

СУЧАСНІ ТЕХНОЛОГІЇ ПРОГРАМУВАННЯ

**122 «Комп'ютерні науки»
КН-18**

2019 / 2020 навчальний рік

Технології # 1

1. Програмне забезпечення: визначення, види.
2. Розвиток мов програмування
3. Технології програмування. Вимоги. Стадії.

https://github.com/eabshkvprof/2020_Mod_Prog_Techn

Програмне забезпечення

Програмне забезпечення (ПЗ / Software):
комп'ютерні програми, процедури,
документація, дані, що забезпечують
належне функціонування комп'ютерної
системи.

ВИДИ ПЗ (умовно):

Системне ПЗ (System software).

- Операційні системи (OS).
- Системи програмування.
- Сервісні програми (утиліти).

Прикладне ПЗ (Application software).

Інструментальне ПЗ.

Розвиток мов (парадигм) програмування

1. Ранні мови програмування (з 194X років)

Операторний підхід

2. Імперативне програмування (з 195X років)

Структурний підхід

Процедурний підхід

3. Декларативне програмування (з 196X років)

Функціональне

Логічне

4. Об'єктно-орієнтоване програмування (з 198X р.)

ООП

Подієве-кероване

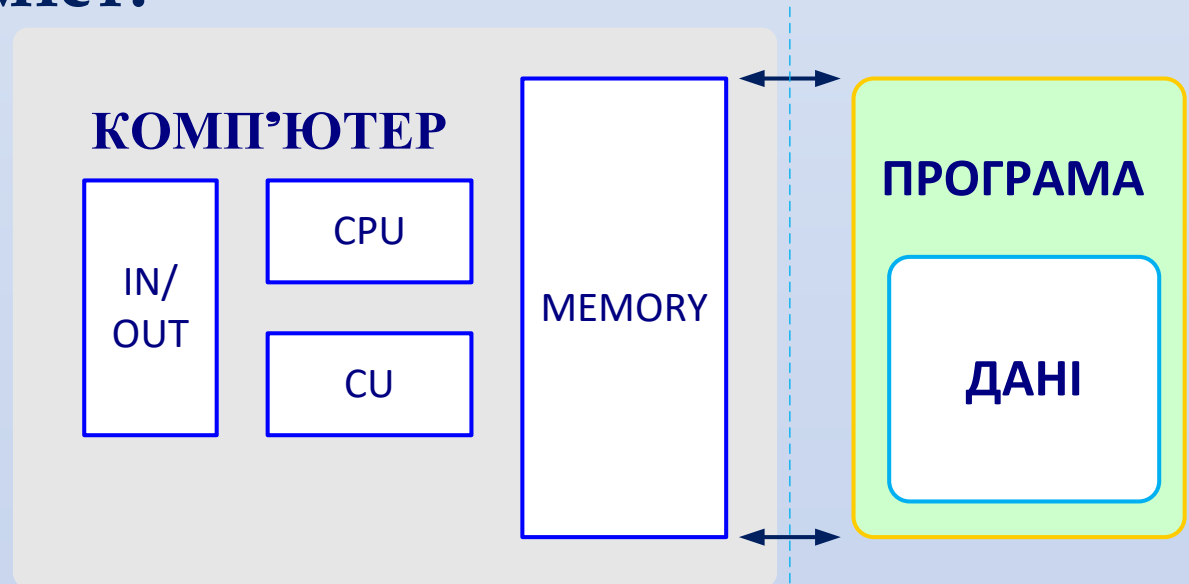
Паралельне

Компонентне програмування (з 199X р.)

Етапи розвитку програмування.

1. Ранні мови - «стихійне» програмування

- Технології відсутні!
- Програмування – це мистецтво.
- Один програміст.



Лінійна послідовність інструкцій (коди, асемблери)

