

ОЦІНКА "ЗДАТНОСТІ ДО СУПРОВОДУ" ТА "ЗДАТНОСТІ ДО ПЕРЕМІЩЕННЯ" СПЕЦІАЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ АВТОМАТИЗОВАНИХ ІНФОРМАЦІЙНИХ СИСТЕМ ВОЄННОГО ПРИЗНАЧЕННЯ

Міжнародні нормативні документи, що прийняті в області якості програмного забезпечення в Україні, характеризуються "здатністю до супроводу" та визначаються як "група властивостей програмного забезпечення, що характеризується зусиллями, потрібними для виконання конкретних модифікацій". Під модифікацією спеціального програмного забезпечення розуміються будь-які зміни, що не є адаптацією. Однією з підхарактеристик здатності до супроводу є аналізованість – "підгрупа властивостей здатності до супроводу (СПЗ), що обумовлює їх пристосованість до діагностики недоліків або відмов і виявленню частин, що мають бути змінені та прогнозу наслідків цих змін". Тестування СПЗ після внесення в нього змін є однією з багатьох технологічних операцій, що виконуються під час супроводу. На рівень тестованості, як підхарактеристики здатності до супроводу, впливає повнота та якість програмної документації, наявність засобів генерації тестових наборів, а також засобів аналізу результатів тестування; повнота тестових наборів і дотримання правил їх складення, що в свою чергу визначає трудомісткість процесу тестування. В статті показано результати оцінки "здатності до супроводу" та "здатності до переміщення" спеціального СПЗ автоматизованих інформаційних систем воєнного призначення.

Наведені в статті метрики дозволяють оцінити та отримати чисельні значення цих параметрів. На їх основі можливо порівнювати відповідні програми за цією характеристикою якості. Крім того, в роботі розглянуті метрики, що базуються на аналізі вихідних текстів програм, графів та супроводжувальної документації, що забезпечує єдиний підхід до автоматизації їх обчислення.

Ключові слова: спеціальне програмне забезпечення, автоматизовані інформаційні системи, модифікація, тестування, метрики.

Вступ. В попередній статті авторів [1] було досліджено теоретичні основи та показана практична реалізація системи оцінки «практичності» та «коректності» спеціального забезпечення автоматизованих інформаційних систем воєнного призначення.

Авторами заплановано видання серії наукових статей, які в цілому спрямовані на розв'язання наступних взаємопов'язаних задач:

1. Визначення номенклатури комплексних показників у вигляді ієрархічної схеми до рівня, коли останні можна спроекувати на числовий інтервал.
2. Проведення класифікації спеціального програмного забезпечення (СПЗ) воєнного призначення та визначення пріоритетності показників-складових в ієрархічній схемі СПЗ воєнного призначення в залежності від специфіки його застосування.
3. Створення методик оцінки якості за інтегральним показником для будь-якого типу (класу) СПЗ.
4. Розробка моделі оцінки якості СПЗ у вигляді сукупності методик числового оцінювання показників-складових.
5. Створення методологічного апарату підвищення ефективності та гарантування цільової якості СПЗ.

Аналіз відомих досліджень та постановка задач. У останніх наукових розробках та публікаціях, що відомі авторам [1-5], розглянуті питання забезпечення практичності та коректності спеціального програмного забезпечення та зроблені основи оцінки здатності по

супроводу та переміщенню програмного забезпечення автоматизованих інформаційних систем [6-8]. У фундаментальній роботі Державного інституту стратегічного дослідження за редакцією Академіка Горбуліна В.П. [9] розглянуто низка гуманітарних, політичних, економічних, військових та безпекових питань. Деякі розділи цієї роботи присвячені питанням забезпечення інформаційної безпеки, кіберзлочинності та якості програмного забезпечення систем воєнного призначення. Аналіз проблеми в цілому ставить низку актуальних завдань створення воєнної науково-технічної системи формальних методів та засад, яка б гарантувала належний рівень якості СПЗ відповідно до його цільового призначення, та забезпечила можливість коректної та адекватної її оцінки.

Дійсна стаття присвячена одному із цих напрямків оцінки "здатності до супроводу" та "здатності до переміщення" спеціального програмного забезпечення автоматизованих інформаційних систем воєнного призначення.

