її кількісні оцінки. Визначення інформації. Інформація, повідомлення, сигнал. Класифікація повідомлень. Кількісне оцінювання інформації – міри Шеннона, Хартлі, алгоритмічний підхід.
2. Властивості ентропії. Умовна ентропія і її властивості – ентропія об'єднання двох статистично залежних ансамблів, ентропія об'єднання декількох статистично залежних ансамблів.
3. Кількість інформації як міра знятої невизначеності – апіорна ймовірність, апостеріорна ймовірність, середня кількість інформації, яка міститься в кожному прийнятому елементі повідомлення при наявності завад.
4. Модель системи передачі. Кодування джерела інформації та кодування каналу. Моделі джерел дискретних повідомлень. Міра надлишковості. Продуктивність джерела дискретних повідомлень.

5. Моделі дискретних каналів. Основні характеристики дискретного каналу - швидкість передачі інформації по дискретному каналу, пропускна здатність дискретного каналу без завад і з завадами.
6. Коди фіксованої і змінної довжини. Нерівність Крафта та Макміллана. Укорочені блокові коди. Деякі коди змінної довжини - гамма і дельта-коди Еліаса, коди Райса і Голомба, коди Івен-Роде, старт-крок-стоп (start-step-stop) коди, коди Фібоначі.
7. Основна теорема Шенона для кодування каналу без завад. Ефективне кодування (ущільнення) некорельованої послідовності знаків методом Шеноно-Фано. Класифікація методів оптимального кодування.
8. Алгоритм RLE. Кодування Хаффмана. Кодування за ступенем новизни. Ймовірнісне кодування.
9. Огляд словникових методів оптимального кодування. Алгоритм LZW.
10. Арифметичне кодування. Арифметичне кодування та нові підходи до оптимального кодування: концепція універсального моделювання і кодування Рісанена-Ленгдона, методи контекстного моделювання.
11. Ущільнення на основі перетворень без втрат. BWT-перетворення. Ущільнення зображень і звуку з втратами.
12. Класифікація завад. Теорема Шеннона про кодування каналу з завадами. Принципи завадостійкого кодування. Зв'язок корегуючої здатності коду з кодовою відстанню. Граничні співвідношення між параметрами завадостійких кодів.
13. Класифікація завадостійких кодів. Лінійні блокові коди. Способи подання лінійних блокових кодів. Деякі блокові коди для виявлення і виправлення помилок - з перевіркою на парність, коди з повторенням.
14. Коди Хеммінга. Побудова блокових кодів з заданою виправною здатністю. Циклічні коди.
15. Неперервні коди для виявлення і виправлення помилок. Згортні коди. Поняття про згортку. Породжувальні поліноми для згортних кодів. Систематична форма згортного коду. Кодування з використанням імпульсної характеристики.
16. Діаграма станів кодера. Гратчаста діаграма згортного кодера. Кодування згортними кодами з використанням гратчастої діаграми.
17. Класифікація методів декодування згортних кодів. Декодування згортних кодів з використанням алгоритму Вітербі.
18. Криптологія – основні означення і терміни. Теорема Шеннона про зв'язок в секретних системах. Основні принципи криптографічного захисту. Класифікація методів криптографічного захисту.
19. Симетричні криптосистеми – шифри Віжінера, Цезаря, Вернама; гамування; стандарт шифрування DES; сучасні стандарти шифрування. Ущільнення даних і криптографічний захист інформації.
20. Асиметричні криптосистеми. Системи з відкритим ключем. Елементи теорії чисел – теорема Ферма, теорема Ейлера, алгоритм Евкліда, алгоритми піднесення до степеня. Алгоритм шифрування RSA.
21. Комп'ютерна стеганографія та візуальна криптографія.

22. Програмне забезпечення і інформаційна безпека. Огляд сучасного програмного забезпечення. Помилки, що приводять до можливості атак на інформацію. Основні положення по розробці ПЗ. Організація захисту від комп'ютерних вірусів.

11. Бази даних.

