

Міністерство освіти і науки України  
Хмельницький національний університет

«Затверджую»  
Проректор з НІР  
\_\_\_\_\_ Матюх С.А.  
«\_\_» \_\_\_\_\_ 2020р.

**ПРОГРАМА**

фахового вступного випробування для навчання  
за освітньо-професійною програмою магістра  
“Інженерія програмного забезпечення”  
спеціальності 121 – Інженерія програмного забезпечення

Затверджено на засіданні кафедри інженерії програмного забезпечення  
Протокол № від \_\_\_\_\_ 2020р.

Завідувач кафедри ІІЗ  
\_\_\_\_\_ д.ф.-м.н., проф. Бедратюк Л.П.

Затверджую  
Декан ФПКТС  
\_\_\_\_\_ к.т.н., проф. Савенко О.С.

Схвалено Вченою радою ФПКТС  
Протокол №\_\_ від \_\_ \_\_\_\_\_ 2020р.

Голова Вченої ради ФПКТС  
\_\_\_\_\_ к.т.н., проф. Савенко О.С.

## **Загальні положення**

Фахове вступне випробування проводиться приймальною комісією Хмельницького національного університету – за спеціальністю 121 – «Інженерія програмного забезпечення».

Під час виконання завдання перевіряються знання, вміння та навички студентів щодо розв'язання певних завдань з дисциплін «Конструювання програмного забезпечення», «Моделювання та оцінка програмного забезпечення», «Архітектура та проектування програмного забезпечення», «Аналіз вимог до програмного забезпечення», «Бази даних», «Об'єктно-орієнтоване програмування».

## **Критерії оцінювання**

При тестуванні абітурієнт одержує завдання з 50 тестових питань – відповідно на перевірку основних знань і вмінь з дисциплін «Конструювання програмного забезпечення», «Моделювання та оцінка програмного забезпечення», «Архітектура та проектування програмного забезпечення», «Аналіз вимог до програмного забезпечення», «Бази даних», «Об'єктно-орієнтоване програмування».

Оцінюється тестове завдання за 200-бальною шкалою (від 100 до 200 балів), причому кожне тестове питання оцінюється рівною кількістю балів. Для подальшої участі в конкурсі абітурієнт повинен набрати не менше 124 балів.

## **Зміст навчального матеріалу**

У програмі наведені в довільному порядку можливі теми, за якими сформовані тестові питання.

## **Конструювання програмного забезпечення**

Вступ в конструювання програмного забезпечення. Історія розвитку. Базові терміни. Елементи конструювання ПЗ. Ключові принципи конструювання. Основи конструювання програмного забезпечення. Планування та управління процесом розроблення та супроводу програмного забезпечення. Технологія конструювання програмного забезпечення. Базові терміни. Види програмного забезпечення. Життєвий цикл процесу розробки ПЗ. Макетування. Стратегії конструювання ПЗ. Високоякісне кодування. Уніфікована мова моделювання UML. Структура мови UML. Предмети, відношення в UML. Діаграми, їх класифікація в UML. Загальні правила і механізми мови моделювання. Моделювання класів. Діаграми класів. Об'єкти і відносини класів. Реалізація програмою на псевдокодi. Діаграми

класів. Керування конструюванням. Моделювання структури. Об'єктно-орієнтовне моделювання поведінки. Кінцеві автомати. Поведінка додатків. Засоби моделювання поведінки. Моделювання взаємодії та поведінки системи. Діаграми станів. Діаграми діяльності. Діаграми взаємодії, кооперації. Вплив UML на процес розробки. Підвищення продуктивності програмування. Рівні моделювання. Основні концепції і фази проектування. Методологія і технологія розробки ПЗ. Методологія RAD - Rapid Application Development. Проектування в конструюванні. Кодування. Загальні принципи використання змінних. Проектування в конструюванні. Методика оптимізації коду. Проектування в конструюванні. Патерни (шаблони проектування). Патерн Фасад. Патерн Спостерігач. Якість програмного забезпечення.

### **Моделювання та оцінка програмного забезпечення**

Теоретичні аспекти процесу моделювання ПЗ. Поняття моделі, процесу моделювання ПЗ. Етапи процесу моделювання. Принципи структурних методів та структурного аналізу. UML – складова процесу розробки ПЗ.. Концептуальна модель UML. Види діаграм UML. Правила UML. Основи структурного моделювання ПЗ. Моделювання класів, об'єктів та екземплярів: класи, атрибути, операції, обов'язки; моделювання словника системи; моделювання розподілу обов'язків у системі; моделювання непрограмних сутностей; створення якісних абстракцій. Основні механізми доповнення й розширення будівельних блоків UML. Моделювання зв'язків і відношень.