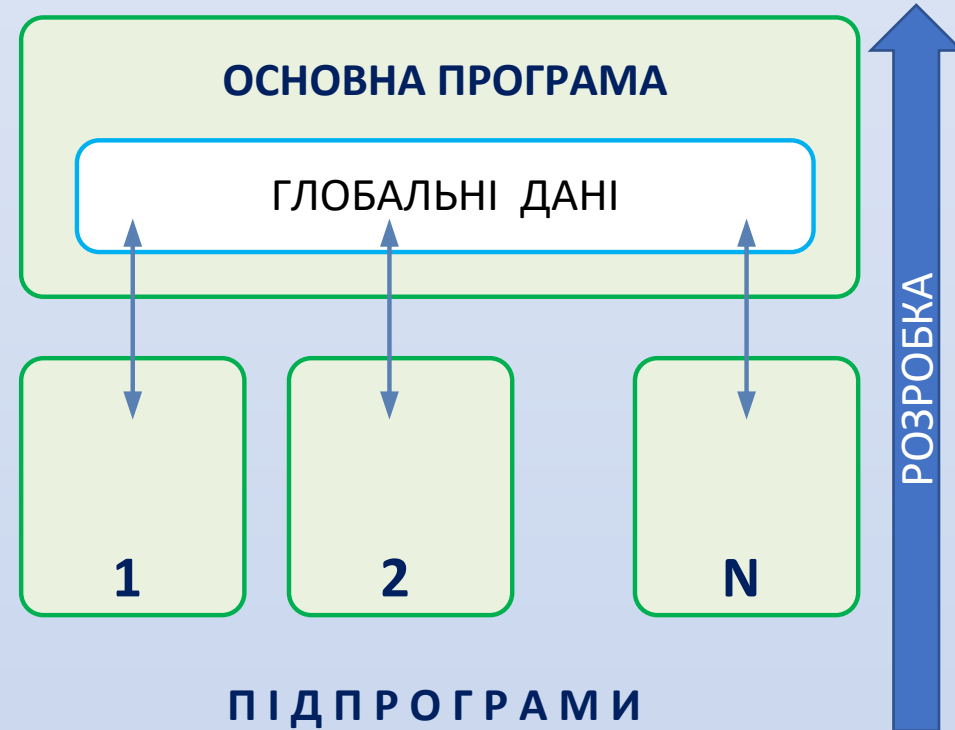
Висока ефективність

Повна залежність від hard

Етапи розвитку програмування.

1. Ранні мови - підпрограми, бібліотеки

Декілька програмістів :
підвищена вірогідність
спотворення
глобальних даних

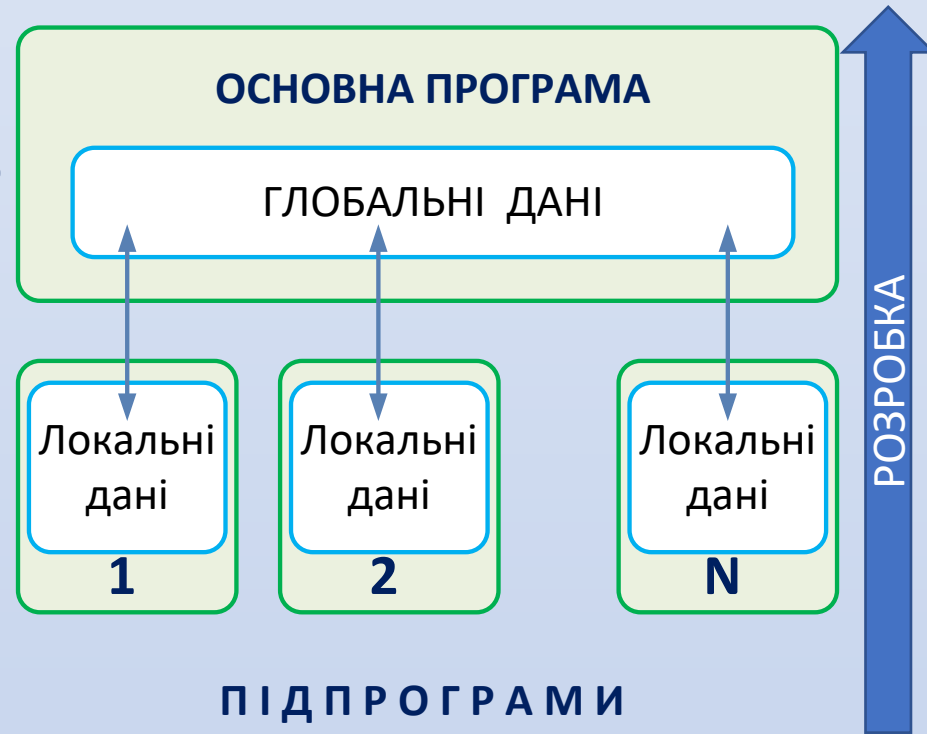


Fortran, Algol, PL/1

Етапи розвитку програмування.

2. Імперативне програмування. Структурний підхід

- Канонічні структури.
- Перші мови «високого рівня».
- Абстрагування від конкретного *hard*.
- Розгалужені підпрограми.