1. Визначення і класифікація інформаційних систем.
2. Автоматизовані банки даних.
3. Вимоги до банків даних.
4. Принципи побудови банків даних.
5. Інфологічна модель даних. Мета та задачі інфологічного моделювання.
6. Трьохрівнева архітектура процесу проектування БД.
7. Загальна характеристика деревовидних моделей даних.
8. Загальна характеристика мережевого принципу подання інформації.
9. Загальна характеристика реляційних моделей даних.
10. Основні операції реляційної алгебри: проекція, об'єднання, різниця, селекція, декартовий добуток, перетин, частка відношень, спеціальні реляційні операції.
11. Нормалізація баз даних як один з основних етапів їх проектування.
12. Індексні методи доступу до даних.
13. Мультиспискові методи доступу до даних. Інвертований метод доступу до даних.
14. Адресні методи доступу до даних.
15. Побудова КЕШ-функції в адресних методах доступу до даних.
16. Моделі подання знань з використанням логіки предикатів, їх переваги та недоліки.
17. Моделі подання знань з використанням правил продукції, їх переваги та недоліки.
18. Продукційні системи Поста.
19. Загальна характеристика семантичної мережі подання знань. Недоліки і переваги семантичних мереж.
20. Фреймові моделі подання знань. Структура фрейму та його властивості.
21. Переваги і недоліки фреймових моделей знань.
22. Реалізація запитів мовою SQL.
23. Поняття транзакції та її властивості.
24. Управління паралельним виконанням транзакцій.
25. Методи відновлення бази даних.
26. Концепція розподілених баз даних. Основні поняття.
27. Принципи створення розподілених баз даних.
28. Фрагментація даних в розподіленій базі даних.
29. Реплікація даних в розподіленій базі даних.
30. Підтримка цілісності бази даних при виникненні перебоїв.

КРИТЕРІЇ ОЦІНЮВАННЯ

Критерії оцінювання знань студентів при складанні фахового вступного випробовування для навчання за освітньо-професійною програмою підготовки «доктор філософії» зі спеціальності 121 – Інженерія програмного забезпечення базуються на загальних «критеріях» оцінювання знань здобувачів вищої освіти на іспитах, які прийняті у ВНТУ.

Оцінка «5» відображує високу якість знань, вмінь та навичок і отримується в разі ґрунтовного викладення теоретичного матеріалу та розв'язання практичних задач. При цьому студент має розуміти фізичну суть процесів і явищ, наводити приклади та порівняння, володіти методикою формального доведення теоретичних положень, ілюструвати відповіді аналітичними залежностями і характеристиками процесів. Під час розв'язання практичних задач студент використовує методики аналізу і синтезу, застосовуючи комп'ютер. Під час відповіді студент проявляє ерудицію, чітко відповідає на будь-які запитання з основних дисциплін спеціальності і фундаментальної підготовки.

Оцінка «4» відображує знання, вміння та навички студента і отримується в разі викладення основ теоретичних питань та розв'язання практичних задач. Легко ліквідує помилки після зауважень членів ДЕК. Під час відповіді ілюструє обізнаність в теоретичних і практичних питаннях з основних дисциплін спеціальності, при цьому відповідає на більшість додаткових питань.

Оцінка «3» характеризує поверхове розуміння теоретичних положень та практичних задач, здатність формулювати власні висновки за допомогою викладача з найпростіших питань дисципліни, знайомство з основними методами розв'язанням задач.

Оцінка «2» характеризує знання окремих фрагментів навчального матеріалу, відсутність розуміння основних теоретичних положень, фізичної суті явищ та об'єктів, математичних методів даної проблемної області, неможливість впоратись з рішенням практичних задач.

