**Лекція 19. Технології конструювання програмного забезпечення**

**На період** карантину в дистанційній формі навчання на надані в кінці запитання потрібно надати письмові відповіді, виклавши на платформу коледжу (<https://github.com/omc-college/ipz41-software_design>) або надіславши їх на електронну адресу викладача. Файл надавати з іменем у форматі:

**KPZ<Номер групи><Номер лекції / лабораторної>[-<Номер завдання>][літера позначення типу роботи L – лекція, R – лабораторна]<Прізвищеанглійською>**.

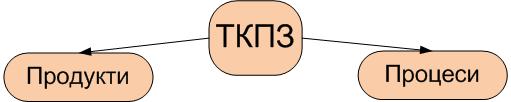
Наприклад, **KPZ4118L**buts.doc.

Відповіді повинні бути не довгими і змістовними. Не копіюйте фрагментів з різних інформаційних джерел, подумайте і викладіть свою точку зору. При наявності відповідей-"близнюків" відповідь буде зараховуватися першому за часом надсилання.

**Строк виконання цієї роботи 20.04.2020**

**Основні визначення.**

Технологія конструювання програмного забезпечення (ТКПЗ) - система інженерних принципів для створення ПЗ, яке надійно і ефективно працює в реальних комп'ютерах. ТКПЗ охоплює питання і створюваних програмних продуктів, і процесів їх створення.



***Призначення ТКПЗ*** полягає у вирішенні проблеми створення якісних комп’ютерних застосунків. Найбільш поширеними з цих проблем є такі:

* ПЗ не використовує потенційних можливостей апаратури;
* вміння будувати нові програми відстає від вимог до програм;
* низка якість розробки програм.

Ключ до рішення цих проблем – це:

* грамотна організація процесу створення ПЗ;
* реалізація технологічних принципів промислового конструювання програмних засобів.

Розрізняють методи, засоби і процедури ТКПЗ.

***Методи*** забезпечують вирішення наступних завдань:

- Планування і оцінювання проекту;

- Аналіз системних і програмних вимог;

- Проектування алгоритмів, структур даних і програмних структур;

- Кодування;

- Тестування;

- Супровід.

***Засоби (утиліти)*** ТКПЗ забезпечують автоматизовану або автоматичну підтримку методів. З метою спільного застосування утиліти можуть об'єднуватися в системи автоматизованого конструювання ПЗ. Такі системи прийнято називати CASE-системами (CASE–система–Computer Aided Software Engineering програмна інженерія з комп’ютерною підтримкою.).

***Процедури*** є зв’язковою ланкою, яка з'єднує методи і утиліти так, що вони забезпечують безперервний технологічний ланцюжок розробки. Процедури визначають:

- Порядок застосування методів і утиліт;

- Формування звітів, форм за відповідними вимогам;

- Контроль, який допомагає забезпечувати якість і координувати зміни;

- Формування контрольних точок, засад та характеристик, за якими керівники оцінюють прогрес.

Процес конструювання програмного забезпечення складається з послідовності кроків, що використовують методи, утиліти і процедури. Ці послідовності кроків часто називають парадигмами ТКПЗ.

Таким чином, процес конструювання ПЗ складається з послідовності кроків, які використовують:

* методи;
* утілити;
* процедури.

**Парадигми ТКПЗ**

Застосування парадигм ТКПЗ гарантує систематичний, упорядкований підхід до промислової розробки, використання і супроводу ПЗ. Фактично, парадигми вносять в процес створення ПЗ інженерний підхід та структурують процес.

Основні парадигми ТКПЗ вже розглядались в курсі цього предмету, а також програмної інженерії. До них відносять:

* класичний життєвий цикл ПЗ (каскадна/водоспадна) модель;
* системний аналіз, який визначає роль кожного елемента в комп’ютерній системі, взаємодію елементів між собою;
* аналіз вимог до ПЗ, за допомогою якого уточняються та деталізуються функції, характеристики та інтерфейс ПЗ;
* прототипування (макетування) як процес створення моделі необхідного програмного продукту, основна мета якого зняти невизначеності у вимогах замовника. Ця модель /макет може приймати одну з трьох форм: 1)паперовий макет або макет на основі ПК (зображує або малює людино-машинний діалог); 2) працює макет (виконує деяку частину необхідних функцій); існуюча програма (характеристики якої потім повинні бути поліпшені);
* стратегії конструювання ПЗ (одноразовий прохід -водоспадна стратегія, інкрементна - заплановане покращання продукту, еволюційна);
* моделі якості процесів конструювання (ISO 9001:2000, ISO/ IEC 15504, СММ).