Fortran, Algol, PL/1
Менша залежність від *hard*
Уніфікація коду

Етапи розвитку програмування.

2. Імперативне програмування.

Процедурний підхід

Декомпозиція
Процедурне
програмування
Модулі

GLOBAL

NONLOCAL

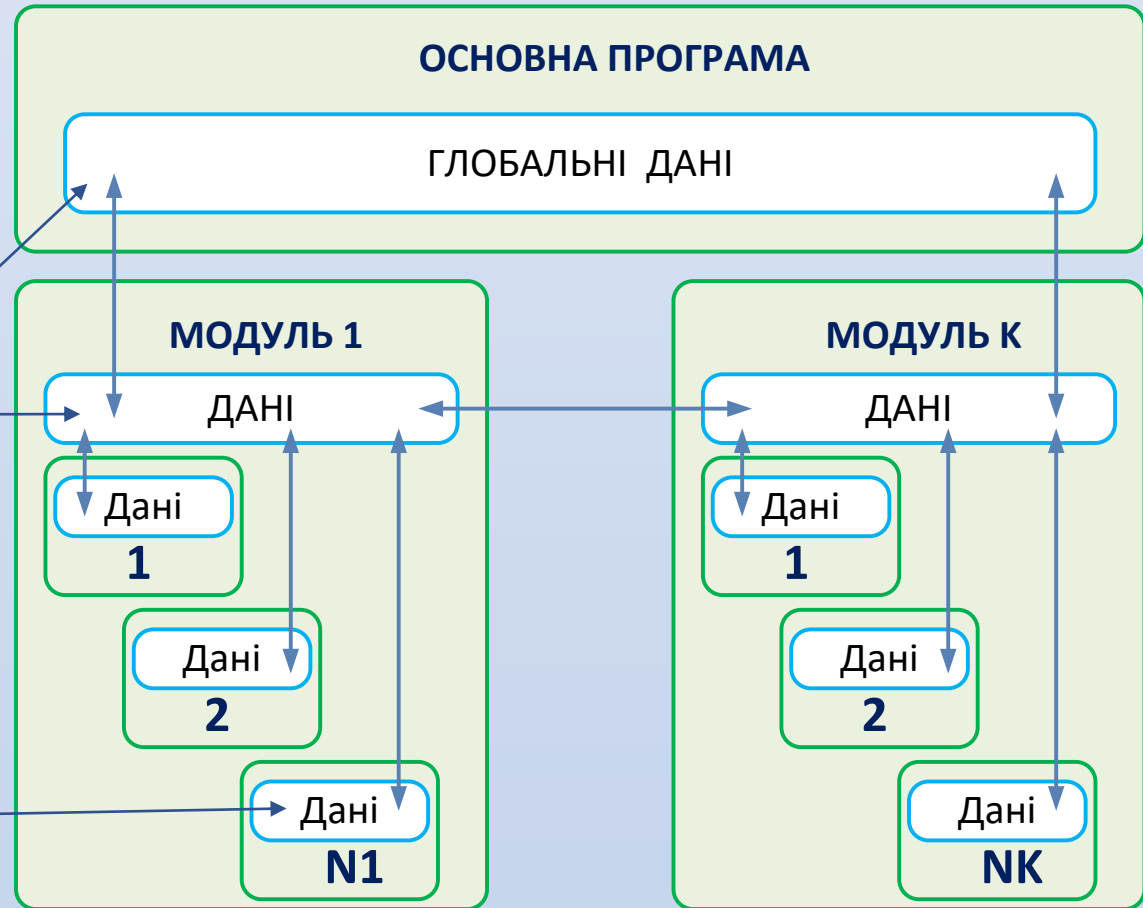
LOCAL

Pascal, C, Basic

Уніфікація коду

Підвищення надійності коду

Підвищення ефективності праці



МОДУЛІ (БІБЛІОТЕКИ)

Етапи розвитку програмування.

3. Декларативне програмування.

Програма є описом дій, які потрібно виконати, а не набором команд.