Основні результати досліджень. Згідно міжнародних стандартів, та нормативних документів, які прийняті в області якості програмного забезпечення в Україні, характеристика "здатність до супроводу" визначається як "група властивостей програмного забезпечення, що характеризується зусиллями, потрібними для виконання конкретних модифікацій". При цьому підкреслюється, що "під модифікацією ПЗ розуміються будь-які зміни, що не є адаптацією (зміни, що проводяться виключно з метою забезпечення функціонування СПЗ на конкретних технічних засобах або під керуванням конкретних ПЗ)" [10].

Однією з підхарактеристик здатності до супроводу є *аналізованість* – "підгрупа властивостей здатності до супроводу СПЗ, що обумовлює їх пристосованість до діагностики недоліків або відмов і виявленню частин, які мають бути змінені, і прогнозу наслідків цих змін" [10]. Для кількісної оцінки цієї підхарактеристики пропонується використовувати такі показники:

1. Коефіцієнт повноти документації, $K_{fulldoc}$. Враховується наявність тексту програм (0, 1), опису програми (0, 1), керівництво програміста (0, 1), керівництво системного програміста (0, 1):

$$K_{fulldoc} = \frac{\sum_{i=1}^4 N_i^{(doc)}}{4}, \quad (1)$$

де $N_i^{(doc)}$ - одиничні показники ступеню реалізації повноти документації.

2. Середній час локалізації несправності, \overline{T}_{loc} , який розраховується за формулою:

$$\overline{T}_{loc} = \frac{1}{N_{fault}} \sum_{i=1}^{N_{fault}} t_i^{(fault)}, \quad (2)$$

де N_{fault} - кількість несправностей, що були локалізовані за певний період експлуатації програмного виробу;

$t_i^{(fault)}$ - час локалізації i -ї несправності.

На основі значення показника \overline{T}_{loc} можливо зробити висновок також про ступінь зрозумілості як підхарактеристики практичності СПЗ.

Підхарактеристика *коригованість* або *відкритість* СПЗ характеризується "зусиллями, необхідними для усунення недоліків, які виявлені в ПЗ" [9,10]. Для її оцінки пропонується використовувати коефіцієнт відкритості, K_{corr} , який характеризує пристосованість програмного виробу до внесення виправлень, змін та доповнень як до тексту програми, та і до супроводжувальної документації:

$$K_{corr} = \frac{\overline{Prod}_{corr}}{\overline{Prod}_{start}}, \quad (3)$$

де \overline{Prod}_{corr} - середня продуктивність праці програмістів під час коригування програми;

$Prod_{start}$ - середня продуктивність праці програмістів під час написання початкового тексту програми.

Тестування СПЗ після внесення в нього змін є однією з багатьох технологічних операцій, що виконуються під час супроводу. На рівень *тестованості* як підхарактеристики здатності до супроводу СПЗ впливає повнота та якість програмної документації, наявність засобів генерації тестових наборів, а також засобів аналізу результатів тестування; повнота тестових наборів і дотримання правил їх складення, що в свою чергу визначає трудомісткість процесу тестування [11]. Поняття повноти тестового набору пов'язане з можливістю перекриття всіх логічних зв'язків у програмі. Тому пропонується оцінювати ступінь повноти тестів за допомогою відповідного коефіцієнта $K_{fulltest}$, який розраховується за формулою:

$$K_{fulltest} = \frac{N_m}{N_{test}}, \quad (4)$$

де N_m - загальна кількість логічних маршрутів в програмі;

N_{test} - загальна кількість підготовлених тестових наборів.

Процес тестування передбачає підготовку набору тестів, машинний експеримент на тестовому наборі та аналіз результатів тестування. Тому одним з показників технологічності супроводу є трудомісткість тестування, оцінювати яку пропонується за допомогою відповідного коефіцієнту, K_{test} :

$$K_{test} = \frac{\sum_{i=1}^{N_{new}} T_i}{\sum_{j=1}^{N_{test}} T_j}, \quad (5)$$

де T_i - трудомісткість підготовки, прогону та аналізу результатів i -го тесту, результатом виконання якого стало виявлення помилки;

N_{new} - кількість тестів, що привели до виявлення нових помилок;

T_j - трудомісткість підготовки, прогону та аналізу результатів j -го тесту з повного набору підготовлених тестів.

