**Вступ. Предмети і завданнякурсу, цілі та принципи КПЗ. Основи КПЗ**

**КПЗ (конструювання ПЗ) — це** процес складання програмного коду та супутніх процесів даної галузі знань (проектування, моделювання, тестування, інтеграція), що використовує різноманітні методи та засоби з метою автоматизації певної предметної області, та досягнення кращих результатів та ефективності конструювання

Метою КПЗ є отримання економічного, ергономічного, архітектурного, якісного і т. д. ефекту

В 60-70их роках 20ст. Почали появлятись відповідні технології КПЗ діаметральними з яких були відповідно локальний і системни підхід до конструювання.

Локальний підхід часто описується водоспадною моделлю. Такий підхід описується класичним життєвим циклом або відповідно каскадним процесом розробки.

Самий же ж локальний підхід до КПЗ полягає у послідовному нарощуванні і вирішені у свідомому нарощуванні і вирішені задач конструювання, ці задачі є послідовними, відносно автономними (незалежними один від одного), а тому потребують на підготовчому процесі конструювання якісної декомпозиції задач.

Цей підхід є дещо ідеалізованим, оскільки багато особливо складніших задач або задачі, будуть виконуватись командою розробників досить складно, а то й і неможливо на початковому етапі розробки, спланувати й чітко визначити задачі конструювання, у чіткій послідовності

Враховуючи вище сказане, локальний підхід більше підходить до добреструктурованих задач, не складних, невеликих проектів, що мають виконуватись невеликою кількістю виконавців, а ще краще 1-2

Протилежним до локального є **системний підхід,** що полягає в грунтованому аналізі об’єкта і попередньому його вивченні конструювання, вивчення задач, підзадач та особливостей, і на основі подальшого конструюванні з узгодженням переглядом частковим як результатів, так і задач на кожному етапі

Системний підхід **характеризується наступним ознаками:**

* Одночасне охоплення КПЗ невеликої кількості задач об’єкта КПЗ
* Максимальна типізація та стандартизація проектування
* Багато аспектне подання структури ПЗ як системи, що складається з багатьох компонентів та відносної автономності їх розробки на кожному етапі
* Ключова роль центральних масивів інформації
* Локальне впровадження і накопичення функціональних задач

**Висновок**: на практиці більшість сучасних підходів та методологій використовують локальний і системні підходи, у більшій чи меншій мірі використовуючи один із них. На практиці ж усі ці методи реалізуються за допомогою методологій і засобів.

**2. Декомпозиція ПЗ. Поняття оптимально структурованої системи**

Декомпозиція це один з методів КПЗ, що полягає у розчленуванні об’єкта конструювання, або задачі на окремі частини, з яких в міру необхідності в подальшому комплектують автоматизовану систему. Декомпозиція відноситься до технологічних так званих методів КПЗ. Вона буває відповідно до суті:

* функціональною (декомпозиція виконується на основі функціональних блоків або функціональних процесів),
* організаційною (елементами є організаційні підрозділи, ієрархія),
* інформаційною
* технічна
* програмна

Сукупність інформації, яка характеризує систему і сукупність зв’язків між її елементами неможливо сприйняти вцілому і повністю, тому власне і використовують декомпозицію для спрощення, але крім того, для оперативного впровадження ПЗ згідно декомпозицій слід забезпечити оптимальну її структурованість

Оптимально структурована система є багаторівневою, багатоцілевою, організованою сукупністю елементів (модулів, сутностей) і має задовільняти такі вимоги:

1. Кожен рівень ієрархії має проглядатись і бути зрозумілим без детального знання нищих рівнів
2. Зв’язки на одному рівні ієрархії мають бути мінімальними
3. Не повинно бути зв’язків між елементами через один рівень ієрархії
4. Елемент вищого рівня ієрархії має викликати елемент наступного рівня і передаючи йому необхідну інформацію має утворити з ним єдине ціле
5. Елемент наступного рівня після закінчення своєї роботи повертає керування елементу, що його викликав разом з результатами своєї роботи