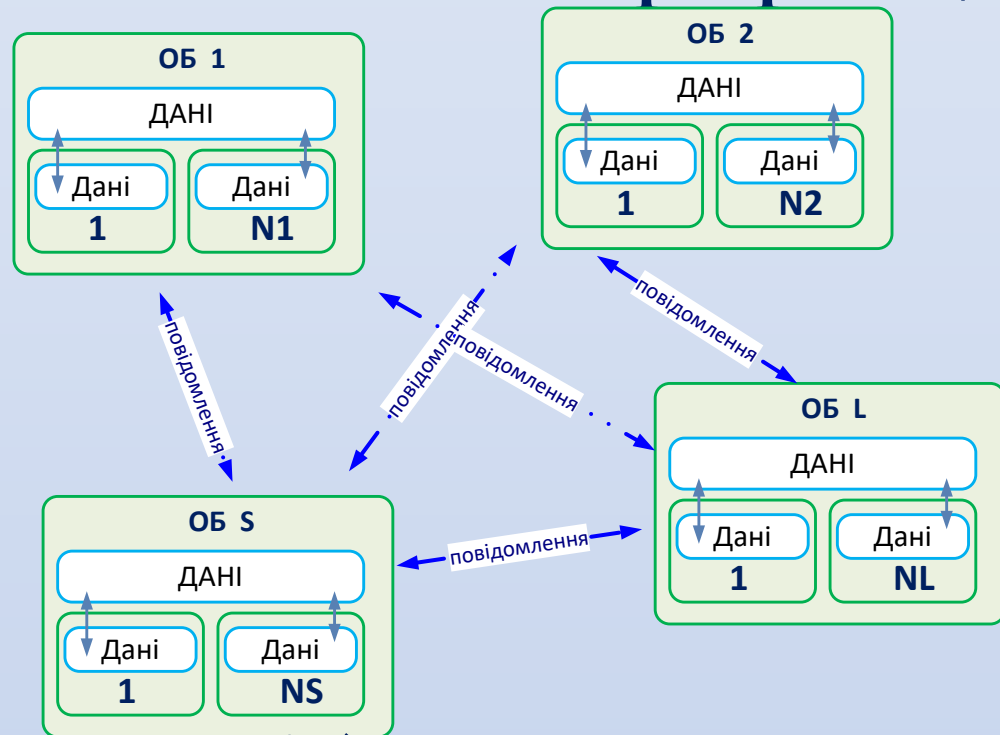
- **Функціональне програмування:** програма це функція, аргументі якої також функції. Вузька сфера застосування – символна обробка, математичні перетворення.
- **Логічне програмування:** програма є сукупністю логічних висловлювань. Базується на класичній логіці. Вузька сфера застосування – ранні системи прийняття рішень, експертні системи.

Lisp, Haskell,
Prolog

Етапи розвитку програмування.

4. Об'єктно-орієнтоване програмування

Основна ідея – наблизити текст програми до опису завдання



ОО Програма є описом об'єктів:

- властивостей (атрибутів),
- сукупностей (класів), відносин між ними,
- способів їх взаємодії
- операцій над об'єктами (методів).

Етапи розвитку програмування.

4. Об'єктно-орієнтоване програмування

ООП

- інтуїтивна близькість до довільної предметної області,
- моделювання як завгодно складних предметних областей,
- орієнтованість на події,
- високий рівень абстракції,
- повторне використання описів,
- параметризація методів обробки об'єктів.

Event-driven – програма є сукупність можливих сценаріїв (скриптів) обробки даних. Виклик сценарію ініціюється деякою подією (користувач, система).

Етапи розвитку програмування.