Згідно стандарту ISO/IEC JTC1/SC7/WG6 – Evaluation and Metrics [12] "метрики *стабільності* повинні надавати можливість оцінити неочікуваний режим роботи системи, включаючи програмне забезпечення в ході його тестування або функціонування після модернізації". Показником, що характеризує стабільність програмного продукту під час розробки нових версій і враховує зміни, зроблені у ході його удосконалення, згідно стандарту ISO/IEC JTC1/SC7/WG6 – Evaluation and Metrics є індекс завершеності SMI , що розраховується за формулою:

$$SMI = \frac{[M_t - (F_a + F_c + F_d)]}{M_t}, \quad (6)$$

де M_t - кількість модулів поточної версії;

F_c - кількість модулів поточної версії, що зазнали змін;

F_a - кількість доданих модулів поточної версії;

F_d - кількість модулів попередньої версії, вилучених із поточної версії.

Будемо вважати, що продукт має тенденцію до стабілізації, якщо індекс завершеності наближається до 1.

Показники *структурності (модульності)* характеризують ступінь досконалості методів декомпозиції та організації взаємозв'язків між компонентами програмного продукту. Для кількісної оцінки цієї підхарактеристики здатності до супроводу СПЗ пропонується використовувати коефіцієнт модульності K_{module} :

$$K_{module} = \frac{m_{str}}{m}, \quad (7)$$

де m_{str} - кількість модулів (складових частин) програмного продукту, при кодуванні яких було дотримано правил структурного програмування;

m - загальна кількість модулів в програмному продукті.

Однією з основних цілей науково-технічної підтримки створення СПЗ є зменшення його складності, як підхарактеристики здатності до супроводу. Саме це дозволяє знизити трудомісткість проектування, розробки, іспитів та супроводження, забезпечити простоту та надійність СПЗ. Цілеспрямоване зниження складності СПЗ являє собою багатокрокову процедуру і потребує попереднього дослідження існуючих показників складності, проведення їх класифікації та співвіднесенням з типами СПЗ і використанням на різних етапах життєвого циклу.

Теорія складності програм орієнтована на управління якістю СПЗ та контроль її еталонної складності в період експлуатації. На теперішній час існує велика кількість показників, які тією чи іншою мірою описують складність програм і за цих умов для їх використання потребується попереднє упорядкування.

При оцінці складності СПЗ будемо виділяти дві основні групи метрик:

- метрики розміру СПЗ;
- метрики складності потоку управління СПЗ;

Метрики *розміру СПЗ* найбільш прості і, очевидно, тому отримали широке розповсюдження. Традиційною характеристикою розміру програм є кількість інтелектуальних елементів вихідного тексту. Під такими елементами будемо розуміти будь-який виконуємий оператор. Очевидно, що оцінка розміру СПЗ недостатня для прийняття рішення щодо його складності, але може бути застосована для класифікації СПЗ, які суттєво відрізняються об'ємами. При зменшенні різниці в об'ємах на перший план висуваються оцінки інших факторів, які впливають на складність. Таким чином, метрика розміру СПЗ є оцінкою за номінальною шкалою, на основі якої визначаються тільки категорії СПЗ без уточнення оцінки для кожної категорії. Найбільш відомою серед оцінок складності програм є метрика М. Холстеда [13]. Її основу складають чотири характеристики програми:

n_1 – кількість унікальних операторів програми (словник операторів);

n_2 – кількість унікальних операндів програми (словник операндів);

N_1 – загальна кількість операторів у програмі;

N_2 - загальна кількість операндів у програмі.

Спираючись на ці характеристики (які можливо отримати безпосередньо в результаті аналізу вихідних текстів програм), М. Холстед вводить наступні оцінки [13]:

словник програм:

$$n = n_1 + n_2 ,$$

довжину програми:

$$N = N_1 + N_2 , \quad (8)$$

об'єм програми:

$$V = N \log_2(n) (\text{біт}). \quad (9)$$

Під бітом розуміється логічна одиниця інформації – символ, оператор, операнд.

Далі М. Холстед вводить n^* - теоретичний словник програми, тобто словарний запас, який необхідний для написання програми з урахуванням того, що необхідна функція вже реалізована в даній мові програмування. Наприклад, можливе здійснення процедури визначення визначення положення (у вигляді цілого числа) першого входження символного рядку всередині іншого символного виразу або Метод-поля (СУБД Visual FoxPro) може

оцінюватись так:

АТС (X, Y, n),

де X – символний рядок;

Y – символний вираз або Мето-поле;

n – номер позиції.