Типові зв'язки: зв'язки: залежності, узагальнення, асоціації; моделювання простих залежностей; моделювання одиничного спадкування; моделювання структурних зв'язків; створення мереж зв'язків. Розширені зв'язки. Інтерфейси, типи та ролі. Діаграми для моделювання статичних характеристик системи. Моделі статичних представлень системи. Діаграми класів. Діаграми об'єктів. Компоненти та інтерфейси. Діаграми компонентів. Моделювання поведінки ПЗ. Моделювання варіантів використання: дійові особи, включення й розширення. Моделювання поведінки елемента. Реалізація варіантів використання за допомогою кооперацій. Діаграми варіантів використання. Моделювання контексту системи. Моделювання вимог до системи. Пряме й зворотне проектування. Моделювання взаємодій об'єктів. Стратегія моделювання взаємодій як обміну повідомленнями між об'єктами. Ролі, посилення, повідомлення, дії та послідовності. Моделювання потоків керування. Створення добре структурованих алгоритмів. Діаграми взаємодії. Моделювання потоків керування за часом. Моделювання потоків керування по організації. Діаграми діяльності. Моделювання потоку робіт. Моделювання операції.

Моделювання подій сигналів. Події сигналу, виклику, часу і зміни. Моделювання серії сигналів. Опрацювання подій в активних і пасивних об'єктах. Моделювання винятків. Кінцеві автомати. Моделювання поведінки працюючих спільно об'єктів. Стани, переходи і діяльності. Моделювання ЖЦ об'єкта. Створення добре структурованих алгоритмів. Процеси і потоки

керування. Активні об'єкти, процеси і потоки. Моделювання множини потоків керування. Моделювання міжпроцесорної комунікації. Побудова абстракцій захищених потоків. Моделювання систем реального часу. Час, тривалість і місце розташування. Моделювання тимчасових обмежень. Моделювання розподілу об'єктів. Діаграми станів. Моделювання реактивних об'єктів. Пряме і зворотне проектування. Основи моделювання архітектури ПЗ. Поняття про архітектуру та раціональний уніфікований процес. Моделювання архітектурних зразків. Зразки і каркаси. Моделювання зразків проектування. Моделювання архітектурних зразків. Забезпечення доступності зразків. Моделювання кооперації. Кооперації, реалізації й взаємодії. Моделювання реалізації варіанта використання. Моделювання реалізації операції. Моделювання механізму. Моделювання пакетів як способів організації елементів моделі. Пакети, видимість, імпорт і експорт. Моделювання груп елементів. Моделювання архітектурних виглядів. Масштабування великих систем. Паттерни проектування. Поняття про патерни (шаблони проектування). Види патернів. Патерн Фасад. Патерн Спостерігач.

### **Архітектура та проектування програмного забезпечення**

Поняття архітектури програми. Цілі вибору архітектури. Декомпозиція. Моделі, каркаси та зразки проектування. Використання моделей.. Каркаси. Класифікація архітектур. Зразки проектування. Компоненти. Типи архітектур та їх моделі. Архітектури, засновані на потоках даних. Незалежні компоненти. Віртуальні машини. Репозиторні архітектури. Рівневі архітектури. Додатки зі змішаною архітектурою. Процедура вибору архітектури. Архітектура: нотація, стандарти та інструментальні засоби. Нотація. Інструментальні засоби. Стандарт IEEE/ANSI для опису проєктів. Контроль якості при виборі архітектури. Якість та вибір архітектури. Вибір з альтернативних архітектур. Перевірка архітектури з допомогою діаграм варіантів використання. Інспектування вибору архітектури. Вплив вибору архітектури на SPMP. Вступ в детальне проектування. Поняття детального проектування. Співвідношення варіантів використання, архітектури та детального проектування. Типова схема процесу детального проектування. Проектування по схемі USDP. Проектування та інтерфейси. Повторно використовуємі елементи. Діаграми послідовності та діаграми потоків даних в детальному проектуванні. Детальні діаграми послідовності. Детальні діаграми потоків даних. Специфікація алгоритмів, класів та функцій. Інваріанти класу. Інваріанти, передумови та післяумови функцій. Блок-схеми. Псевдокод. Використання блок-схем та псевдокоду. Зразки проектування: прийоми детального проектування. Креаційні зразки проектування. Структурні зразки проектування. Зразки проектування, засновані на поведінці додатка. Бібліотеки стандартних шаблонів. Стандарти, нотація та інструментальні засоби проектування. Бібліотека стандартних шаблонів (STL) C++. Стандарт IEEE 890. Інструменти, що використовують

вихідний код: Javadoc. Вплив детального проектування на проект. Оцінка об'єму робіт з допомогою детального проектування. Якість і метрики в детальному проектуванні. Інспектування для детального проектування.