**Типові прийоми конструювання пакетів програм складної структури**.

Аналіз складної системи вимагає її декомпозиції — розбиття на складові елементи. Зазвичай використовують такі схеми декомпозиції:

* алгоритмічну, яка застосовується в звичайних програмних системах (ПС) і основі якої лежить розбиття по діям (алгоритмам);
* об’єктно-орієнтовану, яка забезпечує розбиття по автономних об’єктах реального або віртуального світу, причому об’єкти містять в собі опис дій та опис даних.

**Керування розробкою ПЗ**

***Керування розробкою ПЗ*** – це галузь знань, в якій визначаються аспекти керування і кількісних оцінок інженерії ПЗ.

***Факти, що визначають специфіку ПЗ як продукту*:**

1. відсутність у замовника розуміння складності процесу розробки ПЗ і впливу зміни його вимог до ПЗ в процесі розробки;
2. зміна та поява нових вимог в процесі розробки ПЗ є неминучими;
3. ітеративність процесу розробки ПЗ, яка ускладнює його;
4. рівень новизни та складності ПЗ є дуже високим;
5. технології швидко змінюються, оновлюються та застарівають.

***Керування розробкою ПЗ здійснюється на 3-х рівнях:***

- організаційне керування та керування інфраструктурою;

- керування проектами;

- планування і контроль програм кількісного оцінювання показників ПЗ.

***У керуванні програмною інженерією важливими є знання щодо***:

* керування інтеграцією проекту (project integration management);
* керування змістом проекту (project scope management);
* керування термінами проекту (project time management);
* керування вартістю проекту (project cost management);
* керування якістю проекту (project quality management);
* керування людськими ресурсами проекту (project human resource management);
* керування комунікаціями проекту (project communication management);
* керування ризиками проекту (project risk management);
* керування постачанням проекту (project procurement management).

Всі специфічні факти, рівні, важливі знання, заходи, операції, прийоми щодо керування розробкою ПЗ включає в себе технологія проектування програмного забезпечення.

**Технологія проектування (розроблення) програмного забезпечення(ТППЗ)**

***Технологія*** – комплекс організаційних заходів, операцій та прийомів, спрямованих на виготовлення, обслуговування, ремонт та/або експлуатацію виробу з номінальною якістю та оптимальними витратами.

***Технологія проектування (розроблення) програмного забезпечення (ТППЗ)* –** це комплекс організаційних заходів, операцій та прийомів, спрямованих на розроблення програмних продуктів високої якості в рамках відведеного бюджету і в термін.

ТППЗ – це впорядкована сукупність взаємозв’язаних технологічних процесів в межах життєвого циклу ПЗ.

ТППЗ в загальному випадку можна описати наступною системою понять.

***Технологічний процес*** – сукупність взаємозв’язаних технологічних операцій.

***Технологічна операція*** – основна одиниця роботи, виконувана певною роллю, яка: передбачає чітко визначену відповідальність ролі; дає чітко визначений результат (набір робочих продуктів), який базується на певних початкових даних (іншому наборі робочих продуктів); представляє собою одиницю роботи з жорстко визначеними межами, які встановлюються при плануванні проекту.

***Робочий продукт*** – це інформаційна або матеріальна сутність, яка створюється, модифікується або використовується в деякій технологічній операції (модель, документ, код, тест і т.і.). Робочий продукт визначає область відповідальності ролі і є об’єктом керування конфігурацією.

***Роль* –** визначення поведінки та обов’язків окремої особи або групи осіб в середовищі організації-розробника ПЗ, які здійснюють діяльність в межах деякого технологічного процесу та відповідають за певні робочі продукти.

***Керівництво*** – це практичне керівництво по виконанню однієї або сукупності технологічних операцій. Керівництва містять методичні матеріали, інструкції, нормативи, стандарти та критерії оцінювання якості робочих продуктів.

***Інструментальний засіб*** (CASE-засіб) – це програмний засіб, який забезпечує автоматизовану підтримку діяльності, виконуваної в межах технологічних операцій.

Основною вимогою, яка висувається до сучасних ТППЗ, є їхня відповідність стандартам та нормативним документам, пов’язаним з процесами ЖЦ ПЗ та оцінкою технологічної зрілості організацій-розробників (ISO 12207, ISO 9000, CMM та ін.).