Теоретичний словник в цьому випадку буде складатися з: $n_1^* : \{\text{АТС}(\dots)\}=1$; $n_2^* : \{X, Y, n\}=3$, а його довжина $n^* = n_1^* + n_2^* = 1+3=4$. Використовуючи n^* , М. Холстед вводить оцінку V^* , за допомогою якої описується потенційний об'єм коду, який відповідає максимально компактному тексту програми, що реалізує даний алгоритм:

$$V^* = n^* \log_2(n^*). \quad (10)$$

За допомогою метрик складності потоку управління СПЗ оперують щільністю керуючих переходів всередині програм або взаємозв'язками цих переходів. В обох випадках програму можливо представити у вигляді керуючого орієнтованого графа $G = (V, E)$, де V - вершини, що відповідають операторам, E - дуги, що відповідають переходам. Вперше графічне представлення програм було запропоновано Мак-Кейбом. В якості основної метрики складності він пропонує застосовувати цикломатичну складність графа програми (циклوماتичне число Мак-Кейба, яке характеризує трудомісткість тестування програми). Для обчислення цикломатичного числа Мак-Кейба $Z(G)$ застосовується формула:

$$Z(G) = N_{\text{arc}} - N_{\text{gn}} + 2p,$$

де N_{arc} – число дуг орієнтованого графа G;

N_{gn} – число вершин графа;

p – число компонентів зв'язності графа.

В якості числа компонентів зв'язності графа пропонується застосовувати кількість дуг, які необхідно додати для перетворення графа у сильнозв'язний (граф, будь-які дві вершини якого взаємно досяжні). Ця послідовність пояснюється тим, що граф коректної програми не має недосяжних від точки входу ділянок. Сильнозв'язаний граф коректної програми отримується, як правило, шляхом замикання дугою вершини, що позначає кінець програми, на вершину, що позначає точку входу в цю програму. По суті $Z(G)$ визначає число лінійно незалежних контурів в сильнозв'язному графі, тобто потрібну кількість проходів для покриття всіх контурів графа або кількість тестових прогонів програми, які необхідні для вичерпного тестування за критерієм "працює кожна гілка". Недоліком цикломатичного числа є те, що воно залежить тільки від кількості предикатів, складність яких не враховується. Наприклад, є два оператори умови:

```
IF X>0
```

```
  THEN X=A
```

```
  ELSE
```

```
  ...
```

```
ENDIF
```

та

```
IF ((X>0) AND (IND=1)) OR ((X=0) AND (IND=0))
```

```
  THEN X=A
```

```
  ELSE
```

```
  ...
```

```
ENDIF
```

Обидва оператори припускають єдине розгалуження і можуть бути представлені одним і тим же графом, тому цикломатичне число буде для обох операторів однаковим, не відображаючи складності предикатів, що дуже суттєво при оцінці складності програм.

Ще однією достатньо ефективною оцінкою складності програм є метрика Т. Джилба, в якій логічна складність програми визначається як насиченість коду виразами типу IF-THEN-ELSE. При цьому вводяться дві характеристики: абсолютна складність програми, яка характеризується кількістю операторів умови, C_{abs} та відносна складність програми, яка характеризується насиченістю коду операторами умови, C_{rel} . Тобто, C_{rel} визначається як:

$$C_{rel} = \frac{C_{abs}}{N_{oper}}, \text{ де } N_{oper} - \text{загальна кількість операторів в програмі.}$$

Характеристика "здатність до переміщення" характеризується як "група властивостей програмного забезпечення, що обумовлює його пристосованість для переносу із одного середовища в інші". При цьому під переносом розуміється "сукупність дій, які спрямовані на забезпечення функціонування ПЗ у середовищі, яке відмінне від того, в котрому воно використовувалося раніше, без суттєвих відмінностей у поведінці" [10,14]. При кількісній оцінці цієї характеристики якості СПЗ обов'язково слід враховувати, що адаптація програмного виробу до нового середовища має сенс тільки у тому випадку, коли витрати на адаптацію не перевищують витрат на розробку нового програмного виробу. Враховуючи це, пропонується оперувати коефіцієнтом адаптованості, K_{adapt} :

$$K_{adapt} = 1 - \frac{R_{adapt}}{R_{start}}, \quad (11)$$

де R_{adapt} - витрати ресурсів, необхідних для адаптації програмного виробу до нового середовища експлуатації;

R_{start} - витрати ресурсів на створення нового програмного виробу у новому середовищі.

Зусилля щодо адаптації ПВ до нового середовища функціонування можуть бути зменшені завдяки уніфікації програмних виробів та їх компонентів шляхом створення типових модулів, які можуть використовуватись в кількох програмних виробках. Оцінка ступеню уніфікації ПВ може бути здійснена за допомогою коефіцієнту уніфікації, K_{unif} :

$$K_{unif} = \frac{n - n_o}{n}, \quad (12)$$

де n - загальна кількість модулів в ПВ;

n_o - кількість оригінальних складових частин ПВ.

