

Вступ.

В даний час програмування трансформувалося у цілу індустрію виробництва програмних виробів. Тому вже мало знати лише мову програмування та операційний підхід до складання алгоритмів.

Професійний розробник програмних виробів повинен мати теорію проектування, методи активізації мислення. Йому потрібне вміння оперування моделями, методами генерації рішень та вибору їх оптимальних варіантів.

Створення програмних виробів колективом розробників зумовило необхідність уміння планування робіт та їхнього розподілу між окремими учасниками проекту.

У сучасному програмуванні потрібно активне володіння дедуктивним мисленням, що не досягається шкільною та вузівською освітою. Курс містить теоретичні знання, необхідні як програмістам-кодувальникам програм, так і системним аналітикам. У ній викладаються методики оволодіння дедуктивним мисленням.

На ранніх етапах розвитку програмування, коли програми писалися як послідовностей машинних команд, будь-якої технології програмування була відсутня. Після досягнення спочатку непереборного рівня складності виникла інженерія програмування. До кінця 70-х років програмування, як правило, було роботою окремих обдарованих людей. Через недосконалість перших методик програмування навіть відносно короткі програми (довжиною близько 600 рядків) створювалися протягом кількох місяців. Початок 80-х років відповідало широкому запровадженню у практику програмування методів проектування, запозичених із техніки. Наприклад, за прикладом техніки, впроваджується ГОСТи, що регламентує стадії та етапи програмних розробок. Стандарт належить до групи стандартів єдиної системи програмної документації (ЕСПД). ЕСПД відіграла значну позитивну роль у практиці програмування та пережила без значних змін вже кілька нових технологій програмування, наприклад, технологію структурного програмування та технологію об'єктно-орієнтованого програмування.

Технологія (Матеріал з Вікіпедії – вільної енциклопедії)

Технологія(відін.-грец. τέχνη «**мистецтво**, майстерність, **вміння**» + **λόγος**«**слово**; **думка**, **сенс**, **поняття**») – сукупність **методів** і **інструментів** для досягнення бажаного результату [w1]; у широкому розумінні – застосування **наукового знання** для вирішення практичних завдань[w1][w2]. Технологія включає способи роботи, її режим, послідовність дій[w3].

Технологія є порівняно новим, багатогранним терміном, точне визначення якого вислизає через постійний розвиток сенсу цього поняття, як самого по собі, так і взятого у відносинах з іншими такими ж широкими поняттями: **культура**, **суспільство**, **політика**, **релігія**, **природа**[w4]. На початку ХХ століття термін «технологія» охоплював сукупність засобів, процесів та ідей на додаток до інструментів та машин. До середини ХХ століття поняття визначалося такими фразами як «засоби або діяльність, за допомогою яких людина змінює своє довкілля і маніпулює нею»[w5].

Технологія програмування - це наукова та практично апробована стратегія розробки програм, що містить опис сукупності методів та засобів розробки програм, а також порядок застосування цих методів та засобів.

На сьогодні поняття процесу програмування якісно змінилися. Виробництво програм набуло масового характеру, суттєво збільшився їх обсяг та складність. Розробка програмних комплексів вимагає значних зусиль великих колективів фахівців. Програми перестали бути лише обчислювальними і почали виконувати найважливіші функції з управління та обробки інформації в різних галузях науки, техніки, економіки та ін. З появою систем авоматизованого проектування (САПР) у 80-х роках були зроблені узагальнення теорії проектування технічних систем та пристроїв виявленням інваріантів як проектних процедур, особливо евристичних. Були намічені шляхи та зроблено перші спроби їхньої автоматизації. Найбільш високу складність є автоматизація ранніх етапів проектування. На цих етапах для задоволення потреби подолання дискомфорту необхідно синтезувати ідеї реалізації систем та пристроїв. Паралельний розвиток теорії програмування та теорії проектування зробив актуальним їх системне дослідження. Мета досліджень полягала у досягненні позитивного подальшого взаємного проникнення цих теорій.