СПИСОК ЛІТЕРАТУРИ

Основна

1. Дейтел Г. Введение в операционные системы: В 2-х т. Пер. с англ. / Г. Дейтел. – М.: Мир, 1987.
2. Соловьев Г.Н. Операционные системы ЭВМ / Г.Н. Соловьев, В. Д. Никитин. – М.: ВШ, 1989.
3. Руководство по программированию под управлением MS DOS/Ф. Пьеро и др./ Пер. с франц. – М.: Радио и связь, 1995.
4. Мак-Федрис П. Руководство по Windows 98. Пер. с англ. / П. Мак-Федрис. – М.: Бином, 1999.
5. Мюллер Дж. Полное руководство по Windows 95. Пер. с англ. / Дж. Мюллер, П. Нортон. - М.: Бином, 1998.
6. Джордейн Р. Справочник программиста персональных компьютеров. Пер. с англ. / Р. Джордейн. – М.: Финансы и статистика, 1992.
7. Ирвин К. Язык ассемблера для процессоров Intel / К. Ирвин. – М.: Вильямс, 2002.
8. Абель П. Язык ассемблера для IBM PC и программирование / П. Абель. – М.: ВШ, 1992.
9. Архітектура і програмування 32-х розрядних мікропроцесорів: Навчальний посібник/ А.М. Петух, В.П. Майданюк, Є.Л. Ольшевський - Вінниця: ВДТУ, 2000. - 130 с. Укр. мовою/
10. Керниган Б. Язык программирования Си. Пер. с англ. / Б. Керниган, Д. Риччи/ - М.: Финансы и статистика, 1992.
11. Страуструп Б. Язык программирования C++. Пер. с англ./ Б. Страуструп. – СПб: Бином, 2002.
12. Черносвитов А. Visual C++ 6 и MFC. Пер. с англ. / А. Черносвитов. – СПб: Питер, 2002.
13. Майданюк В.П. Системне програмування і операційні системи. Навчальний посібник / В. П. Майданюк, Г. Б. Ракитянська, В. А. Каплун. – Вінниця, ВНТУ, 2004. – 90 с.
14. Харт Дж. Системное программирование в среде Windows: Пер. с англ. / Дж. Харт. – М.: Издательский дом “Вильямс”, 2005. – 592 с.
15. Русинович М. Внутреннее устройство Microsoft Windows: Windows Server 2003, Windows XP, Windows 2000. Мастер - класс. Пер. с англ. / М. Русинович, Д. Соломон.–4-е изд.- СПб.: Питер, 2005. – 992 с.
16. Шеховцев В. А. Операційні системи / В. А. Шеховцев. – К.: BHV, 2005. – 574 с.
17. Руководство по Windows Vista – Корпорация Microsoft 2006. – 334 с.
18. Boyce J. Windows 7 Bible.- Published by Wiley Publishing, Inc. 10475, 2009. – 1251 с.
19. Пауэрс Л. Microsoft Visual Studio 2008 / Л. Пауэрс, М. Снелл: Пер. с англ. - БХВ-Петербург, 2009. — 1200 с.

20. Хортон А. Visual C++ 2010: полный курс / А. Хортон: Пер. с англ. – М.: ООО "И. Д. Вильямс", 2011. – 1216 с.
21. Основы теории информации и кодирования / И. В. Кузьмин, В. А. Кедрус. – К: Вища шк., 1986.-238с.
22. Дмитриев В. И. Прикладная теория информации / В.И. Дмитриев. – М.: Высш. шк., 1989. – 420 с.
23. Хэмминг Р.В. Теория кодирования и теория информации / Р.В. Хэмминг: Пер. с англ. – М.: Радио и связь, 1983. – 176 с.
24. Майданюк В. П. Кодування та захист інформації. Навчальний посібник / В. П. Майданюк. - Вінниця: ВНТУ, 2009. - 164 с.
25. Ватолин Д. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео / Д. Ватолин, А. Ратушняк, М. Смирнов, В. Юкин. – М.: ДИАЛОГ-МИФИ, 2002. – 384 с.
26. Методи і засоби КІТ. Кодування зображень. Навчальний посібник / В.П. Майданюк. – Вінниця, ВДТУ, 2001. – 63 с.
27. Харкевич А.А. Борьба с помехами / А. А. Харкевич. – М.: Наука, 1965. – 270 с.
28. Кларк Дж. Кодирование с исправлением ошибок в системах цифровой связи / Дж. Кларк, Дж. Кейн: Пер. с англ. – М.: Радио и связь, 1987. – 392 с.
29. Защита информации в персональных ЭВМ / А. В. Спесивцев и др. – М.: Радио и связь, 1992.
30. Соколов А. В. Защита от компьютерного терроризма / А.В. Соколов, О.М. Степанюк. – СПб.: БХВ-Петербург; 2002. – 496с.
31. Методичні вказівки до виконання лабораторних робіт з дисципліни “Операційні системи ЕОМ” /Укладач Майданюк В.П., Романюк О.Н. – Вінниця, ВДТУ, 1997. – 41 с.
32. Анин Б.Ю. Защита компьютерной информации / Б. Ю. Анин. – СПб: Санкт-Петербург, 2000. – 384 с.
33. Человеко-машинное взаимодействие: Учебное пособие / Новосибирский государственный технический университет. – Новосибирск, 2006. – 97 с.
34. Интерфейс "Користувач-комп'ютер": Навчальний посібник / В.П. Майданюк, А.М. Петух. - Вінниця: ВДТУ, 1999. - 66 с.
35. Мандел Т. Разработка пользовательского интерфейса / Т. Мандел. - Пер. с англ. – М.: ДМК Пресс, 2001. – 416 с.
36. Коутс Р. Интерфейс "Человек - компьютер" / Р. Коутс, И. Влейминк. - Пер. с англ. - М.: Мир, 1990. – 501 с.
37. Титтел Э. Создание VRML миров / Э. Титтел, К. Сандерс, Ч. Скот, П. Вольф. – Пер. с англ. – К.: Издательская группа ВНУ, 1997.
38. Ганеев Р. М. Проектирование интерфейса пользователя средствами Win32 API / Р. М. Ганеев. – М.: Горячая линия, 2002. – 340 с.
39. Microsoft Visual Studio 2008/10 Documentation.\
40. Шеховцев В. А. Операційні системи / В. А. Шеховцев. – К.: ВНУ, 2005. – 576 с.
41. Ларман К. Применение UML 2.0 и шаблонов проектирования. Введение в объектно-ориентированный анализ, проектирование и итеративную

