

Павло СОВ'ЯК

здобувач вищої освіти 1 курсу ОС «Бакалавр»
спеціальності 208 «Агроінженерія»

Науковий керівник : **Наталія МАРЧУК**

канд. фіз-мат. наук , доцент, асистент кафедри математики,
інформатики та академічного письма

Подільський державний аграрно-технічний університет,
м. Кам'янець-Подільський

ПРОЦЕСИ ТА ІНСТРУМЕНТИ ВЕРИФІКАЦІЇ ЯКОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Верифікація програмного забезпечення — процес посвідчення, що програми та їх компоненти виконують запропоновані їм вимоги. Метою верифікації є посвідчення в тому, що програмне забезпечення відповідає висунутим вимогам. Паралельно з цим фіксуються нові дефекти, додані в процесі розробки. Процес верифікації є складовою частиною більш загального процесу забезпечення домовленого рівня якості розроблюваної системи.

Верифікація — доказ того, що вірогідний факт або твердження є істинним. Термін використовується в залежності від того, як обґрунтовується істина: базується вона на приведенні одного доказу або аргументу - чи вона повинна підтверджуватися можливістю багаторазово відтворювання, тобто перевірятися практикою.

Верифікація націлена на скорочення помилок. Але дуже важливо розуміти, що верифікація - це контрольований ззовні процес, що демонструє наявність у системі багів і умови їх прояву.

Верифікація дозволяє гарантувати, що програмна система реалізована без непередбачуваної функціональності, відповідає висунутим вимогам, специфікаціям і стандартам. Верифікація так само керований процес.

Процес верифікації вимог до ПЗ є невід'ємною частиною всього процесу розробки. Верифікація тісно пов'язана системи. Поняття верифікації іноді плутають з поняттями валідації, тестування і навіть налагодження, і метою цього поста є внесення ясності, що є що.

У процесі розробки програмних засобів тією чи іншою мірою використовується комп'ютерна підтримка процесів розробки ПЗ. Це досягається шляхом подання хоча б деяких програмних документів ПЗ (насамперед, програм) на комп'ютерних носіях даних (наприклад, дискетах) і наданню в розпорядження розроблювача ПЗ спеціальних ПЗ або включених до складу комп'ютера спеціальних пристроїв, створених для якої-небудь обробки таких документів. У якості такого спеціального ПЗ можна вказати компілятор з якої-небудь мови програмування.

Компілятор рятує розроблювача ПЗ від необхідності писати програми мовою комп'ютера, що для розроблювача ПЗ був би вкрай незручний, - замість цього він становить програми на зручному йому мові програмування, які відповідний компілятор автоматично переводить на мову комп'ютера. Як спеціальний пристрій, що підтримує процес розробки ПЗ, може служити емулятор якої-небудь мови. Емулятор дозволяє виконувати (інтерпретувати) програми мовою, відмінною від мови комп'ютера, що підтримує розробку ПЗ, наприклад мовою комп'ютера, для якого ця програма призначена.

ПЗ, призначене для підтримки розробки інших ПЗ, будемо називати програмним інструментом розробки ПЗ, а пристрій комп'ютера, спеціально призначене для підтримки розробки ПЗ, будемо називати апаратним інструментом розробки ПЗ.

Інструменти розробки ПЗ можуть використовуватися в плинні всього життєвого циклу ПЗ для роботи з різними програмними документами. Так текстовий редактор може використовуватися для розробки практично будь-якого програмного документа. З погляду функцій, які інструменти виконують при розробці ПЗ, їх можна розбити на наступні чотири групи:

- редактори,
- аналізатори,
- перетворювачі,
- інструменти, що підтримують процес виконання програм.

Редактори підтримують конструювання (формування) тих або інших програмних документів на різних етапах життєвого циклу. Як уже згадувалося,

для цього можна використовувати один який-небудь універсальний текстовий редактор. Однак, більше сильну підтримку можуть забезпечити спеціалізовані редактори: для кожного виду документів - свій редактор. Зокрема, на ранніх етапах розробки в документах можуть широко використовуватися графічні засоби опису (діаграми, схеми й т.п.). У таких випадках досить корисними можуть бути графічні редактори. На етапі програмування (кодування) замість текстового редактора може виявитися більше зручним синтаксично керований редактор, орієнтований на використовувану мову програмування.

Аналізатори роблять або статичну обробку документів, здійснюючи різні види їхнього контролю, виявлення певних їхніх властивостей і нагромадження статистичних даних (наприклад, перевірку відповідності документів зазначеним стандартам), або динамічний аналіз програм (наприклад, з метою виявлення розподілу часу роботи програми по програмних модулях).

Перетворювачі дозволяють автоматично приводити документи до іншої форми подання (наприклад, форматери) або перекладати документ одного виду до документа іншого виду (наприклад, конвертори або компілятори), синтезувати який-небудь документ із окремих частин і т.п.

Інструменти, що підтримують процес виконання програм, дозволяють виконувати на комп'ютері опису процесів або окремих їхніх частин, представлених у вигляді, відмінному від машинного коду, або машинний код з додатковими можливостями його інтерпретації. Прикладом такого інструмента є емулятор коду іншого комп'ютера. До цієї групи інструментів варто віднести й різні відладчики. Суті-по-істоті, кожна система програмування містить програмну підсистему періоду виконання, що виконує найбільш типові для мови програмування програмні фрагменти й забезпечує стандартну реакцію на виникаючі при виконанні програм виняткові ситуації (таку підсистему ми будемо називати виконавчою підтримкою), - також можна розглядати як інструмент даної групи.

Для верифікації адекватності та повноти відображення реальних потреб користувачів необхідно проводити валідацію програмного забезпечення. Завершальним етапом верифікації програмного забезпечення замовником є

приймальне тестування. Воно направлено на верифікацію відповідності програмного забезпечення вимогам і проводиться для визначення чи задовольняє програмне забезпечення приймальним критеріям та допомагає замовнику прийняти рішення про готовність програмного забезпечення.

Список використаних джерел:

1. Величко О. М., Грабовський О. В., Гордієнко Т. Б. Особливості застосування Vмоделі при розробленні та оцінюванні якості програмного забезпечення засобів вимірювальної техніки. *Збірник наукових праць ОДАТРЯ*. 2019. Вип. 1 (14). С. 6–11.
2. Писарчук О.О., Безкоровайна Ю.М., Дишлевий О.П., Скалова В.А. Методика багатокритеріального оцінювання відповідності програмного забезпечення вимогам замовника. *Наукоємні технології*. 2019. №1 (41). Київ: НАУ, С. 3-9.
3. Кулямин В. Методы верификации программного обеспечения. Москва: Институт системного программирования РАН, 2008, 117 с.
4. Мушеник І.М. Технології дослідницького навчання і проєктивної освіти. *Освітній простір XXI ст.: виклики та перспективи: збірник наукових праць Всеукр. наук.-практ. інтернет-конф. молодих вчених і здобувачів вищої освіти (22 квітня 2021р., м. Кам'янець-Подільський)*. Кам'янець-Подільський: Подільський державний аграрно-технічний університет, 2021. С.109-114.