Ступінь універсалізації алгоритму роботи та програмного коду ПВ можливо оцінити за допомогою відповідного коефіцієнту, K_{globb} :

$$K_{globb} = \frac{N_{globb}}{n}, \quad (13)$$

де N_{globb} - кількість підпрограм, пристосованих для використання в будь-яких модулях програмного виробу.

Таким чином, наведені метрики дозволяють оцінити та отримати чисельні значення здатності до супроводу та здатності до переміщення СПЗ, на основі яких можливо порівнювати програми за цією характеристикою якості. Крім того, розглянуті метрики базуються на аналізі вихідних текстів програм, графів та супроводжувальної документації, що забезпечує єдиний підхід до автоматизації їх обчислення.

Висновки. Дана стаття є другою в серії наукових робіт авторів за наступними напрямками: визначення номенклатури комплексних показників у вигляді ієрархічної схеми до рівня, коли останні можна спроектувати на числовий інтервал; проведення класифікації СПЗ воєнного призначення та визначення пріоритетності показників-складових в ієрархічній схемі СПЗ воєнного призначення в залежності від специфіки його застосування; створення методик оцінки якості за інтегральним показником для будь-якого типу (класу) СПЗ; розробка

моделі оцінки якості СПЗ у вигляді сукупності методик числового оцінювання показників-складових; створення методологічного апарату підвищення ефективності та гарантування цільової якості СПЗ.

Наведені метрики дозволяють оцінити та отримати чисельні значення здатності до супроводу та здатності до переміщення СПЗ, на основі яких можливо порівнювати програми за цією характеристикою якості. Крім того, розглянуті метрики базуються на аналізі вихідних текстів програм, графів та супроводжувальної документації, що забезпечує єдиний підхід до автоматизації їх обчислення.

ЛІТЕРАТУРА:

1. Ленков С.В., Гришак О.М., Жиров Г.Б., Пампуха І.В. Оцінка "практичності" та "коректності" спеціального програмного забезпечення автоматизованих інформаційних систем воєнного призначення. Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка. К.: ВІКНУ, 2022. № 74. С. 83-89.

2. Leffingwell D. Calculating the return on investment from more effective requirements management. *American Programmer*, №10(4), 1997. Pp. 13-16.

3. Barlas S. Anatomy of a Runaway: What Grounded the AAS // *IEEE Software*, January 1996. - Pp. 104-106.

4. Grady R.B. An economic release decision model: insights into software project management // In proceedings of the applications of software measurement conference, Orange Park, FL: Software Quality Engineering, 1999. Pp. 227-239.

5. Gibbs W. Software's Chronic Crisis // *Scientific America*, September 1994. - P. 86-95.

6. Основы инженерии качества программных систем / Ф.И. Андон, Г.И. Коваль, Г.М. Коротун, В.Ю. Суслов: НАН Украины, Институт программных систем. - К.: Академперіодика, 2002. - 503с.

7. Інформаційні технології. Оцінювання процесів життєвого циклу програмних засобів. – Офіційне видання. – К.: Держспоживстандарт України, 2004. (Національний стандарт України).

8. Толубко В. Б., Сбітнев А. І., Пермяков О. Ю. Методологічні основи проектування прикладного програмного забезпечення для АСУ воєнного призначення: Монографія, - К.: НАОУ, 2004. -249 с.

9. Світова гібридна війна: український фронт / За заг.ред. В.П. Горбуліна. Національний інститут стратегічних досліджень. – К.: НІСД, 2017. – 496 с.

10. ДСТУ 2850-94 Програмні засоби ЕОМ. Показники і методи оцінювання якості.

11. Молодцова О.П. Управління якістю програмної продукції: Навч. посібник // Київський національний економічний університет. – К.: КНЕУ, 2001. – 248с.

12. ISO/IEC 9126-2 Information Technology – Software quality characteristics and metrics – Part 2: External metrics.

13. Холстед М.Х. Начало науки о программах. – М.: Финансы и статистика, 1981. – 128с.

14. Характеристики качества программного обеспечения / Б. Бозм, Дж. Браун, Х. Каспар и др. Пер. с англ. Е.К. Масловского. - М.: Мир, 1981. – 206 с.

