1. **ОГЛЯД СТАНДАРТІВ І ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**
   1. **Життєвий цикл програмного продукту**

На початку 1970-х років спеціалісти вперше заговорили про наступаючу “кризу ПЗ”. Вона була пов'язана з усе більш підсилюючимся впливом ПО на життя людей, внаслідок чого процеси розробки вимагали постійного вдосконалення. Тому була запропонована концепція життєвого циклу розробки ПО (Software life cycle, SLC). Ця модель реалізує послідовність подій, що відбуваються при розробці ПЗ. Визначення циклу SLC так само як і суперечки щодо сенсу його існування, було предметом багатьох розмов і публікацій, пов'язаних з індустрією ПО. Незважаючи на існування різних точок зору, необхідність в документально підтвердженому процесі розробки ПО залишилася настільки ж актуальною . У 1970 році У.У.Ройс (W.W.Royce) провів ідентифікацію декількох стадій в типовому циклі SLC. Саме Ройс і Баррі Боем (Barry Boehm) припустили. що здійснення контролю кожної стадії процесу розробки призведе до поліпшення якості ПО, а також до збільшення продуктивності при розробці програм. Наприклад, робота з проектування інтерфейсів програмного модуля може бути відкладена до тих нір. поки не будуть визначені кінцеві вимоги. Завдяки цьому скорочується кількість можливих переробок. В цьому випадку мова йдеться про каскадну модель циклу SLC (див. Рис. 1.1). Дії по розробці ПО "протікають" на графіку послідовно, від блоку до блоку.



Рисунок 1.1. Каскадна модель розробки ПЗ

Насправді, більшість дій в ході виконання проектів не відбуваються за лінійним законом. Найчастіше розробникам буває необхідно повернутися до попередньої стадії, щоб виконати завдання, які не були в свій час дозволені відповідним чином. Якщо на стадії розробки виявляється відсутність або неправильне формулювання вимог, розробник не просувається вперед, а повертається назад - до стадії специфікацій вимог. Після завершення і корекції специфікацій вимог повторно вводиться і починає виконуватися стадія розробки. Щоб відобразити повторюваний характер розробки ПО, в графіку рис. 1.1 було додано зворотні стрілки, в результаті чого вийшов графік стандартного життєвого індустріального циклу, показаний на рис. 1.2.



Рисунок 1.2. Повторювана каскадна модель життєвого циклу розробки ПЗ (SLC)

Функції забезпечення якості зазвичай включають такі види діяльності:

* Планування, що полягає в підготовці плану робіт, в якому знаходять відображення вимоги до якості, розподіляються завдання між виконавцями, складаються графіки і розділяється відповідальність.
* Розробку стратегії, методів і процедур, що складається в створенні нормативних інструкцій, що охоплюють всі етапи проектування, в тому числі, встановлення вимог до системи, програмування, тестування відповідно до потреб конкретного проекту. **Ключовим моментом цієї діяльності є можливо більш раннє увагу до питань якості.**
* Розвиненіша коштів, призначених для досягнення високої якості програмного забезпечення, які передбачають адаптацію наявних і створення нових ручних і автоматичних процедур, що дозволяють контролювати його відповідність встановленим вимогам, що стосуються ефективності функціонування і стандартів якості.
* Проведення ревізії, що складаються в аналізі процедур проектування та нормативної документації з точки зору їх відповідності встановленим планам стандартизації процесу розробки програмного забезпечення, в стеженні за виконанням цих планів і наявністю коректує документації.
* Спостереження за випробуваннями програмних засобів, що припускає складання звітів про випробування з аналізом проблем, причин помилок і обгрунтованому коригувальних дій.
* Збереження робочої документації, що стосується звітів про проблеми проектування і програмування системи, контрольних прикладів, протоколів випробувань, тестових даних, оглядів якості та інших аспектів.
  1. **ОГЛЯД** **СТАНДАРТІВ**