***Згідно цих нормативів, ТППЗ повинна підтримувати наступні процеси*:**

1. керування вимогами,
2. аналіз та проектування ПЗ,
3. розроблення ПЗ,
4. експлуатація,
5. супровід,
6. документування,
7. керування конфігурацією та змінами,
8. тестування, керування проектом.

Повнота підтримки процесів ЖЦ ПЗ повинна підтримуватись комплексом інструментальних засобів. Відповідність стандартам означає також, зокрема, використання загальноприйнятих стандартних нотацій та угод, стандартних методів моделювання, які повинні бути оформлені у вигляд нормативів до початку процесу проектування. Недотримання проектних стандартів ставить розробників у залежність від фірми-виробника даного засобу, ускладнює формальний контроль коректності проектних рішень, зменшує можливості залучення додаткових колективів розробників, заміни виконавців та відчуження проекту через обмежену кількість фахівців, знайомих з даним методом ***(нотацією)***.

Іншою важливою вимогою є адаптованість до умов застосування, яка досягається за рахунок поставки технології в електронному вигляді разом із CASE-засобами та бібліотеками процесів, шаблонів, методів, моделей та інших компонентів, призначених для побудови ПЗ того класу систем, на який орієнтована технологія. Електронні технології повинні містити засоби, які забезпечують їх адаптацію та розвиток за результатами виконання конкретних проектів. Процес адаптації полягає у видаленні непотрібних процесів та дій ЖЦ ПЗ, в заміні неприйнятних або у додаванні власних процесів та дій, а також методик, стандартів та керівництв.

При виборі методології розробки ПЗ слід керуватися тим, що складність методології повинна бути порівнянна з складністю структури програмного продукту, і невиправдана для продукту даної складності складність методології тільки невиправдано збільшить вартість розробки. Прикладом сучасної методології проектування може бути проблемно-орієнтоване проектування.

**Впровадження ТППЗ**

***При впровадженні ТППЗ слід керуватись рекомендаціями стандартів*** [IEEE 1209-1992, IEEE 1348-1995, ISO/IEC 14102:1995], які відбивають досвід, накопичений багатьма поколіннями користувачів та розробників ТППЗ.

***Під впровадженням ТППЗ розуміють всі дії* –** від оцінювання початкових потреб до повномасштабного використання ТППЗ ту різних підрозділах організації.

***Процес впровадження складається з наступних етапів*:**

1) визначення потреб у ТППЗ, характеристик об’єкту впровадження та проектів створення ПЗ;

2) визначення вимог, які висуваються до ТППЗ (аналіз характеристик об’єкту впровадження та проектів, обґрунтування вимог до ТППЗ, визначення пріоритетів вимог);

3) оцінювання варіантів ТППЗ – попереднє експертне оцінювання, яке полягає у аналізі доступних ТППЗ на предмет відповідності вимогам, та деталізоване оцінювання, яке полягає у формуванні детального опису кожної ТППЗ-претендента;

4) вибір ТППЗ на основі порівняльного аналізу технологій та з врахуванням експертної оцінки;

5) адаптація ТППЗ до умов застосування шляхом формування конкретної робочої конфігурації ТППЗ, адаптованої до умов об’єкту впровадження. Під час впровадження ТППЗ накопичується статистика та оцінюється ефективність її впровадження з точки зору ряду критеріїв (мінімум трудомісткості супроводу ПЗ, мінімум витрат на супровід ПЗ та ін.).

**Оцінювання та вибір ТППЗ**

***Метою процесу оцінювання ТППЗ*** є визначення функціональності та якості ТППЗ для наступного вибору. ***Оцінювання виконується відповідно до конкретних критеріїв***, його результати містять як об’єктивні, так і суб’єктивні дані по кожній ТППЗ. Процеси оцінювання й вибору тісно взаємопов’язані. За результатами оцінювання цілі вибору та/або критерії вибору, а також їхні вагові коефіцієнти можуть вимагати модифікації. В таких випадках може знадобитись повторне оцінювання. Коли аналізуються кінцеві результати оцінювання та до них застосовуються критерії вибору, то може бути рекомендовано придбання технології. Альтернативою може стати відсутність адекватної технології, в такому випадку рекомендується розробити нову технологію, модифікувати існуючу або відмовитись від впровадження.

***Процес вибору ТППЗ включає в себе наступні дії***:

1) формулювання задач вибору, включаючи цілі, припущення та обмеження;

2) виконання всіх необхідних дій по вибору, включаючи визначення та ранжування критеріїв, визначення технологій-кандидатів, збір необхідних даних та застосування ранжованих критеріїв до результатів оцінювання для визначення засобів з найкращими показниками;

3) виконання необхідної кількості ітерацій з метою вибору (або відхилення) технології, яка має подібні до інших показники.