REFERENCES:

1. Ljenkov S.V., Gryshhak O.M., Zhyrov G.B. and Pampuha I.V. (2022). Ocinka "praktychnosti" ta "korektnosti" special'nogo programnogo zabezpechennja avtomatyzovanyh informacijnyh system vojennogo pryznachennja. *Collection of scientific works of the Military Institute of Kyiv National Taras Shevchenko University*. Kyiv: MIKNU. No 74. Pp. 83-89.

2. Leffingwell D. Calculating the return on investment from more effective requirements management. *American Programmer*, №10(4), 1997, Pp. 13-16.

3. Barlas S. Anatomy of a Runaway: What Grounded the AAS // *IEEE Software*, January 1996. Pp. 104-106.

4. Grady R.B. An economic release decision model: insights into software project management // In proceedings of the applications of software measurement conference, Orange Park, FL: Software Quality Engineering, 1999. Pp. 227-239.

5. Gibbs W. Software's Chronic Crisis // *Scientific America*, September 1994. Pp. 86-95.

6. Andon F.I., Koval' G.I., Korotun G.M. and Suslov V.Ju. (2002) *Osnovy inzhenerii kachestva programmnyh sistem*. NAN Ukrainy, Institut programmnyh sistem. Kyiv, Akademperiodika. 503 p.

7. Informacijni tehnologii'. (2004). Ocinjuvannja procesiv zhyttjevogo cyklu programnyh zasobiv. Oficijne vydannja. Kyiv: Derzhspozhyvstandart Ukrai'ny, (Nacional'nyj standart Ukrai'ny).
8. Tolubko V.B., Sbitnjev A.I. and Permjakov O.Ju. (2004). Metodologichni osnovy proektuvannja prykladnogo programnogo zabezpechennja dlja ASU vojnogo pryznachennja. Kyiv: NAOU, 249 p.
9. Svitova gibrydna vijna: ukrai'ns'kyj front. Za zag.red. V.P. Gorbulina. (2017). Nacional'nyj instytut strategichnyh doslidzhen'. Kyiv: NISD, 496 p.
10. DSTU 2850-94 Programni zasoby EOM. Pokaznyky i metody ocinjuvannja jakosti.
11. Molodcova O.P. (2001). Upravlinnja jakistju programnoi' produkci. Kyi'vs'kyj nacional'nyj ekonomichnyj universytet. Kyiv, KNEU, 248p.
12. ISO/IEC 9126-2 Information Technology – Software quality characteristics and metrics – Part 2: External metrics.
13. Holsted M.H. (1981). Nachalo nauky o programmah. M.: Fynansy y statystyka, 128 p.
14. B. Воэм, Dzh. Braun, H. Kaspar y dr. (1981). Harakterystyky kachestva programnogo obespechenja. Per. s angl. E.K. Maslovskogo. M.: Myr, 206 p.

Doctor of Technical Science, Professor Lienkov S.V.,
Doctor of Technical Science, Professor Gusak Yu.A.,
Doctor of Technical Science, Professor Selyukov O.V.,
PhD Pampukha I.V., Solodeeva L.V.

ASSESSMENT OF "MAINTENANCE CAPACITY" AND "MOVABILITY" OF SPECIAL SOFTWARE FOR AUTOMATED INFORMATION SYSTEMS IN SYSTEMS

International normative documents adopted in the field of software quality in Ukraine are characterized by "ability to maintain" and are defined as "a group of software properties characterized by the effort required to perform specific modifications." Modification of special software means any changes that are not adaptations. One of the sub-characteristics of the ability to support is analysis - "a subset of properties of the ability to support (SDR), which determines their suitability for diagnosing defects or failures and identifying parts to be changed and predict the consequences of these changes." Testing of SDRs after making changes to it is one of the many technological operations performed during maintenance. The level of testing, as a sub-characteristic of the ability to support, is influenced by the completeness and quality of software documentation, the availability of tools for generating test kits, as well as tools for analyzing test results; completeness of test kits and compliance with the rules of their compilation, which in turn determines the complexity of the testing process. The article shows the results of the assessment of the "ability to support" and "ability to move" special SDR automated information systems for military purposes.

The metrics presented in the article allow us to evaluate and obtain numerical values of these parameters. Based on them, it is possible to compare the relevant programs for this quality characteristic. In addition, the paper considers metrics based on the analysis of the source texts of programs, graphs and supporting documentation, which provides a unified approach to automating their calculation.

Keywords: *special software, automated information systems, modification, testing, metrics.*