

Кіровоградський національний технічний університет

**ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ
СИНТЕЗУ ТЕСТОВИХ МОДЕЛЕЙ ПОВЕДІНКИ ОБ'ЄКТІВ
ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ АСУ НА СТАДІЇ
ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОЕКТУВАННЯ**

Доповідач

Доренський Олександр Павлович,
викладач кафедри програмного
забезпечення

Науковий керівник

Смірнов Олексій Анатолійович,
доктор технічних наук, професор,
завідувач кафедри програмного
забезпечення

Кіровоград - 2015

Актуальність теми

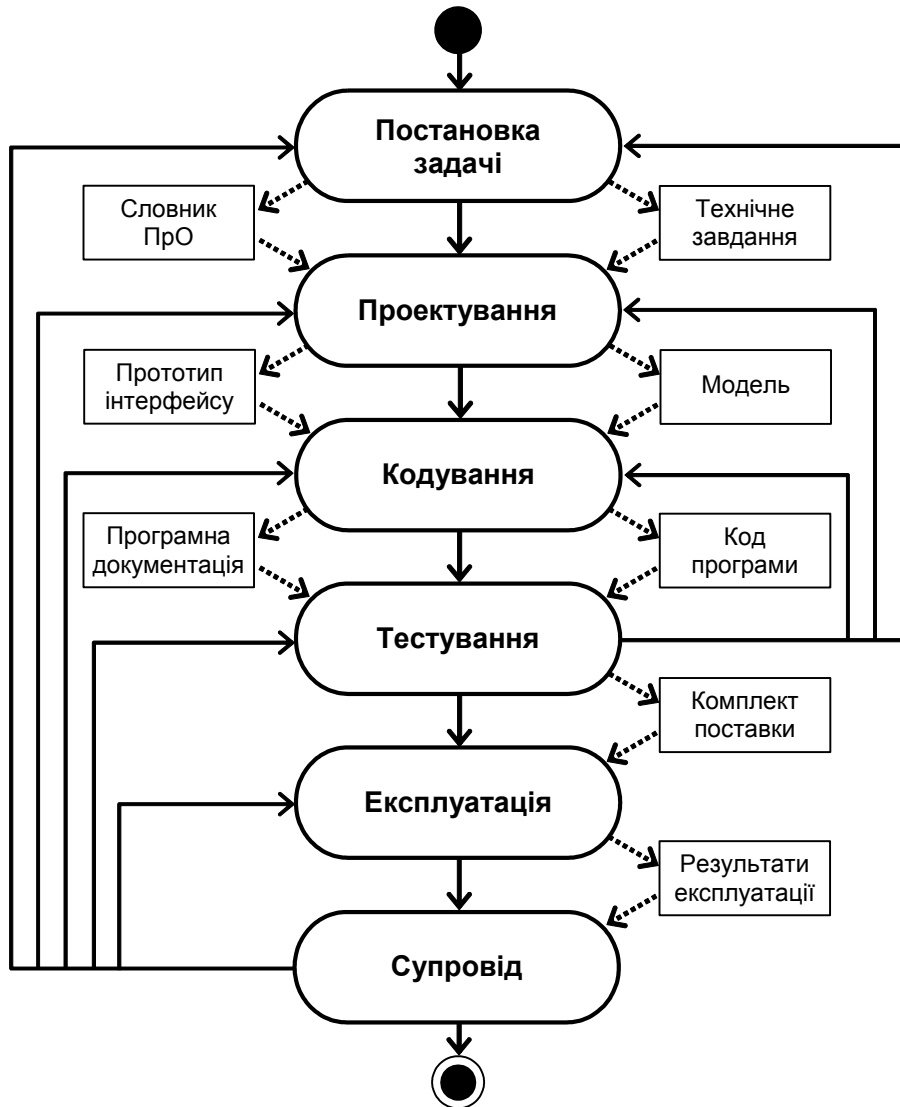


Рис. 1.1. Модель життєвого циклу програмного забезпечення

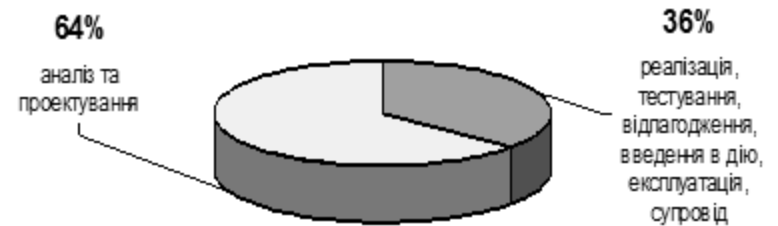


Рис. 1.2. Діаграма статистичного розподілу помилок, які допускаються на різних стадіях ЖЦ ПЗ

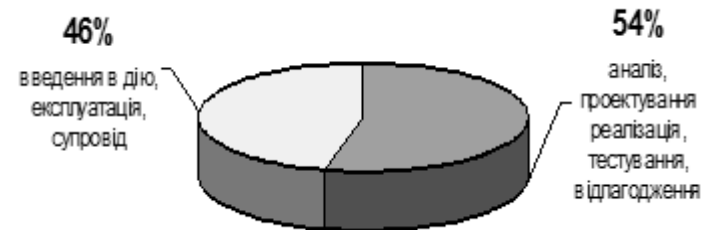


Рис. 1.3. Діаграма статистичного розподілу помилок, які піддаються виявленню на різних стадіях ЖЦ ПЗ

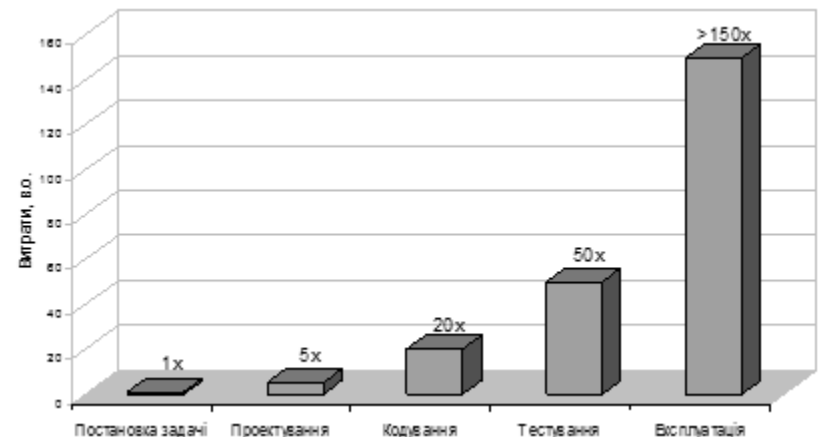


Рис. 1.4. Відносні витрати на виявлення й виправлення помилок в залежності від стадій ЖЦ ПЗ