Основи КПЗ

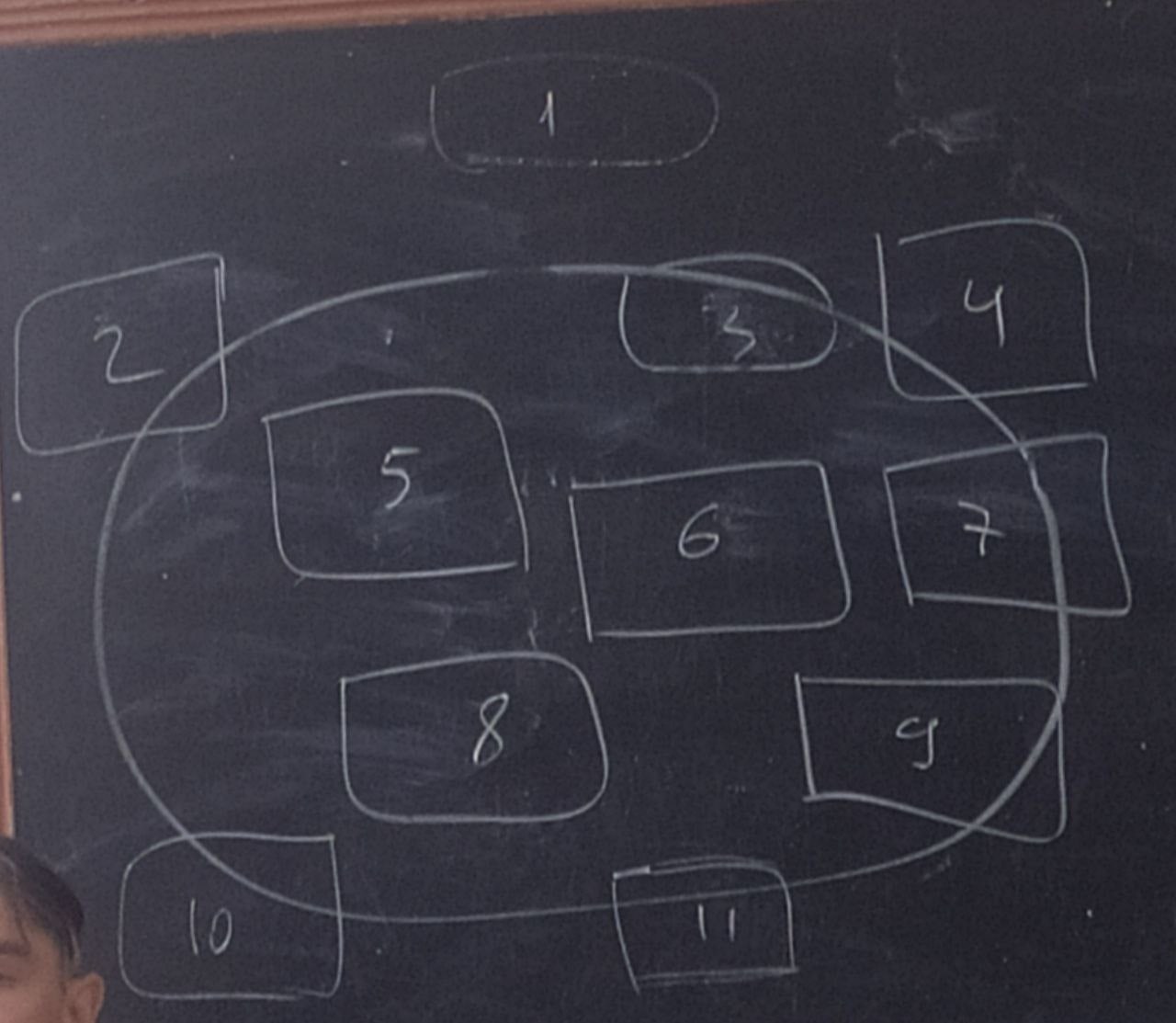
Місце КПЗ в життєвому циклі програмної системи

Розробка ПЗ це складний процес, в який входить багато складових, зокрема:

1. Визначення проблем
2. Вироблення вимог
3. Створення плану конструювання
4. Розробка архітектури ПЗ, або високорівневе проектування
5. Детальне проектування
6. Кодування і відлагодження
7. Блочне тестування
8. Інтеграційне тестування
9. Інтеграція
10. Тестування систем
11. Коригувальне супроводження