Типовий процес оцінювання та/або вибору може використовувати набір критеріїв різних типів. Кожний критерій повинен бути обраний та адаптований експертом з врахуванням особливостей конкретного процесу. ***Початковими даними для оцінювання та вибору є набір параметрів ТППЗ:***

1) функціональні характеристики, орієнтовані на процеси ЖЦ ПЗ (керування проектом, керування вимогами, керування конфігурацією та змінами, аналіз та проектування ПЗ і т.і.);

2) функціональні характеристики застосування (середовище функціонування, сумісність з іншими ТППЗ, відповідність технологічним стандартам);

3) характеристики якості (надійність, зручність використання, ефективність, супроводжуваність, можливість переносу);

4) загальні характеристики (витрати на технологію, ліцензійна політика, оціночний ефект від впровадження ТППЗ, потрібна для впровадження інфраструктура, доступність та якість навчання, сертифікація постачальника, підтримка постачальника).

***Загальний набір критеріїв, застосовуваних для оцінювання ТППЗ, наведено у таблиці 1.***

Таблиця 1. Критерії оцінювання ТППЗ

[](https://sites.google.com/site/osnoviprogramnoieinzeneriie/lekciie/lekcia-6/%D0%BB%D0%B5%D0%BA6.png?attredirects=0)

*На основі даного набору параметрів аналізуються та класифікуються існуюча ТППЗ.* В результаті виконаного оцінювання ***може виявитись, що жодна доступна технологія не задовольняє в потрібній мірі всіх критеріїв і не покриває всі потреби проекту.*** В такому випадку може застосовуватись набір засобів, який дозволяє побудувати єдине технологічне середовище.

**Пілотний проект**

Перед повномасштабним впровадженням обраної ТППЗ виконується пілотний проект, ***метою якого є експериментальна перевірка вірності рішень, прийнятих на попередніх етапах, та підготовка до впровадження****.*

Важливою функцією ***пілотного проекту є прийняття рішення*** стосовно придбання або відмови від використання ТППЗ.

Провал пілотного проекту дозволяє уникнути подальших більш значних і коштовних невдач.

***Пілотний проект повинен мати наступні характеристики:***

1) типовість предметної галузі;

2) масштабованість;

3) презентабельність, показовість;

4) критичність;

5) авторитетність;

6) готовність проектної групи.

***В процесі оцінювання пілотного проекту*** організація повинна визначити свою позицію за наступними **трьома *питаннями:***

1) чи доцільно впроваджувати ТППЗ?;

2) які конкретні особливості пілотного проекту призвели до його успіху (або невдачі)?;

3) які проекти або підрозділи організації могли б отримати вигоду від використання ТППЗ?

***Можливим рішенням може бути одне з наступних:***

1) впровадити ТППЗ;

2) виконати додатковий пілотний проект;

3) відмовитись від ТППЗ;

4) відмовитись від використання ТППЗ взагалі.

Процес переходу до практичного використання ТППЗ починається з розроблення та наступної реалізації плану переходу, який відбиває поетапний підхід до переходу – від ретельно обраного пілотного проекту до проектів з розмаїттям характеристик, яке істотно зросло. ***План переходу повинен визначати початкову практику застосування та процедури використання засобів.*** Реальне застосування будь-якої ТППЗ в конкретних організації і проекті неможливе без відпрацювання ряду стандартів, правил, угод, яких повинні дотримуватись всі учасники проекту. Для успішного впровадження ТППЗ істотною є послідовність в її застосуванні, а також характер майбутнього використання ТППЗ як окремими розробниками, так і групами.

***Приклади провідних ТППЗ різних компаній-постачальників*:** Rational Unified Process (RUP), метод Oracle (Oracle Method),інструментальні засоби компанії Borland, які реалізують керування повним життєвим циклом додатків (Application Life Cycle Management - ALM). Розглянемо Rational Unified Process (RUP)

**Rational Unified Process (RUP) – основні принципи**

**Rational Unified Process** (**RUP**) є ітеративним процесом розробки програмного забезпечення створеним [Rational Software](https://uk.wikipedia.org/w/index.php?title=Rational_Software&action=edit&redlink=1) — підрозділом [IBM](https://uk.wikipedia.org/wiki/IBM) з 2003. RUP не є єдиним, конкретним розпорядчим процесом, а скоріше програмним каркасом, інфраструктурою програмних рішень, що полегшує розробку складних систем, або [фреймворком](https://uk.wikipedia.org/wiki/%D0%A4%D1%80%D0%B5%D0%B9%D0%BC%D0%B2%D0%BE%D1%80%D0%BA) процесу, що має бути адаптованим організаціями які займаються розробкою та командами розробників які оберуть елементи процесу, які підходять під їх потреби.