## **Аналіз вимог до програмного забезпечення**

Поняття інформаційної системи. Означення ІС. Класифікація ІС: по масштабу, архітектурі, характеру використання інформації, системі представлення даних, по стандартам управління і технологіям комунікації які підтримуються, степені автоматизації. Роль вимог в задачі впровадження АІС. Методи класифікації вимог. Означення поняття вимог. Класифікація вимог: вимога до продукту і процесу, рівні вимог, системні вимоги і вимоги до програмного забезпечення, функціональні, нефункціональні вимоги і характеристики продукту, класифікація RUP. Методології і стандарти які регламентують роботу з вимогами. Властивості вимог. Повнота, ясність коректність і узгодженість(несуперечливість), верифікація (придатність до перевірки), необхідність і корисність при експлуатації, здійснюваність (виконуваність), трасування, впорядкованість по важливості і стабільності, наявність кількісної метрики, вимоги яких не повинно бути. Процес аналізу вимог. Робочий потік аналізу вимог, для чого потрібно аналізувати вимоги, хто створює і використовує вимоги, організація роботи з вимогами на прикладі MSF Контекст задачі аналізу вимог. Аналіз вимог, бізнес-аналіз , аналіз проблемної області. Роль глосарія при АТ. Методології бізнес-аналізу. Вимоги і архітектура АІС. Аналіз вимог і інші робочі потоки програмної інженерії Виявлення вимог. Джерело вимог. Стратегії виявлення вимог. Що потрібно пам'ятати при опитуванні. Формування бачення. Введення продукту. Границі проекту. Бачення RUP. Бачення/рамки MSF. Класифікація і специфікація вимог. Актори і варіанти використання. Глосарій. Специфікація варіанту використання. Специфікація не функціональних вимог. Атрибути вимог. Розширений аналіз вимог. Моделювання. Вибір моделей. Моделі UML, які пояснюють функціональність системи. Діаграми UML, які пояснюють внутрішню будову системи. Альтернативні мови моделювання. Розширений аналіз вимог. Ілюстровані сценарії і прототипи. Цілі прототипування. Класифікація прототипування. Ілюстровані сценарії прецедентів. Середня інтенсивність використання. Документування вимог. Документування вимог в RUP. Документування вимог на основі міжнародного стандарту ISO / ІЕС / ІЕЕЕ. Вимоги до зовнішнього інтерфейсу. Інші нефункціональні вимоги. Документування вимог в MSF. Перевірка вимог. Верифікація і валідація. Деякі типові проблемні ситуації процесу формування оцінки вимог. Методи і засоби перевірки вимог. Управління вимогами. Принципи і прийоми керування вимогами. Управління змінами. Трасування вимог Удосконалення процесів роботи з вимогами Моделі вдосконалення. Принципи вдосконалення. Процес вдосконалення. Оцінка результатів і прийняття рішень. Вимоги до управління проектом.

Планування проекту на основі RUP. Вимоги в гнучких методологіях. Аналіз вимог і керування ризиками. Стратегії і роботи по управлінню ризиком. Аналіз сучасного розвитку АІС. Критерії вибору ПЗ. Процес вибору.