У курсі розглядаються такі теми:

Перша тема містить відомості з основ теорії проектування, необхідні для ознайомлення з термінологією проектування взагалі та основними принципами проведення програмних проектів. Розглядаються такі методологічні поняття проектування як елементи системного підходу, а також одного з його найважливіших методів — блочно-ієрархічного підходу. У темі пояснюється місце стандартів програмування. Вводяться поняття життєвого циклу програмного виробу, і навіть стадій та етапів проведення програмних розробок. Розкриваються основні поняття моделювання систем та роль моделювання розробки проектів програмних систем, проводяться приклади моделей.

У другій темі розглядаються методи активізації мислення на ранніх етапах проектування програмних виробів, що дозволяє вирішити задачу вибору найкращого варіанта з множини допустимих проектних рішень, які задовольняють пред'явленим вимогам. Методи пошукового конструювання, запозичені з техніки, адаптуються стосовно програм. Розглядаються приклади видів діалогів програм, що дозволяє підвищити ефективність розробки зовнішніх функціональних специфікацій.

У третій темі викладається інженерний технологічний підхід до розробки програм, згідно з яким досягається скорочення термінів розробки програмних продуктів завдяки комбінації етапів та видів робіт, орієнтованої на різні класи програмного забезпечення та на особливості колективу розробників.

Четверта тема розкриває поняття фізичної та логічної структури даних програм. У темі розглядається набір операцій над структурами даних програм, наводиться класифікація логічних структур даних, розбираються базові структури даних, динамічні та динамічно пов'язані структури даних, і навіть файлові структури даних. Розглядаються методи документування структур даних.

П'ята тема містить опис методики розробки структурованих алгоритмів у формі проектної процедури розробки функціональних описів. Розглядаються рекомендації щодо використання проектної процедури стосовно областей, що знаходяться поза сферою програмування: техніки, організаційного забезпечення.

У шостій темі вводиться поняття архітектури програмної системи, наводяться відомості з низки способів об'єднання окремих програм у єдиний програмний комплекс.

Сьома тема містить опис технології структурного програмування, що вважається застарілою, але нині ще використовується як самостійно, і у гібридних об'єктно-орієнтованих проектах. Ряд фундаментальних ідей цієї технології ефективно використовуються сучасними технологіями.

У восьмій темі розглядається технологія об'єктно-орієнтованого проектування. Розбираються основні поняття технології. Розглядаються кроки етапів виконуваних робіт. Розглядаються приклади виконання проектів малої та середньої складності.

Дев'ята тема Містить поняття технологій візуального програмування. Ця технологія дозволяє в діалоговому режимі створювати «скелет» програми.

У десятій темі розкривається поняття САПР програмних розробок, заснованих на CASE-засобах, що дозволяють у наочній формі моделювати предметну область, аналізувати цю модель на всіх етапах розробки та супроводу програмного проекту та розробляти програми відповідно до інформаційних потреб користувачів. У розділі розглядається CASE-засіб IBM Rational Rose, призначений для автоматизації етапів аналізу та проектування програмних систем, а також для генерації кодів різними мовами та випуску проектної документації.

Одинадцята тема присвячена тестуванню програм, що дозволяють досягти заданого рівня найважливішого критерію якості програмних виробів – надійності. У розділі викладаються аксіоми тестування, прийоми налагодження, різні підходи до тестування програм.

У дванадцятій темі описуються основні засади менеджменту програмних розробок. Розглядаються принципи організації колективу розробників програмних виробів, посадові обов'язки та функції окремих працівників.

Додаток 1 необхідно для розуміння стадій і етапів розробки програм за ГОСТами, але воно не замінює, можливо, текст стандарту, що змінився.

Додаток 2 містить приклад виконання навчального технічного завдання. Цей приклад розкриває принципи складання технічного завдання, але також замінює стандарт.

Додаток 3 дає уявлення про фонд евристичних прийомів проектування програм.

Додаток 4 містить опис елементів мови програмування Object Pascal, він необхідний кращого розуміння тем 8 і 9.

Додаток 5 розкриває основні терміни та визначення, що використовуються в цьому тексті.