Сам же термін КПЗ описує детальне створення робочої програмної системи за допомогою комбінації кодування, верифікації, модульного та інтеграційного тестування і відлагодження

На рисунку 1 показано місце конструювання як частину кроків серед процесів, що проходять при побудові ПЗ



1. Визначення проблем
2. Розробка вимог
3. Детальне проектування
4. Корективна підтримка
5. Планування конструювання
6. Кодування і відлагодження
7. Інтеграція
8. Unit-тести
9. Інтеграційне тестування
10. Архітектура ПЗ
11. Системне тестування

Конструювання серед процесів побудови ПЗ

Процеси конструювання зображені всередині еліпса, відповідно головними компонентами КПЗ є:

1. Кодування
2. Відлагодження

Однак воно включає і детальне проектування, блочне тестування (unit test), інтеграційне тестування, планування

Інода конструювання називають «кодування» або «програмуванням». Кодування в даному випадку, не є точним виразом, оскільки вона має на увазі механічну трансляцію розробленого плану (моделі, алгоритму) в команди мови програмування, тоді як конструювання є зовсім не механічним процесом і часто пов’язано з творчістю та аналізом. Сенс термінів коструювання і програмування досить близькими і точними.

Дана область знань пов’язана із іншими областями ПІ, найсильніше з проектуванням і тестуванням. Причиною того є те, що сам процес КПЗ зачіпає важливі аспекти діяльності проектування і тестування, воно(конструювання) відштовхується від результатів проектування, а тестування передбачає роботу з результатами

Досить складно визначити межу між проектування, конструюванням і тестуванням оскільки вони пов’язані в єдиний комплекс процесів життєвого циклу і в залежності від обраної моделі ЖЦ і обраної методології таке поєднання і розподілення може мати різний вигляд. Хоча ряд операцій з проектування, детального дизайну, може відбуватись до стадії конструювання. Проте великий обсяг даних робіт виконується паралельно з коструюванням, або як його частин. Це є сутність зв’язку з областю знань Software Divide