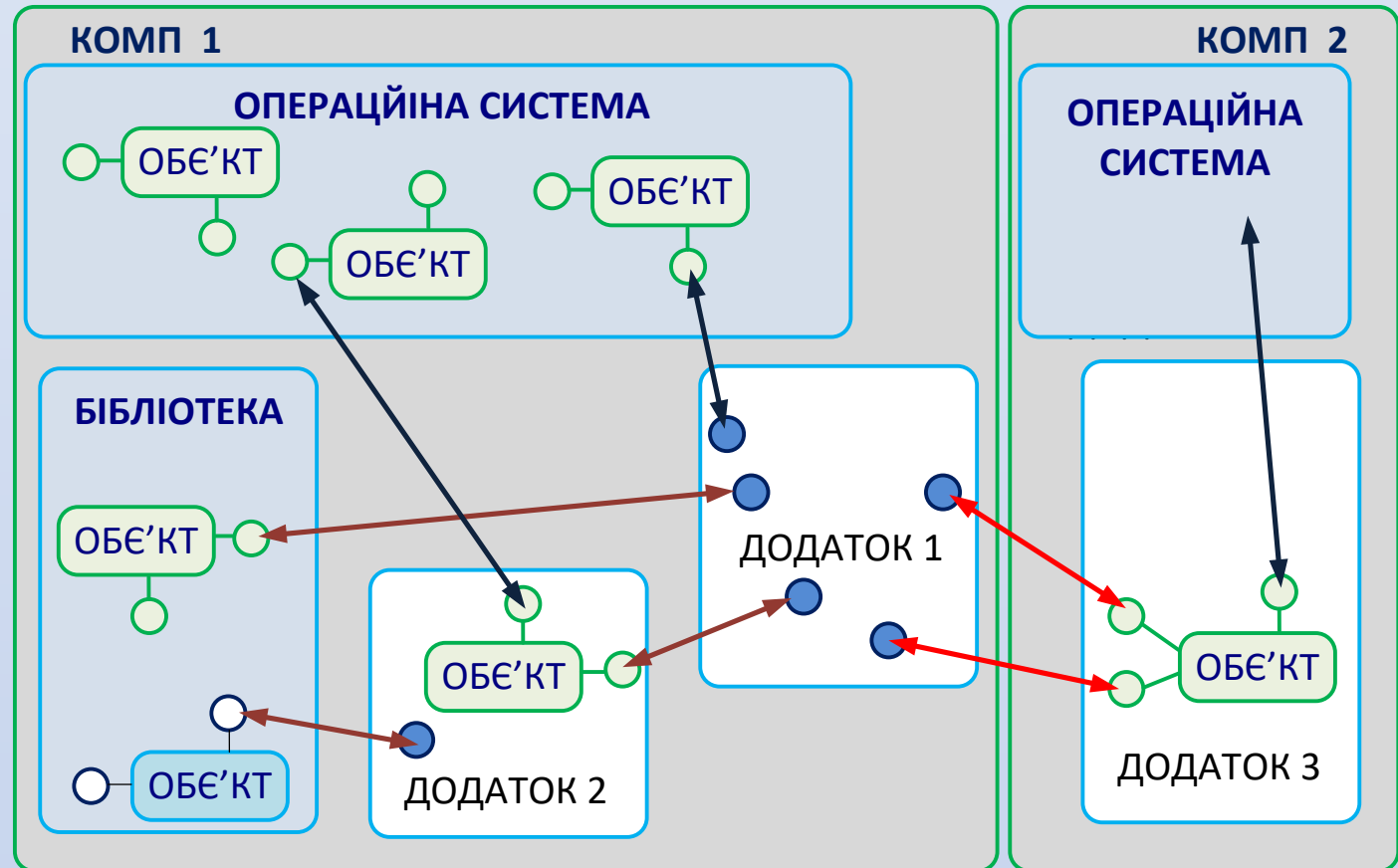
5. Компонентне програмування

Компонентне програмування визначає набір правил та обмежень, спрямованих на побудову великих програмних систем, здатних до розвитку впродовж тривалого життєвого циклу.

Програмна система складається із окремо створених елементів (компонентів), які викликають один одного через інтерфейси. Зміни в існуючу систему вносяться додаванням нових компонентів або заміною існуючих. При цьому нові компоненти, які замінюють раніше створені, повинні наслідувати інтерфейси базового.

Етапи розвитку програмування.

5. Компонентне програмування



CASE технології, шаблони

https://en.wikipedia.org/wiki/Component-based_software_engineering

Етапи розвитку програмування.

5. Компонентне програмування

Програмна ← компонента → Апаратна
взаємозамінність та надійність

Програмна компонента інкапсулює як структури даних, так і алгоритми, застосовані до структур даних.

Компонентна інженерія програмного забезпечення базується ідеях об'єктно-орієнтованого програмування та їх розвитку:

- об'єкти ПЗ,
- архітектури ПЗ,
- рамки (каркаси, frameworks) ПЗ,
- шаблони (patterns) ПЗ.

Етапи розвитку програмування.

5. Компонентне програмування

Object – деяка сутність у цифровому просторі, що володіє певним станом та поведінкою та має визначені властивості (атрибути) та операції над ними (методами).

<https://en.wikipedia.org/wiki/Object>

Software architecture – концепція побудови ПЗ: структура програмної системи та методики створення таких структур і систем. Кожна структура містить програмні елементи (об'єкти?!), відносини між ними та властивості як елементів, так і відносин.

https://en.wikipedia.org/wiki/Software_architecture

https://en.wikipedia.org/wiki/Architectural_pattern

Етапи розвитку програмування.