Представлення проектних рішень на логічному рівні

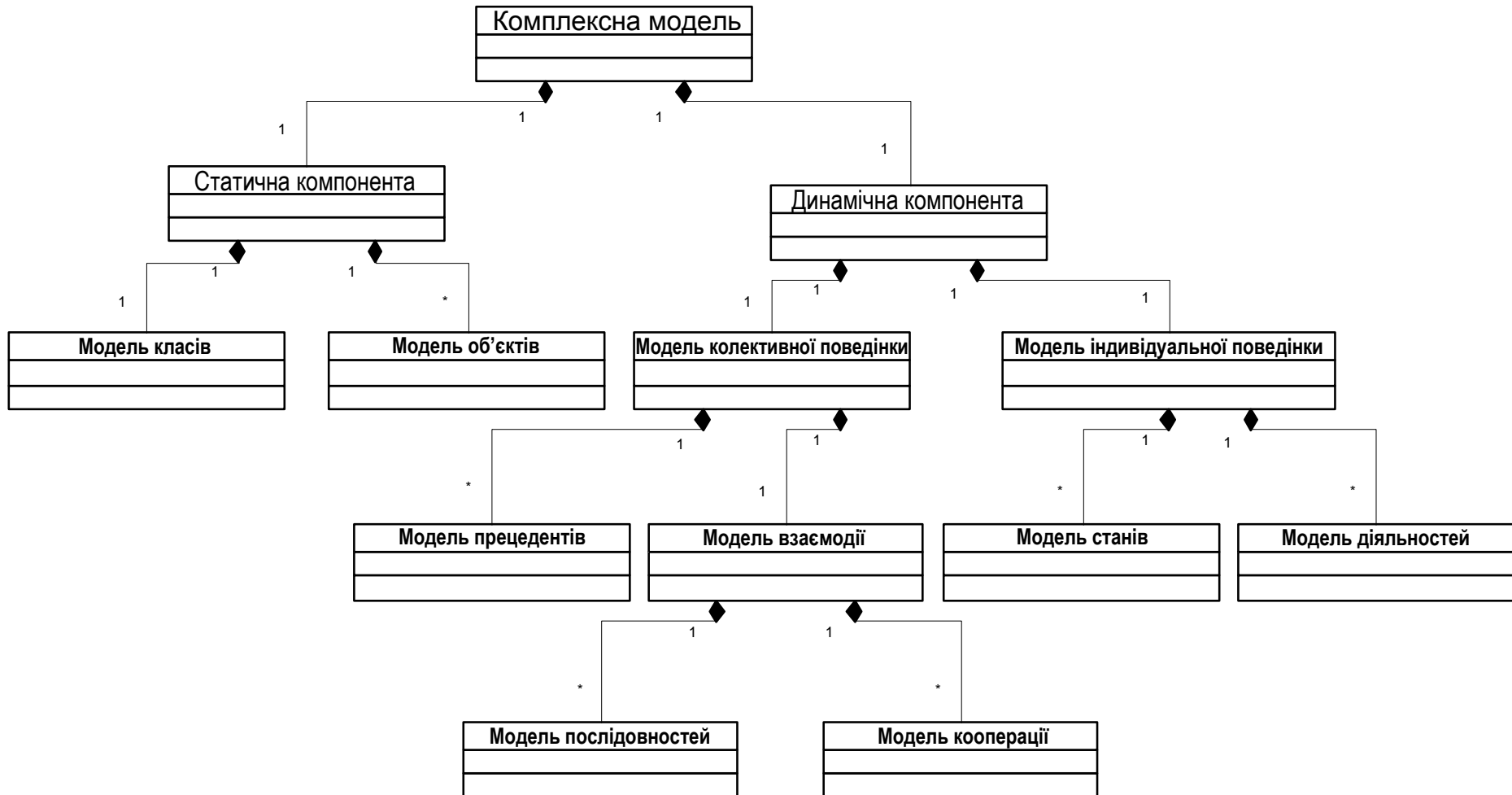


Рис. 2.1. Комплексна модель об'єктно-орієнтованого програмного забезпечення

Вдосконалений метод синтезу моделей станів

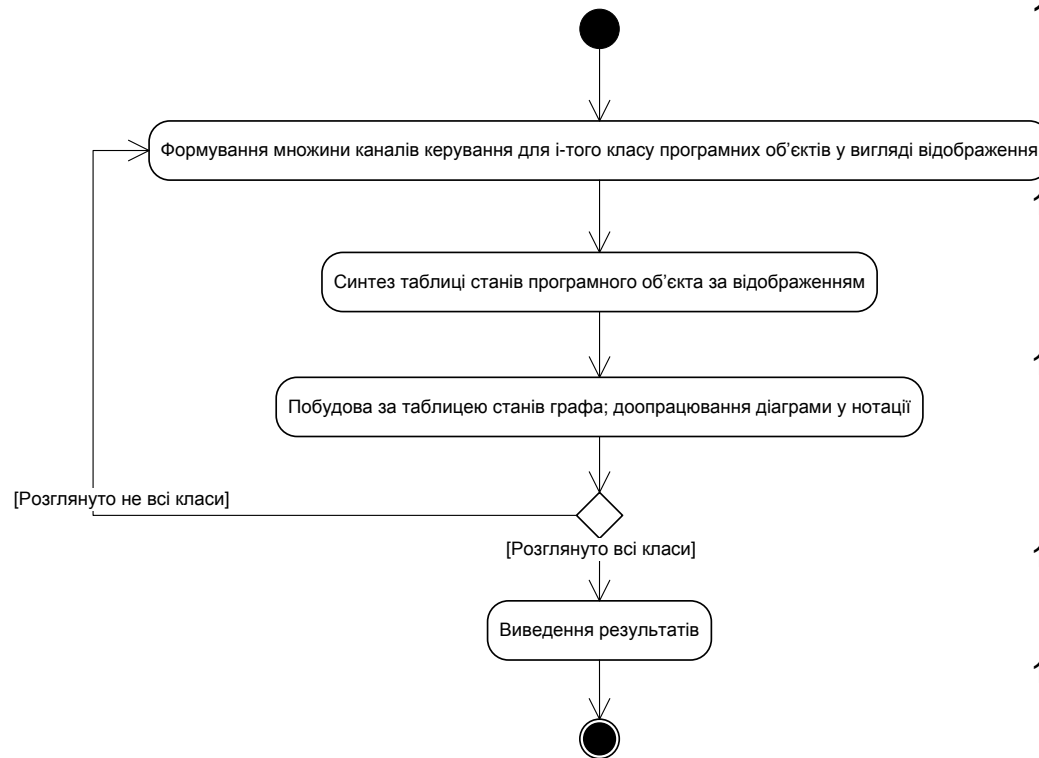


Рис. 3.1. Структура вдосконаленого метода

1. Опис елементів моделі станів екземплярів класів ПЗ у термінах теорії множин:

$$MS_{sc} = (SS_{sc}, SE_{sc}, SH_{sc}, F_{SeSs}).$$

- 1.1. Формалізація структури моделі станів в межах абстрактної теорії скінченних автоматів Мура.
- 1.2. Розроблення моделі послідовностей в межах заданого прецедента:

$$MM_{su} = (SO_{su}, ST_{su}, SL_{su}, SH_{su}, F_{SotSl}).$$

- 1.3. Виконання операції трасування для кожного програмного об'єкта.
- 1.4. Побудова множини каналів керування для поточного класу у вигляді:

$$F_{SteSch} : STE_{sc} \rightarrow CH_{sc}.$$

2. Приведення F_{SteSch} до автоматного виду F_{SteSch}^A , синтез скінченно-автоматної моделі:

$$A_{sc} = (SS_{sc}, SE_{sc}, SH_{sc}, F_{SeSs}, F_{SsSh}, ss_0, SS_{sc}^{sac}).$$

3. Побудова за таблицею станів графа.
4. Виведення результатів.