- разработку/ Ларман К. - Москва: Вильямс, 2009.-736 с.- ISBN 978-5-8459-1185-8.
- 42.Соммервилл И. Инженерия программного обеспечения/ Соммервилл И. - Москва: И.Д. "Вильямс", 2002-624 с.-ISBN 5-8459-0330-0, 0-201-39815-X.
 - 43.Объектно-ориентированный анализ и проектирование с примерами приложений/ [Гради Буч, Роберт А. Максимчук, Майкл У. Энгл та інші] ; пер. с англ. Ключин Д.-Москва:Вильямс, 2010.-720 с. -ISBN 978-5-8459-1401-9, 0-201-89551-X.
 - 44.N. Wirth. Program Development by Stepwise Refinement/ N. Wirth // Communications of the ACM -vol.26(1).– 1983.
 - 45.Dahl O.-J. Structured Programming/ Dahl O.-J., Dijkstra E., Hoare C.A.R.– London, England: Academic Press, 1972.- 220pp.- ISBN 0-12-200550-3.
 - 46.Лингер Р. Теория и практика структурного программирования / Лингер Р., Миллс Х., Уитт Б.; пер. с англ. – М.: Мир, 1982.-406 с.
 - 47.Салливан Э. Время - деньги. Создание команды разработчиков программного обеспечения / Салливан Э.- М.:Microsoft Press, Русская редакция, 2002.-342с.- ISBN 5-7502-0189-9, 0-7356-1184-X.
 - 48.Буч. Г. UML. Руководство пользователя / Буч Г., Рамбо Дж., Джекобсон А. – СПб:ДМК-Пресс, 2004.-ISBN 978-5-8459-5015-9.
 - 49.G. Booch, J. Rumbaugh, I. Jacobson. The Unified Modeling Language Reference Manual – Second Edition, Addison-Wesley, 2004. - ISBN 5-94074-144-4.
 - 50.Олифер В. Г. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. / В.Г.Олифер, Н.А. Олифер. 3-е изд. - СПб.: Питер, 2008. - 958 с: ил.
 - 51.Закер К. Компьютерные сети. Модернизация и поиск неисправностей: / К. Закер. 2-е изд. – СПб.: БХВ-Петербург, 2006. – 1008с.: ил.
 - 52.Кулаков Ю.А. Комп'ютерні мережі. Підручник/ Ю.А. Кулаков, Н.М. Луцький; за ред. Ковтанюка –К.: Юніор, 2003.-400с., іл.
 - 53.Таненбаум Э. Компьютерные сети. / Э. Таненбаум. 4-е изд. – СПб.: Питер, 2003.-848с.:ил.
 - 54.Шапиро Д. Windows 2000 Server. Библия пользователя. / Д. Шапиро, Д. Бойс.– Діалектика, 2004 г.
 - 55.Олгтри Т. В. Модернизация и ремонт сетей. / Т. В. Олгтри – Вильямс. – Москва, 2001.
 - 56.Немет, Э. Unix. Руководство системного администратора. / Э. Немет, Г. Снайдер, С. Сибасс – Питер. – 2006.
 - 57.примерами приложений на C++, 2-е изд./Пер. с англ. – М.: "Издательство Бином", СПб.: "Невский диалект", 1999. - 560 с., ил.
 - 58.Прата, Стивен. Язык программирования C++. Лекции и упражнения, 6-е изд. : Пер. с англ. — М. : ООО "И.Д. Вильямс", 2012. - 1248 с.
 - 59.Бьерн Страуструп. Язык программирования C++. Специальное издание. Пер. англ. – М.:Изд-во "Бином", 2011 г. -1136 с.
 - 60.Брюс Эккель. Философия Java. Библиотека программиста. – СПб: Питер, 2001.-880 с.:ил.
 - 61.Айвор Хортон. Visual C++ 2008. Базовый курс. – Москва-Санкт-Петербург-Киев: «Диалектика», 2009 -1280 с.
 - 62.Ларман, Крэг. Применение UML 2.0 и шаблонов проектирования, 3-е