5. Компонентне програмування

Software framework - програмний каркас - це абстракція, в якій ПЗ, що забезпечує загальну функціональність, може вибірково змінюватися додатковим кодом, написаним користувачем, адаптуючи таким чином до потреб додатку. Можуть включати програми підтримки, компілятори, бібліотеки кодів, набори інструментів та інтерфейси прикладного програмування (API), які об'єднують всі різні компоненти, щоб забезпечити розробку проекту чи системи.

Етапи розвитку програмування.

5. Компонентне програмування

Software design pattern - загальне, багаторазове вирішення проблем, що часто зустрічається в заданому контексті в розробці ПЗ.

!!! Шаблон проектування показує, як вирішити проблему, та який може бути використано у багатьох різних ситуаціях.

Шаблони дизайну - це формалізовані **найкращі практики**, які програміст може використовувати для вирішення загальних проблем при проектуванні програми чи системи.

Технологія (програмування)

Технологія (від грец. *τεχνη* – майстерність, техніка + грец. *λογος* – передавати) – наука («корпус знань) про способи (набір і послідовність операцій, їх режими) забезпечення потреб людства за допомогою технічних засобів (знарядь, пристроїв, ...).

Технологія - комплекс наукових та інженерних знань, втілених в способах і засобах праці, наборах матеріально-речових факторів виробництва, видах їх поєднання для створення певного продукту або послуги.

Технологія програмування - сукупність методів, процесів і засобів, що застосовуються для створення програмного продукту.

Технологія (програмування)

Компоненти технологічного процесу (програмування) :

- мета реалізації процесу;
- об'єкт (предмет), що підлягає технологічним змінам;
- способи і методи впливу на об'єкт ;
- засоби технологічного впливу на об'єкт ;
- впорядкованість і організація, які протиставлені стихійним процесам.

Технологія (програмування)

Загальні вимоги до технологічного процесу (програмування) :

- високий ступінь **поділу процесу** на стадії (фази);
- системна повнота (цілісність) процесу, який повинен включати весь **набір елементів**, що забезпечують необхідну завершеність дій в досягненні поставленої мети;
- регулярність процесу і **однозначність** його фаз, їх **стандартизація і уніфікація**;
- технологія є нерозривно пов'язаною із процесом - сукупністю та **послідовністю** дій, які виконуються в часі;
- автоматизація операцій на всіх стадіях.

Технологія програмування



Технологія програмування

Особливості сучасних систем ПЗ

- Складність опису - безліч функцій, процесів, даних, складні взаємозв'язки.
- Безліч тісно взаємодіючих компонентів - мають свої локальні завдання і цілі функціонування.
- Необхідність інтеграції існуючих і нових додатків.
- Функціонування в неоднорідному середовищі, на декількох апаратних платформах, ОС.
- Істотна тимчасова протяжність створення та існування ПЗ.
- Різноманітність і роз'єднаність окремих груп розробників.

Технологія програмування

Базові стадії (фази) технологічного процесу створення програмного продукту

- Аналіз.
- Проектування.
- Програмування.
- Тестування + налагодження.
- Експлуатація + супровід.

Рекомендована ЛІТЕРАТУРА

- **Томашевський О.М., Цегелік Г.Г., Вітер М.Б., Дудук В.Ш. Інформаційні технології та моделювання бізнес-процесів. Навч. посіб. – К.: «Видавництво «Центр учбової літератури», 2012. – 296 с.**
- **Карпенко М.Ю., Манакова Н.О., Гавриленко І.О. Технології створення програмних продуктів та інформаційних систем. Навч. посіб. - Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. – Харків : ХНУМГ ім.О.М. Бекетова, 2017. – 93 с.**
- **Алексенко О.В. Технології програмування та створення програмних продуктів. Конспект лекцій. – Суми, Сумський державний університет, 2013. – 133с.**

Рекомендована ЛІТЕРАТУРА

- **Иванова Г.С. Технология программирования.** Учебник. – М.: «Кнорус», 2018. – 336 с.

Контрольні запитання

- Надайте опис етапів розвитку мов програмування. Поясніть основні переваги сучасного етапу – компонентного програмування.
- Визначте основні види сучасного програмного забезпечення. Надайте особливості сучасних систем ПЗ.
- Визначте поняття технології. Поясніть особливості технологій програмування. Надайте перелік компонентів технологічного процесу. Опишіть базові вимоги до технологічного процесу .

The END
Mod 2. Lec 1.