## Бази даних

Предмет, об'єкт, метод, мета і задачі дисципліни "Бази даних". Базові визначення і поняття. Призначення й основні компоненти системи баз даних. Існуючі вимоги до організації баз даних. Основна література по дисципліні і її короткий аналіз. Бази даних. Функції СКБД. Файлові системи. Структури файлів. Іменування файлів. Захист файлів. Режим багатокористувацького доступу. Області застосування файлів. Споживачі інформаційних систем. Основні функції СКБД. Типова організація сучасної СКБД. Ієрархічні системи. Ієрархічні структури даних. Маніпулювання даними. Обмеження цілісності. Мережеві системи. Мережні структури даних. Маніпулювання даними. Обмеження цілісності. Переваги і недоліки ранніх підходів до реалізації. Загальні поняття реляційного підходу до організації БД. Основні концепції і терміни. Базові поняття реляційних баз даних. Тип даних. Домен. Схема відношень, схема бази даних, кортеж, відношення. Фундаментальні властивості відношень. Реляційна модель даних. Загальна характеристика. Цілісність сутності і посилань. Базові поняття реляційної моделі даних. Множини. Операції над множинами. Декартовий добуток множин. Відношення. Приклади відношень. Бінарні відношень (відношення степеня 2). Відношення еквівалентності. Відношення порядку. Функціональне відношення, n-арне відношення (відношення степеня n). Транзитивне замикання відношень. Загальна характеристика реляційної моделі даних. Типи даних. Прості типи даних. Структуровані типи даних. Посилальні типи даних. Типи даних, використовувані в реляційній моделі. Домени. Відношення, атрибути, кортежі відношення. Визначення і приклади. Властивості відношень. Перша нормальна форма. Реляційна алгебра. Огляд реляційної алгебри. Замкнутість реляційної алгебри. Відношення, сумісні по типі. Оператор перейменування трибутів. Теоретико-множинні оператори. Об'єднання. Перетинання. Вирахування. Декартовий добуток. Спеціальні реляційні оператори. Вибірка (обмеження, селекція). Проекція. З'єднання. Загальна операція з'єднання. Тета-з'єднання. Екві-з'єднання. Природне з'єднання. Розподіл. Приклади використання реляційних операторів. Залежні реляційні оператори. Оператор з'єднання. Оператор перетинання. Оператор розподілу. Примітивні реляційні оператори. Оператор декартового добутку. Оператор проекції. Оператор вибірки. Оператори об'єднання і вирахування. Запити, невимовні засобами реляційної алгебри. Погана нормалізація відношень. Невимовність транзитивного замикання реляційними операторами. Крос-таблиці. Мови запитів до баз даних. Мова SQL. Оператори SQL. Оператори DDL (Data Definition Language) оператори визначення об'єктів бази даних. Оператори DML (Data Manipulation

Language) оператори маніпулювання даними. Оператори захисту і керування даними. Приклади використання операторів маніпулювання даними. INSERT вставка рядків у таблицю. UPDATE - відновлення рядків у таблиці. DELETE видалення рядків у таблиці. Приклади використання оператора SELECT. Вибір даних з однієї таблиці. Вибір даних з декількох таблиць. Використання імен кореляції (альясів, псевдонімів). Використання агрегатних функцій у запитах. Використання агрегатних функцій з угрупованнями. Використання підзапитів. Використання об'єднання, перетинання і різниці. Синтаксис оператора вибірки даних (SELECT)BNF-нотація. Синтаксис оператора вибірки. Синтаксис з'єднаних таблиць Синтаксис умовних виражень роздязнула WHERE. Порядок виконання оператора SELECT. Реалізація реляційної алгебри засобами оператораSELECT (Реляційна повнота SQL. Оператор декартового добутку. Оператор проєкції. Оператор вибірки. Оператор об'єднання. Оператор вирахування. Оператор з'єднання. Оператор перетинання. Оператор розподілу.

СКБД MySQL Загальна характеристика СКБД MySQL. Об'єкти БД: таблиці, форми, запити, звіти. Структура СКБД MySQL. Створення БД.Створення таблиць БД. Типи та властивості полів БД. Заповнення, редагування таблиць. Ключові поля. Відношення між таблицями. Створення схеми даних. СКБД Access (Open Office.org Base). Загальна характеристика СКБД MsAccess (Open Office.org Base).

Нормальні форми відношень. Етапи розробки бази даних. Критерії оцінки якості логічної моделі даних. Адекватність бази даних предметній області. Легкість розробки і супроводження бази даних. Швидкість операцій відновлення даних (вставка, відновлення, видалення). Швидкість операцій вибірки даних. 1НФ (Перша Нормальна Форма). Аномалії оновлення. Аномалії вставки (INSERT). Аномалії оновлення (UPDATE). Аномалії видалення (DELETE). Функціональні залежності. Визначення функціональної залежності. Функціональні залежності відношень і математичне поняття функціональної залежності. 2НФ (Друга Нормальна Форма). Аналіз декомпованих відношень (аномалії вставки, оновлення та аномалії видалення, що Залишилися для 2НФ), 3НФ (Третя Нормальна Форма) Алгоритм нормалізації (приведення до 3НФ). Аналіз критеріїв для нормалізованих і ненормалізованих моделей даних Порівняння нормалізованих і ненормалізованих моделей. OLTP і OLAP-системи Коректність процедури нормалізації декомпозиція без утрат. Теорема Хеза. Нормальні форми більш високих порядків. НФБК (Нормальна Форма Бойса-Кодда). 4НФ (Четверта Нормальна Форма). 5НФ (П'ята Нормальна Форма). Алгоритм нормалізації для 5НФ. Моделі даних. Основні поняття ER-діаграм. Приклад розробки простої ER-моделі. Концептуальні і фізичні ER-моделі Транзакції і цілісність баз даних. Приклад порушення цілісності бази. Поняття транзакції. Обмеження цілісності. Класифікація обмежень цілісності. Класифікація обмежень цілісності за способами реалізації. Класифікація обмежень цілісності за часом перевірки. Класифікація обмежень цілісності по області дії. Обмеження домену. Обмеження атрибута.