- издание.: Пер. с англ. – М.: ООО “Издательский дом Вильямс”, 2007.-736 с.: ил.
63. Нейгел, Кристиан, Ивьен, Билл, Глинн, Джей, Уотсон, Карли, Скиннер, Морган. C# 2008 и платформа .NET 3.5 для профессионалов.: Пер. с англ. – М.: ООО “Издательский дом Вильямс”, 2009.-1392 с., ил.
64. Хорстманн, Кей С., Корнелл, Гарри. Java 2. Библиотека профессионала, том I. Основы, 7-е изд.: Пер. с англ.- М.: Издательский дом “Вильямс”, 2007.-896 с.: ил.
65. Мельник А.О. Архітектура комп'ютера. Наукове видання / А.О. Мельник. – Луцьк: Волинська обласна друкарня, 2008. – 470 с.
66. Петух А.М. Прикладна теорія цифрових автоматів. Навчальний посібник. Ч.1 / А.М. Петух, В.В. Войтко– В.: ВДТУ, 2001 – 75с.
67. Петух А.М. Схемотехніка ЕОМ. Навчальний посібник / А.М. Петух, Д.Т. Обідник– В.: ВДТУ, 1999 – 83 с.
68. Таненбаум Э. Архитектура компьютера. 5-е изд. / Э. Таненбаум – С.Пб., Питер, 2007. – 844 с.
69. Мюллер С. Модернизация и ремонт ПК. 18-е изд. / С. Мюллер – М.: ООО «И.Д. Вильямс», 2009. – 1280 с.
70. Гук М. Аппаратные средства IBM PC. Энциклопедия. 3-е изд. / М. Гук М. - С.Пб., Питер, 2006. – 1072 с.
71. Гуржій А.М. Архітектура, принципи функціонування і керування ресурсами IBM PC: Навч. Посібник / А.М. Гуржій, С.Ф. Коряк, В.В. Самсонов, – Харків: ТОВ “Компанія СМІТ”, 2003. – 512 с.
72. Поворознюк А.И. Архитектура компьютеров. Архитектура микропроцессорного ядра и системных устройств: Учеб. Пособие. Ч.1 / А.И. Поворознюк. – Харьков: Торнадо, 2004. – 355 с.
73. Бойко В. І. Схемотехніка електронних систем: У 3 кн. Кн. 3. Мікропроцесори та мікроконтролери: Підручник / В.І. Бойко, А.М. Гуржій, В.Я. Жуйков та ін. – К.: Вища шк., 2004 – 399 с.
74. Самофалов К.Г. Цифровые ЭВМ: Теория и проектирование / К.Г. Самофалов, В.И. Корнейчук, В.П. Тарасенко.– К. Вища школа, 1989. – 424с.
75. Савельев А.Я. Прикладная теория цифровых автоматов/ А.Я. Савельев.– М.: Высшая школа, 1987. – 272с.
76. Самофалов К.Г. Прикладная теория цифровых автоматов / К.Г. Самофалов, А.М. Романкевич, В.Н.Валуйский и др.– К.: Вища школа, 1987. – 224с.
77. Бондаренко М.Ф. Комп'ютерна дискретна математика / М.Ф. Бондаренко, Н.В. Білоус, А.Г. Руткас – Харків: Компанія СМІТ, 2004 – 480с.
78. Горбатов В.А. Основы дискретной математики: Учебное пособие для студентов вузов / В.А. Горбатов. – М.: Высш.шк., 1986. – 311 с.
79. Гаврилов Г.П. Сборник задач по дискретной математике / Г.П. Гаврилов, А.А. Сапоженко. – М.: Наука, 2009. – 386 с.
80. Емеличев В.А. Лекции по теории графов / В.А. Емеличев и др. – М. – Наука, 1990. – 384с.