Метод синтезу тестової структури станів об'єкта ООПЗ АСУ

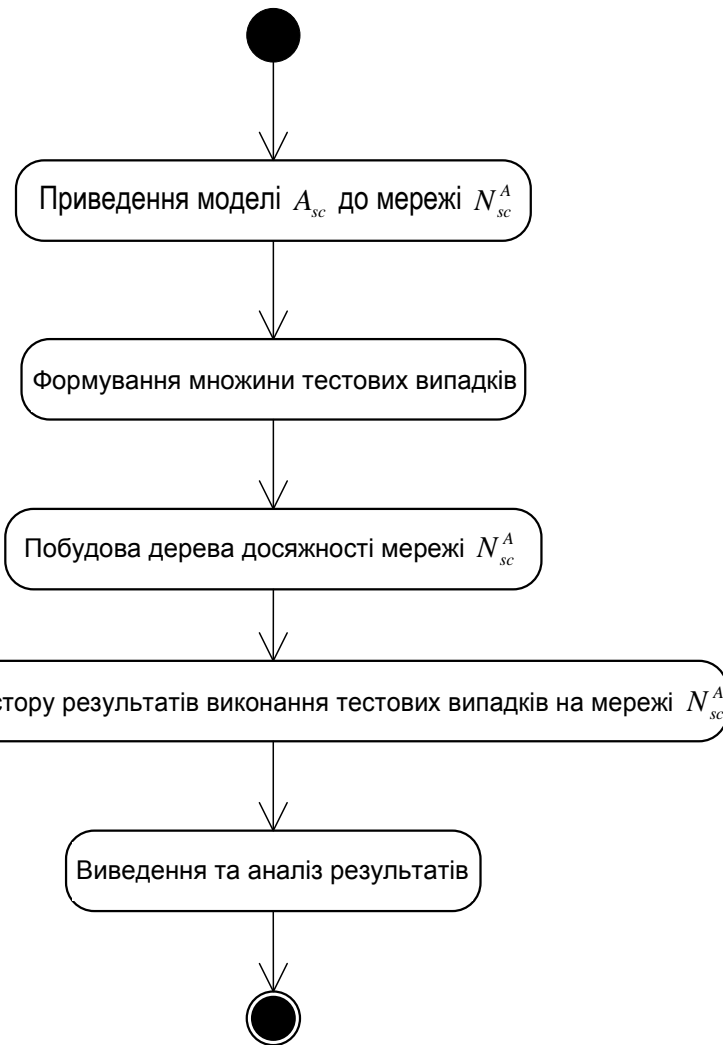


Рис. 4.1. Структура метода

$$A_{sc} = (SS_{sc}, SE_{sc}, SH_{sc}, F_{SeSs}, F_{SsSh}, SS_0, SS_{sc}^{sac}) \rightarrow$$

$$N_{Asc} = (B^A, Z^A, F_{BZ}^A, F_{ZB}^A, M_0^A),$$

$$F_{ZSe}^A : Z^A \rightarrow SE_{sc}, \quad F_{BSs}^A : B^A \rightarrow SS_{sc},$$

$$F_{BSs}^A : B^A \rightarrow SS_{sc}, \quad F_{BZ}^A, F_{ZB}^A \text{ відповідно до } F_{SeSs}.$$

Перетворення МС у тестову структуру:

1. Елемент множини станів $SS_{sc} \rightarrow$ елемент множини позицій B^A ;
2. Елемент множини подій $SE_{sc} \rightarrow$ елемент множини переходів Z^A ;
3. Функція переходів станів $F_{SeSs} \rightarrow$ пряма й зворотна функції інцидентності F_{BZ}^A, F_{ZB}^A .

Новизна метода полягає у:

1. Формальне подання моделі станів об'єктів ПЗ АСУ на основі автоматних мереж Петрі;
2. Формальне представлення процесу тестування моделі станів об'єктів ПЗ АСУ як процес аналізу властивостей живучості (безпеки) та досяжності мережі Петрі.

Метод синтезу тестових структур взаємодії об'єктів ООПЗ АСУ



Рис. 5.1. Структура метода

1. Побудова моделі кооперації:

$$MCO_{su} = (SO_{su}, ST_{su}, SL_{su}, SH_{su}, SR_{su}, F_{SoS1}, F_{SoSr}).$$