Обмеження кортежу. Обмеження відношення. Обмеження бази даних. Реалізація декларативних обмежень цілісності засобами SQL. Загальні принципи реалізації обмежень засобами SQL. Синтаксис обмежень стандарту SQL. Синтаксис операторів SQL, що використовують обмеження. Транзакції і паралелізм. Робота транзакцій у суміші. Проблеми паралельної роботи транзакцій. Проблема втрати результатів відновлення. Проблема незафіксованої залежності (читання «брудних» даних, неакуратне зчитування). Проблема несумісного аналізу. Неповторювальне зчитування. Фіктивні елементи (фантоми). Несумісний аналіз. Конфлікти між транзакціями. Блокування. Рішення проблем паралелізму за допомогою блокувань. Види відновлення даних. Індивідуальний відкат транзакції. Відновлення після м'якого збою. Відновлення після твердого збою.

СКБД в архітектурі "клієнт-сервер". Архітектура "клієнт-сервер". Відкриті системи. Клієнти і сервери локальних мереж. Системна архітектура "клієнт-сервер". Сервери баз даних. Взаємодія різних типів СКБД. Технології доступу та обміну даних між різними типами СКБД. Розробка мережевих баз даних. Проблеми розробки та способи проектування «клієнт-серверних» додатків база даних. Розподілені бази даних. Різновиди розподілених систем БД. Іменування об'єктів і організація розподіленого каталогу. Розподілена компіляція запитів. Керування транзакціями і синхронізація. Інтегровані чи федеративні системи і мультибази даних. Системи керування базами даних наступного покоління. Орієнтація на розширену реляційну модель. Абстрактні типи даних. Генерація систем баз даних, орієнтованих на додатки. Оптимізація запитів, керована правилами. Підтримка історичної інформації і темпоральних запитів. Часові бази даних. Проблеми часу в базах даних. Зв'язок часових баз даних СКБД із загальними поняттями об'єктно-орієнтованого підходу та реляційними моделями. Приклади мов програмування ООБД. Мови запитів часових баз даних.

## **Об'єктно-орієнтоване програмування**

Технології об'єктно-орієнтованого проектування програмних систем. Парадигми програмування. Об'єктно-орієнтований підхід до програмування. Фундаментальні поняття ООП. Критика ООП. Сучасні технології та платформи проектування програмних систем. Особливості створення додатків в .NetFramework та особливості мови С#. Базові поняття .NetFramework. Створення додатків з допомогою .NetFramework. Класи та об'єкти С#. Основні поняття. Оголошення та структура класу. Дані та методи класу. Декларації private, protected, public. Ініціалізація об'єктів класу. Конструктори та деструктори, їх призначення, оголошення, та варіанти викликів. Звичайні, констатні та статичні дані та методи, особливості їхнього оголошення. Одержання детальної інформації про клас.

Абстрактні класи. Розгорнуті й вказівкові типи. Класи й структури. Структури. Вбудовані структури. Перерахування. Особливості перерахувань. Концепція наслідування. Відношення між класами. Відношення «є» і «має».



Успадкування. Механізм успадкування. Позитивні аспекти успадкування. Позитивні аспекти успадкування і їх практичні можливості. Реалізація успадкування в C#. Додавання полів нащадком. Конструктори батьків і нащадків. Додавання методів і зміна методів батька. Статичний контроль типів і динамічне зв'язування. Інтерфейси. Означення Інтерфейсу. Два способи реалізації інтерфейсу. Перетворення до класу інтерфейсу. Вбудовані інтерфейси.