81. Кук Д., Бейз Г. Компьютерная математика / Д. Кук, Г. Бейз. – М.: Мир, 2005. – 360 с.
82. Донской В. И. Дискретная математика. Учебное пособие / В. И. Донской. – Симферополь: Сонат, 2000. – 360 с.
83. Гнатів Б.В. Дискретна математика. Учебний посібник / Б.В. Гнатів. Львів: В-во Нац. ун-ту «Львівська політехніка», 2003. – 90с.
84. Бардачов Ю.М. Дискретна математика / Ю.М. Бардачов та ін. – К.: «Вища школа», 2002. – 287с.
85. Оре О. Теория графов / О. Оре. – М.: Наука, 2008. – 336 с.
86. Новиков Ф. А. Дискретная математика для программистов / Ф. А. Новиков. — СПб.: Питер, 2001. – 290с.
87. Яблонский С. В. Введение в дискретную математику / С. В. Яблонский. – М.: Высш. шк., 2001. – 340с.
88. Трохимчук Р. М. Основи дискретної математики: Практикум / Р. М. Трохимчук. – К.: МАУП, 2004. – 648 с.
89. Романюк О. Н. Організація баз даних і знань : Навч. посіб. для студ. спец. "Інтелект. системи прийняття рішень" ден. та заоч. форм навчання. Ч. 1 / О. Н. Романюк, Т. О. Савчук; Вінниц. держ. техн. ун-т. - Вінниця, 2001. - 123 с.
90. Романюк О. Н. Організація баз даних і знань : Навч. посіб. для студ. вищ. навч. закл. / О. Н. Романюк, Т. О. Савчук; Вінниц. нац. техн. ун-т. - Вінниця : Універсум-Вінниця, 2003. - 216 с.
91. Гарсиа-Молина Г. Системы баз данных. Полный курс / Г. Гарсиа-Молина, Дж. Ульман, Дж. Уидом. - М.: «Вильямс», 2003. – 1088 с.
92. Коннолли Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. 3-е издание / Т. Коннолли, К. Бегг. – М.: «Вильямс», 2003. – 1440 с.

Допоміжна

1. Зюко А. Г. Элементы теории передачи информации / А. Г. Зюко. – К.: Техніка, 1969. – 300 с.
2. Балашов К. Ю. Сжатие информации: анализ методов и подходов / К. Ю. Балашов. – Минск, 2000. – 42 с (Препринт / Ин-т техн. Кибернетики НАН Беларуси; № 6)
3. Семенюк В. В. Экономное кодирование дискретной информации / В. В. Семенюк – СПб.: СПбГИТМО (ТУ), 2001. – 115 с.
4. Питерсон У. Коды, исправляющие ошибки / У. Питерсон, Э. Уэлдон: Пер. с англ.. - М.: Мир, 1976. – 589 с.
5. Секреты и ложь. Безопасность данных в цифровом мире / Б. Шнайдер. – СПб.: Питер, 2003. – 386 с.
6. ГОСТ 28147-89. Системы обработки информации. Защита криптографическая. Алгоритмы криптографического преобразования.
7. Рекомендации по преподаванию программной инженерии и информатики в университетах = Software Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering; Computing