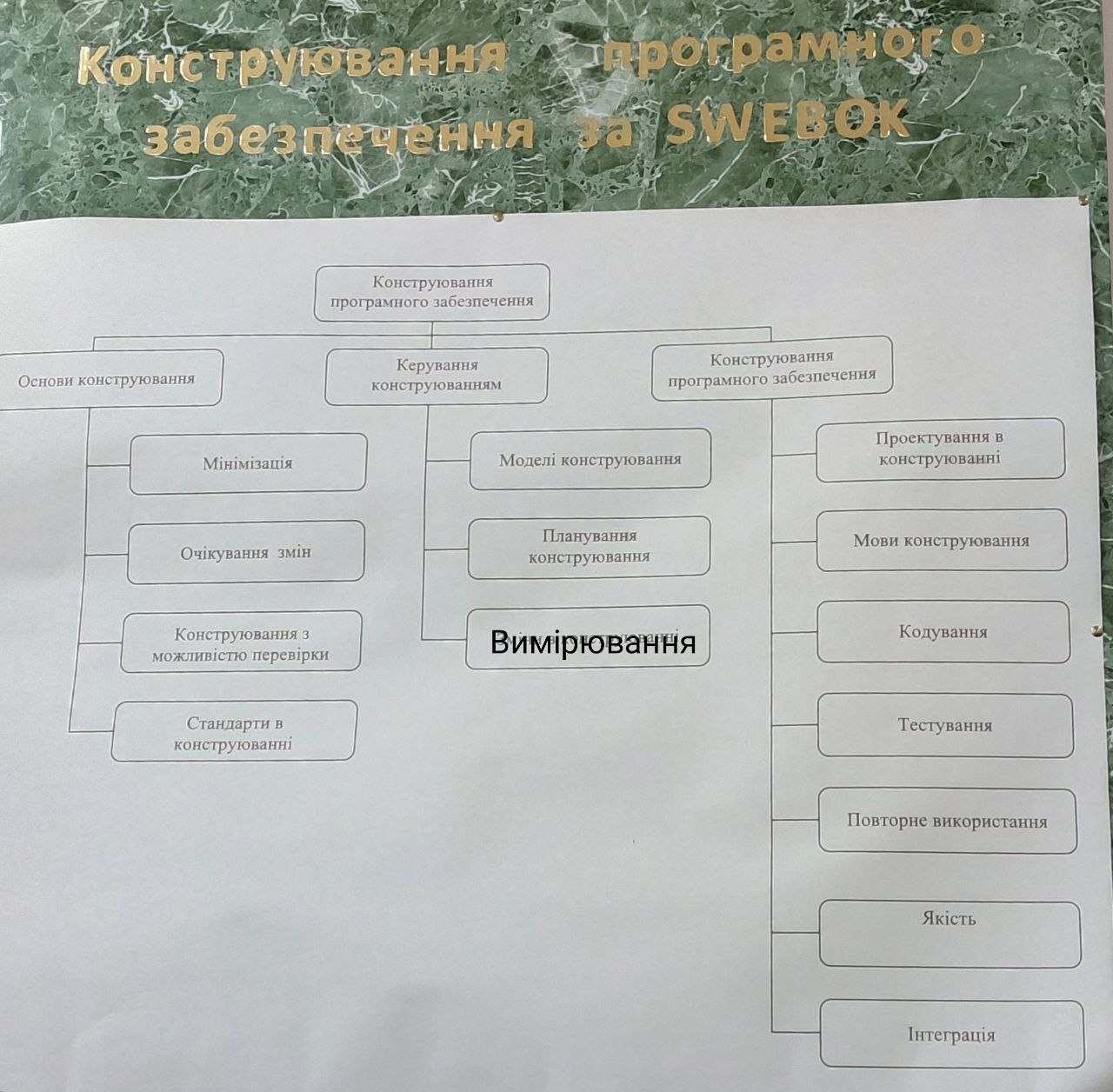
В свою чергу, протягом всієї діяльності з КПЗ інженери використовують модульне та інтеграційне тестування таким чином дана галузь знань пов’язана з Software Testing. У процесі КПЗ звичайно створується більша частина активів програмног проекту (конфігураційних елементів), тому в реальних проектах просто неможливо розглядати діяльність по конструюванню у відриві від галузі знань «Конфігураційне управління»

Оскільки конструювання неможливе без використання відповідного інструментарію і, ймовірно, дана діяльність є найбільш інструментально насиченою у конструювані важливу роль грає область знань «Інструменти і методи ПІ»

Безумовно питання забезпечення якості значимі для усіх галузей знань та ЖЦ. У той же час код є основним результуючим активом або елементом програмного проекту. Тому очевидно напрошується і присутній зв’язок КПЗ з областю знань якості ПЗ.

З пов’язаних дисциплін ПІ найбільш тісний і природний зв’язок даної галузі знань присутній з комп’ютерними науками, оскільки саме в них розглядається питання побудови й використання алгоритмів і практик кодування

Нареші КПЗ стосується і керування проектами (Project Managment), при чому в тій мірі наскільки діяльність з керування КПЗ важлива для досягнення результатів конструювання

Отже область знань конструювання графічно може бути зображена у вигляді наступної схеми  


Область знань конструювання ПЗ

Нижче приведено деякі конкретні задачі, що виникають в процесі розробки ПЗ:

1. Перевірка виконання умов необхідних для успішного конструювання
2. Визначення способів подальшого тестування коду
3. Проектування та написання класів та методів
4. Створення та присвоєння імен змінних та іменованих констант
5. Вибір керуючих структур та організація блоків команд
6. Блочне інтеграційне тестування, відлагодження власного коду
7. Взаємний огляд коду та низькорівневих програмних структур членами команди
8. Шліфування коду шляхом його ретельно форматування та коментування
9. Інтеграція програмних компонентів створених окремо
10. Оптимізація коду направлена для підвищення його швидкодії і зниження міри використання ресурсів, рефакторінг

З іншого боку, з видів діяльності, що проходять в процесі розробки ПЗ, до конструювання не відносяться:

1. Керування процесом розробки
2. Вироблення вимог
3. Розробка високорівневої архітектури програми
4. Проектування інтерфейсу користувача
5. Тестування системи і її супроводження

Для кожного з цих пунктів є своя наука