Множинне наслідування. Проблеми множинного спадкування. Колізія імен. Успадкування від загального батьківського класу. Делегати. Означення делегата. Функції вищих порядків. Операції над делегатами. Клас Delegate. Основні методи й властивості класу Delegate. Приклад використання делегатів. Події. Події для багатоадресної передачі. Порівняння методів екземплярів зі статичними методами, що використовуються в якості обробника подій. Використання подійних засобів доступу. Рекомендації по обробці подій в середовищі .NET Framework. Використання вбудованого делегата EventHandler. Використання подій. Приклади використання. Універсальність. Класи з родовими параметрами. Спадкування і універсальність. Синтаксис універсального класу. Клас з універсальними методами. Два основних механізми об'єктної технології. Стек. Від абстрактного, універсального класу до конкретних версіями. Обмежена універсальність. Синтаксис обмежень. Список з можливістю пошуку елементів за ключем. Родове породження класу. Універсальність і спеціальні випадки класів. Універсальні структури. Універсальні інтерфейси. Універсальні делегати. .Net Framework і універсальність. Коректність і стійкість програмних систем. Життєвий цикл програмної системи. Три закони програмотехніки. Перший закон програмотехніки (закон для розробника). Другий закон програмотехніки (закон для користувача). Третій закон програмотехніки (закон того, хто вичікує). Налаштування. Створення надійного коду. Мистецтво налаштування. Налаштування і умовна компіляція. Класи Debug. Метод Флойда та затвердження тверджень. Класи StackTrace і BooleanSwitch. Налаштування та інструментальне середовище Visual Studio. Обробка виняткових ситуацій. Обробка виключень у мовах C/C++. Схема обробки виключень в C#. Усунення винятків. Створення об'єктів винятків. Захоплення винятку. Паралельна робота обробників виключень. Блокування. Схема Бертрана обробки виняткових ситуацій. Клас винятків.

## Перелік джерел посилання

## Перелік джерел посилання

1. Савенко О.С., Кльоц Ю.П., Лисенко С.М. Системне програмне забезпечення. Хмельницький: ХНУ, 2016. 403с.
2. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений: Пер. с англ. - М.: Вильямс, 2007. 1410 с.
3. Сорокати́й Р.В., Пасі́чник О.А. Основи об'єктно-орієнтованого програмування мовою C#. навч. посіб. Хмельницький: ХНУ, 2013. 193 с.
4. Бородкіна І.Л. Бородкіна Г.О. Інженерія програмного забезпечення: посібник для студентів вищих навчальних закладів: навч. посіб. Київ: Центр навчальної літератури (ЦНЛ), 2019. 204 с.
5. Браунде Э. Технология разработки программного обеспечения. Санкт-Петербург : Питер., 2011. – 655 с.
6. Орлов С.А. Технологии разработки программного обеспечения. (Разработка сложных программных систем):учеб. пособие. Санкт-Петербург : Питер. 2012. 464с.
7. Карпенко М. Ю., Манакова Н. О., Гавриленко І. О. Технології створення програмних продуктів та інформаційних систем: навч. посібник. Харків: Нац. ун-т міськ. госп-ва ім. О. М. Бекетова, 2017. 93 с.
8. Мацяшек Л.А. Анализ требований и проектирование систем. Разработка информационных систем с использованием UML. пер. с англ. Москва: Вильямс. 2012. 432с.
9. Томашевський В.М. Моделювання систем: Підручник. Київ:Видавнича група ВНУ. 2015. 352с.
10. Алан Купер. Интерфейс. Основы проектирования взаимодействия. 4-е изд.. Санкт-Петербург: Питер. 2020. 720с.
11. Мінухін С.В., Беседовський О.М., Знахур С.В. Методи і моделі проектування на основі сучасних CASE-засобів: навч. посіб. Харків: ХНЕУ. 2010.-272с.
12. Форкун Ю.В. Методичні вказівки до виконання курсового проекту для студентів напряму підготовки “Програмна інженерія” з дисципліни «Архітектура та проектування програмного забезпечення. Хмельницький: ХНУ, 2015. – 46 с.
13. ДСТУ ISO/IEC 2382:2017 (ISO/IEC 2382:2015, IDT). Інформаційні технології. Словник термінів. Частина 15. Мови програмування (ISO/IEC 2382:2015, IDT). Вид. офіц. Київ: Держстандарт України, 2019.
14. Інформатика: Комп'ютерна техніка. Комп'ютерні технології. Посіб. За ред. О.І.Пушкаря. Київ: Видавничий центр “Академія”, 2012. 696с.
15. Макконнелл С. Совершенный код. Мастер-класс. пер. с англ. Москва: Издательско-торговый дом "Русская Редакция", 2011. 896.
16. Дудзяний І.М. Об'єктно-орієнтоване моделювання програмних систем: Навчальний посібник. Львів: Видавничий центр ЛНУ імені Івана Франка, 2010. 108 с.