RUP в значній мірі відповідає стандартам та нормативним документам, зв’язаним з процесами ЖЦ ПЗ та оцінкою технологічної зрілості організацій-розробників. Її основні принципи представлені на рис.1.

Перший принцип є визначальним. Відповідно до нього розроблення системи виконується у вигляді декількох короткотривалих міні-проектів фіксованої тривалості (2-6 тижнів), які називаються ітераціями. Кожна ітерація включає свої власні етапи аналізу вимог, проектування, реалізації, тестування, інтеграції та завершується створенням працюючої системи. Ітераційний цикл базується на постійному розширенні та доповненні системи в процесі декількох ітерацій з періодичним зворотнім зв’язком та адаптацією подаваних модулів до ядра ПЗ.

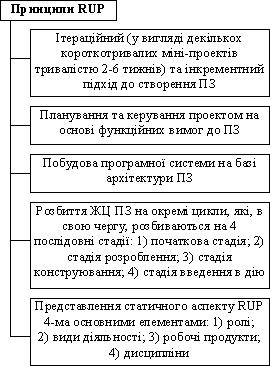
[](https://sites.google.com/site/osnoviprogramnoieinzeneriie/lekciie/lekcia-6/%D0%BB%D0%B5%D0%BA6.1.png?attredirects=0)

Рис. 1 - Основні принципи технології RUP

Згідно RUP, ЖЦ ПЗ розбивається на окремі цикли, в кожному з яких створюється нове покоління продукту.

***Кожний цикл, в свою чергу, розбивається на чотири послідовні стадії:***

1) початкова стадія (inception);

2) стадія розроблення (elaboration);

3) стадія конструювання (construction);

4) стадія введення в дію (transition).

Кожна стадія завершується в чітко визначеній контрольній точці. В цей момент часу повинні досягатись важливі результати і прийматись критично важливі рішення про подальше розроблення.

***Початкова стадія* -** це всебічне вивчення всіх можливостей реалізації проекту, яке може тривати місяці. Під час початкової стадії розробляється **бізнес-*план проекту*** – визначається, скільки він коштуватиме і який прибуток принесе, визначаються межі проекту і виконується деякий початковий аналіз для оцінювання розмірів проекту. Результати початкової стадії: загальний опис системи; початкова модель варіантів використання; початковий проектний глосарій; початковий бізнес-план; план проекту зі стадіями та ітераціями; один або декілька прототипів.

***На стадії розроблення*** виявляються більш детальні вимоги до системи, виконується високорівневий аналіз предметної області та проектування для побудови базової архітектури системи, створюється план конструювання та усуваються найбільш ризиковані елементи проекту. Найважливішим результатом стадії розроблення є опис базової архітектури (модель предметної області та технологічна платформа), яка є основою подальшого розроблення, тобто проектом для наступних стадій. Результати стадії розроблення: модель варіантів використання; перелік додаткових вимог із нефункційними вимогами включно; опис базової архітектури майбутньої системи; працюючий прототип; уточнений бізнес-план; план розроблення всього проекту з усіма ітераціями та критеріями оцінювання кожної ітерації.

Стадія розроблення займає близько п’ятої частини загальної тривалості проекту. ***Основними ознаками завершення стадії розроблення є дві події*:**

1) розробники можуть оцінити з досить високою точністю, скільки часу потрібно на реалізацію кожного варіанту використання;

2) ідентифіковані всі найбільш серйозні ризики, і ступінь розуміння найбільш важливих з них така, що відомо, як із ними впоратись.

***Результатом стадії конструювання є продукт, готовий до передачі кінцевим користувача. Як мінімум, він має містити*:** ПЗ, інтегроване на потрібних платформах; керівництва користувача; опис поточної реалізації.

***Призначенням стадії введення в дію*** є передача готового продукту у розпорядження користувачів. Дана стадія включає:

1. бета-тестування, паралельне функціонування з існуючою системою, яка підлягає заміні;
2. конвертування баз даних;
3. оптимізацію продуктивності;
4. навчання користувачів та фахівців служби супроводу.