- Curricula 2001: Computer Science: пер. с англ. - М.: ИНТУИТ.РУ «Интернет-Университет Информационных Технологий», 2007. — 462 с.
8. Головач В. Дизайн пользовательского интерфейса. Искусство мыть слона / В. Головач. - <http://usethics.ru/>, 2010. – 97 с.
 9. Деннинг В. Диалоговые системы "человек-ЭВМ". Адаптация к требованиям пользователя / В. Деннинг, Г. Эссинг, С. Маас. - Пер. с англ. - М.: Мир, 1984. - 112с.
 10. Островский А.М. Социально-философские основания гуманизации человеко-компьютерного взаимодействия: Монография / А.М. Островский. — М.: Издатель Островский А.М., 2010.- 583 с.
 11. Фельдман С. К. Системное программирование на персональном компьютере. / С. К. Фельдман. – 2-е изд. – М.: Бук_пресс, 2006.— 512 с.
 12. Каплун В. А. Захист операційних систем. Навчальний посібник / В. А. Каплун, В. П. Майданюк. – Вінниця: ВНТУ, 2007. – 185 с.
 13. Програмування мовою асемблера: Навч. посібник / В.В. Кухарчук, В.Ю. Кучерук. - Вінниця: "УНІВЕРСУМ - Вінниця", 1989. - 60 с
 14. Григорьев В.Л. Программирование однокристалльных микропроцессоров / В.Л. Григорьев. - М.: Энергоатомиздат, 1987.
 15. Сканлон Л. Персональные ЭВМ IBM PC и XT. Программирование на языке ассемблера: Пер. с англ. / Л. Сканлон.- М.: Радио и связь, 1989с.
 16. Брэдли Д. Программирование на языке ассемблера для персональной ЭВМ фирмы IBM: Пер. с англ. / Д. Брэдли. - М.: Радио и связь, 1988с.
 17. Шнайдер А. Язык ассемблера для персонального компьютера фирмы IBM: Пер. с англ. / А. Шнайдер. - М.: Мир, 1988.
 18. Нортон П. Программно-аппаратная реализация IBM PC: Пер. с англ. / П. Нортон. - М.: Радио и связь, 1991.

Інформаційні ресурси

1. Майданюк В. П. Кодування та захист інформації. Навчальний посібник. - Вінниця: ВНТУ, 2009. - 164 с. Режим доступу: <http://inomzn.vntu.edu.ua/ukr/book.php?idx=678> (дата звернення 06.03.2016). — Назва з екрана.
2. Майданюк В. П. Кодування та захист інформації. Дистанційний курс. Режим доступу: <http://www.cde.vntu.edu.ua/courseinfo.php?cid=138> (дата звернення 06.03.2016). — Назва з екрана.
3. Всё о сжатии данных, изображений и видео. Режим доступу: <http://www.compression.ru> (дата звернення 06.03.2016). — Назва з екрана.
4. Единое окно доступа к образовательным ресурсам. Режим доступу: <http://window.edu.ru> (дата звернення 06.03.2016). — Назва з екрана.
5. Национальный открытый университет. Режим доступу: <http://www.intuit.ru> (дата звернення 06.03.2016). — Назва з екрана.

6. Димитров И. Юзабилити. Кратко о фактах / И. Димитров. - <http://www.docme.ru/doc/2498/yuzabiliti.-kratko-o-faktah>. Режим доступа: <http://window.edu.ru> (дата звернення 06.03.2016). — Назва з екрана.
7. Единое окно доступа к образовательным ресурсам. Режим доступа: <http://window.edu.ru> (дата звернення 06.03.2016). — Назва з екрана.
8. Национальный открытый университет. Режим доступа: <http://www.intuit.ru> (дата звернення 06.03.2016). — Назва з екрана.
9. Майданюк В.П., Ракитянська Г.Б., Каплун В.А. Системне програмування і операційні системи. Навчальний посібник. – Вінниця, ВНТУ, 2004. Режим доступа: <http://inomzn.vntu.edu.ua/ukr/book.php?idx=76>. (дата звернення 06.03.2016). — Назва з екрана.
10. Ресурсний центр Microsoft. Портал для преподавателей. Режим доступа: <http://www.microsoft.com/Rus/Msdnaa/Curricula/default.aspx> (дата звернення 06.03.2016). — Назва з екрана.
11. Единое окно доступа к образовательным ресурсам. Режим доступа: <http://window.edu.ru> (дата звернення 06.03.2016). — Назва з екрана.