17. Ларман, Крэг. Применение UML и шаблонов проектирования. 2-е издание.: пер. с англ. Москва: Вильямс, 2011. 624 с.
18. Мещанінов О. П. Моделювання систем: Навчальний посібник. - Миколаїв: Вид-во МФ НаУКМА, 2011. 268с.
19. Мінухін С.В. Case-технології [Текст]: Конспект лекцій. Харків:ХНЕУ.- 2015. 156с.
20. Орлов С. Технологии разработки программного обеспечения: Учебник. Санкт-Петербург, Питер, 2012. 464 с.
21. Гибкая методология разработки программного обеспечения. Санкт-Петербург: Русская редакция, 2011. 127 с
22. Резник С., Бьерк А., де ла Маза М. Scrum с Team Foundation Server 2010. Профессиональный подход. Москва: ЭКОМ Паблишерз, 2012.
23. Майкл С.В. Тернер Основы Microsoft Solutions Framework. Санкт-Петербург: Русская редакция, 2013. 336 с.
24. Федоров А. Знакомство с Microsoft Visual Studio Team System 2008. Санкт-Петербург: Русская редакция, 2010. 34 с.
25. Бази даних. Методичні вказівки до курсового проектування для студентів напряму підготовки “Програмна інженерія”. Хмельницький : ХНУ, 2014. – 28 с.
26. Кузнецов С. Основы баз данных : учебное пособие. Москва: Интернет-Университет Информационных Технологий, 2011. 484 с.
27. Гольцман В. MySQL 6.0. Санкт-Петербург: Питер, 2015. 255 с.
28. Маран М.М. Программная инженерия. Санкт-Петербург: Лань, 2016. 196 с.
29. Ендрю Троелсен, Філіпп Язык программирования C# 7 и платформы .NET и .NET Core. Київ: Діалектика, 2020. 656 с.
30. Томас Кормен, Чарльз Лейзерсон, Рональд Рівест, Кліффорд Штайн. Алгоритмы: построение и анализ. Київ: Діалектика, 2020. 716 с.

## Зразок базового тестового завдання

1. Головним компонентом конструювання ПЗ є:
  - a. кодування та відлагодження;
  - b. створення методики кодування.
  - c. архітектура ПЗ;
  - d. аналіз вимог
  - e. інша відповідь
  
2. Призначення конструктора як функції-члена класу полягає в тому, що ...
  - a. Конструктор копіює об'єкти різних класів
  - b. Конструктор викликається автоматично під час оголошення класу
  - c. Конструктор повертає значення даних-членів класу з методів, що змінюють значення цих даних-членів
  - d. Конструктор створює об'єкт класу
  - e. Інша відповідь
  
3. Об'єктно-орієнтоване програмування — це
  - a. Методологія програмування, яка базується на поданні програми у вигляді сукупності процедур
  - b. Інша відповідь
  - c. Методологія програмування, яка базується на поданні програми у вигляді сукупності модулів
  - d. Методологія програмування, яка базується на поданні програми у вигляді сукупності класів, які утворюють ієрархію, на принципах спадкування
  - e. Методологія програмування, яка базується на поданні програми у вигляді сукупності процедур і функцій
  
4. Абстракція – це:
  - a. Інша відповідь
  - b. Характеристика змінних, які використовуються в процедурах
  - c. Різновид процедур в програмі
  - d. Різновид процедур та функцій
  - e. Характеристика змінних, які використовуються в функціях
  
5. Модульність – це:
  - a. Розбиття програми на функції
  - b. Розбиття програми на процедури та функції
  - c. Розбиття програми на окремі класи
  - d. Розбиття програми на процедури
  - e. Інша відповідь
  
6. Наслідування – це:
  - a. Відношення між модулями, коли один модуль може використовувати можливості іншого

- b. Відношення між змінними, коли значення однієї змінної також є значенням іншої с. Інша відповідь
- d. Відношення між функціями, коли функціональні можливості однієї функції може використовувати інша
- e. Відношення між ділянками оперативної пам'яті, коли одну ділянку пам'яті можуть використовувати різні змінні

7. Типізація – це:

- a. Інша відповідь
- b. Характеристика змінних, які використовуються в модулях
- c. Різновид абстракції, який дозволяє використовувати змішані класи
- d. Обмеження, яке накладається на функцію і перешкоджає змішуванню різних функцій
- e. Обмеження, яке накладається на модуль і перешкоджає змішуванню різних модулів

8. Операція створення і (або) ініціалізації об'єкта — це:

- a. Деструктор
- b. Інша відповідь
- c. Ітератор
- d. Модифікатор
- e. Селектор

9. Операція знищення об'єкта — це:

- a. Селектор
- b. Конструктор
- c. Ітератор
- d. Модифікатор
- e. Інша відповідь

10. Операція, яка надає доступ для визначення стану об'єкта без його змін— це:

- a. Ітератор
- b. Конструктор
- c. Інша відповідь
- d. Модифікатор
- e. Деструктор

11. Виокремлення інформаційних об'єктів предметної області (таблиць), які підлягають зберіганню в БД, а також визначення характеристик об'єктів і зв'язків між ними відбувається на етапі:

- a. Моделювання предметної області
- b. Інфологічного моделювання
- c. Концептуального моделювання
- d. Даталогічного моделювання

е. Інша відповідь

12. Вираз DELETE FROM КАСА означає:

- a. Видалення окремого рядка
- b. Видалення атрибутів таблиці
- c. Видалення усіх рядків таблиці
- d. Видалення таблиці
- e. Інша відповідь

13. Вираз SELECT Назва AS Назва\_Організації... означає:

- a. Імені стовпця для вибору даних
- b. Перевизначення імені таблиці
- c. Перевизначення імені стовпця
- d. Перевизначення імені таблиці
- e. Інша відповідь

14.Віртуальні таблиці зберігають:

- a. Текст запиту SELECT, і дані
- b. Тільки дані результату виконання запиту SELECT
- c. Тільки текст запиту SELECT
- d. Тільки індекси і хеш-таблиці
- e. Інша відповідь

15.Вкладеність тригерів є допустимою?

- a. Так
- b. Ні
- c. Тригери – це термін до якого не допустимо визначення вкладеності
- d. Тільки в скбд ms access
- e. Інша відповідь

16. Властивості унікальності та ненадмірності характерні для:

- a. Альтернативних ключів
- b. Первинних ключів
- c. Потенційних ключів
- d. Простих ключів
- e. Інша відповідь

17.Внесення змін та розвиток БД відносяться до наступного етапу життєвого циклу БД:

- a. Функціонування
- b. Супроводження
- c. Тестування та оцінки
- d. Інкапсуляції
- e. Інша відповідь

18. В процедурі оголошеній змінній задати значення можна командою:

- a. DECLARE
- b. SETUP
- c. SELECT d. USE
- e. інша відповідь

19. Дані поля n\_z таблиці bd1 потрібно скопіювати в таблицю bd2:

- a. insert into bd2 n\_z select n\_z from bd1
- b. insert bd2 n\_z select n\_z from bd1 where bd2.n\_z
- c. insert into bd2 n\_z from bd1.n\_z
- d. select bd1.n\_z as bd2.n\_z
- e. інша відповідь

20. До яких прийомів відноситься затвердження вимог ПЗ

- a. Прийоми отримання основних вимог
- b. Прийоми, що визначають повний набір вимог клієнтів
- c. Прийоми, що визначають повний набір вимог для продукту
- d. Прийоми отримання вторинних вимог
- e. Інша відповідь

21. Що із перерахованого відповідає за забезпечення логіки обробки даних в інформаційній системі?

- a. Програмні засоби
- b. База даних реляційного типу
- c. Валідація
- d. Інтерфейс користувача
- e. Інша відповідь

22. З якою метою використовуються вимоги програмістом?

- a. Розробка програмного коду
- b. Складання тестових сценарії
- c. Проектування підсистем
- d. Розробка вимог
- e. Інша відповідь

23. Через що реалізується варіант використання

- a. Через функцію системи
- b. Через реєстр вимог
- c. Через аналіз
- d. Через актора
- e. Інша відповідь

24. Який із аналізів вивчає взаємодію автоматизованої інформаційної системи і її середовища?

- a. Аналіз середовища
- b. Архітектурний аналіз
- c. Аналіз проблемної області
- d. Аналіз вимог
- e. Інша відповідь

25. Після етапу формування вимог до системи виконується етап ...

- a. Розробка концепції об'єкту
- b. Розробка концепції системи
- c. Вивчення об'єкту
- d. Оформлення звіту про виконання робіт
- e. Інша відповідь