**Основні елементи RUP.**

Статичний аспект RUP представлений 4-ма основними елементами:

1. ролі (поведінка та відповідальність особи або групи осіб);
2. види діяльності (одиниця виконуваної конкретним виконавцем роботи;
3. відповідають поняттю технологічної операції);
4. робочі продукти;
5. дисципліни (відповідає поняттю технологічного процесу і представляє собою послідовність дій, яка призводить до одержання значного результату).

RUP передбачає 6 основних дисциплін (побудова бізнес-моделей, визначення вимог, аналіз та проектування, реалізація, тестування, розгортання) та 3 допоміжних дисципліни (керування конфігурацією та змінами, керування проектом, створення інфраструктури). Кожна з перерахованих дисциплін підтримується певним інструментальним засобом комплексу Rational Suite.

***RUP спирається на інтегрований комплекс інструментальних засобів Rational Suite:***

- Rational Suite AnalystStudio – визначення та керування повним набором вимог до розроблюваної системи;

- Rational Suite DevelopmentStudio – проектування та реалізація ПЗ;

- Rational Suite TestStudio – автоматичне тестування додатків;

- Rational Suite Enterprise – підтримка повного ЖЦ ПЗ.

***До складу Rational Suite, крім самої технології RUP як продукту, входять наступні компоненти:***

- Rational Rose – засіб візуального моделювання мовою UML;

- Rational XDE – засіб аналізу та проектування;

- Rational Requisite Pro – засіб керування вимогами, призначений для організації сумісної роботи групи розробників;

- Rational Rapid Developer – засіб швидкого розроблення додатків на платформі Java 2 Enterprise Edition;

- Rational ClearCase – засіб керування конфігурацією ПЗ;

- Rational SoDA – засіб автоматичної генерації проектної документації;

- Rational ClearQuest – засіб для керування змінами та відстежування дефектів у проекті;

- Rational Quantify – засіб кількісного визначення «вузьких» місць, які впливають на загальну ефективність роботи системи;

- Rational Purify – засіб для локалізації помилок часу виконання програми, які важко виявляються;

- Rational PureCoverage – засіб ідентифікації ділянок коду, пропущених при тестуванні;

- Rational TestManager– засіб планування функціонального та навантажувального тестування;

- Rational Robot – засіб запису та відтворення тестових сценаріїв;

- Rational TestFactory – засіб тестування надійності;

- Rational Quality Architect – засіб генерації коду для тестування.

*Для самостійного вивчення*: Поглибити матеріал лекції за наданою літературою. Вивчення лекційного матеріалу та додаткових джерел. Розгляд запитань і виконання завдань для самостійної роботи, запропонованих на лекції.

*Література*

1. Лавріщева К.М. Програмна інженерія. Електронний підручник. URL: <http://csc.knu.ua/uk/library/books/lavrishcheva-6.pdf>.
2. Соммервил И. Инженерия программного обеспечения / И.Соммервил.–М. : Издательский дом «Вильямс», 2002. –623 с.
3. Якобсон А. Унифицированный процесс разработки программного обеспечения / А.Якобсон, Г.Буч, Д.Рамбо. –СПб.: Питер, 2002. – 496 с.Блэк Р. Ключевые процессы тестирования / Р.Блэк. – М.: Лори, 2006. –544 с.
4. ДСТУ ISO/IEC 14764:2002. Інформаційні технології. Супроводження програмного забезпечення. - К.: Держстандарт України, 2002.
5. 13. ДСТУ ISO/IEC 90003:2006. Програмна інженерія. Настанови щодо застосування ІSO 9001:2000 до програмного забезпечення (ІSO/ІЕС 90003:2004, IDT) - К.: Держстандарт України, 2006.
6. 14. ДСТУ ISO/IEC TR 12182:2004. Інформаційні технології. Класифікація програмних засобів (ISO/IEC TR 12182:1998, IDT) - К.: Держстандарт України, 2004.

*Контрольні запитання для самоперевірки*.

1. Дати визначення технології конструювання програмного забезпечення ***(***ТКПЗ).
2. В чому полягає призначення ТКПЗ?
3. Які методи, засоби та процедури використовуються в ТКПЗ?
4. Дати визначення технологічного процесу створення ПЗ.
5. Дати визначення технології проектування ПЗ.
6. Чим відрізняються один від одного стратегії конструювання програм?

*Контрольні запитання для надання письмових відповідей*.

1. Наведіть приклад ТППЗ, надайте узагальнений опис та порівняйте його з RUP, визначивши переваги та недоліки