2. Синтез Е-мережних тестових структур:

$$F_{ZSoT} : Z^E \rightarrow (CO_{su} \cup ST_{su});$$

$$F_{BSe} : B_Q^E \rightarrow SE_{su};$$

$$F_{bSa} : \{b^j[i]\} \rightarrow F_{Se\sigma(Sa)}(F_{SISe}(sl));$$

$$F_{zSa} : \{z[i]\} \rightarrow F_{Sc\sigma(Sa)}(F_{SoSc}(so));$$

$$F_{\zeta Sh} : \{\zeta(z)\} \rightarrow SH_{su};$$

$$\zeta : \left[\prod_l \rightarrow \left(\{\phi_{li}^b\}^{in} \cup \{\phi_{ls}^z\} \cup \{\phi_{lj}^b\}^{out} \right) \right]$$

3. Опис Е-мережного переходу тестової структури:

$$F_{SesbSs} : SE_{sc} \times SS_{sc} \times SBC_{sc} \rightarrow SS_{sc}; \quad F_{SsSh} : SS_{sc} \rightarrow SH_{su};$$

$$F_{Sh\sigma(Sha)} : SH_{sc} \rightarrow \sigma(SHA_{sc}); \quad F_{SsSha} : SS_{sc} \rightarrow \sigma(SHA^{sd});$$

$$F_{SsSe} : SS_{sc} \rightarrow \sigma(SE).$$

4. Е-мережна тестова структура взаємодії програмних об'єктів: $N_E^{MVO} = (B^E, B_P^E, B_R^E, Z^E, F_{BZ}^E, F_{ZB}^E, M_0^E).$

Метод синтезу тестових структур взаємодії об'єктів ООПЗ АСУ

$$MCO_{su} = (SO_{su}, ST_{su}, SL_{su}, SH_{su}, SR_{su}, F_{SoSI}, F_{SoSr}) \rightarrow N_E^{MVO} = (B^E, B_P^E, B_R^E, Z^E, F_{BZ}^E, F_{ZB}^E, M_0^E)$$

Перетворення моделі взаємодії у тестову структуру:

1. Програмний об'єкт $so \in SO_{su}$, зовнішня сутність $st \in ST_{su} \rightarrow$ Е-мережний перехід $z \in Z^E$;
2. Подія класу $se \in SE_{su} \rightarrow$ позиція-черга Е-мережі $b \in B_Q^E$;
3. Екземпляр події $sl \in SL_{sc} \rightarrow$ мітка в позиції-черзі Е-мережі b ;
4. Атрибути події $se \in SE_{su}, se = F_{sIse}(sl) \rightarrow$ множина описувачів мітки $\{b^j[i]\}$;
5. Атрибути класу $sc \in F_{SoSc}(so), so \in SO_{su} \rightarrow$ множина описувачів переходів $\{z[i]\}$;
6. Функція породження $F_{SoSI} : (SO_{su} \cup ST_{su}) \times SL_{su} \rightarrow \{0,1\} \rightarrow$ зворотна функція інцидентності F_{ZB}^E ;
7. Взаємозв'язки об'єктів $F_{SoSr} : SO_{su} \rightarrow SR_{su} \rightarrow$ пряма та зворотна функції інцидентності F_{BZ}^E, F_{ZB}^E ;
8. Діяльність класу $sh \in SH_{su} \rightarrow$ процедура Е-мережного переходу $\zeta : \left[\Pi_l \rightarrow \left(\{\phi_{li}^b\}^{in} \cup \{\phi_{ls}^z\} \cup \{\phi_{lj}^b\}^{out} \right) \right]$.

Новизна метода полягає у:

1. Формальне представлення тестової моделі взаємодії об'єктів програмного забезпечення АСУ на основі формального апарату Е-мереж, що забезпечує спільне відображення структури керування й потоків даних;
2. Розвиток узагальненого алгоритму функціонування Е-мережного макропереходу, який враховує характеристики об'єкта ООПЗ АСУ в межах відповідної моделі взаємодії програмних об'єктів АСУ.

Інформаційна технологія синтезу тестових моделей поведінки об'єктів ООПЗ АСУ

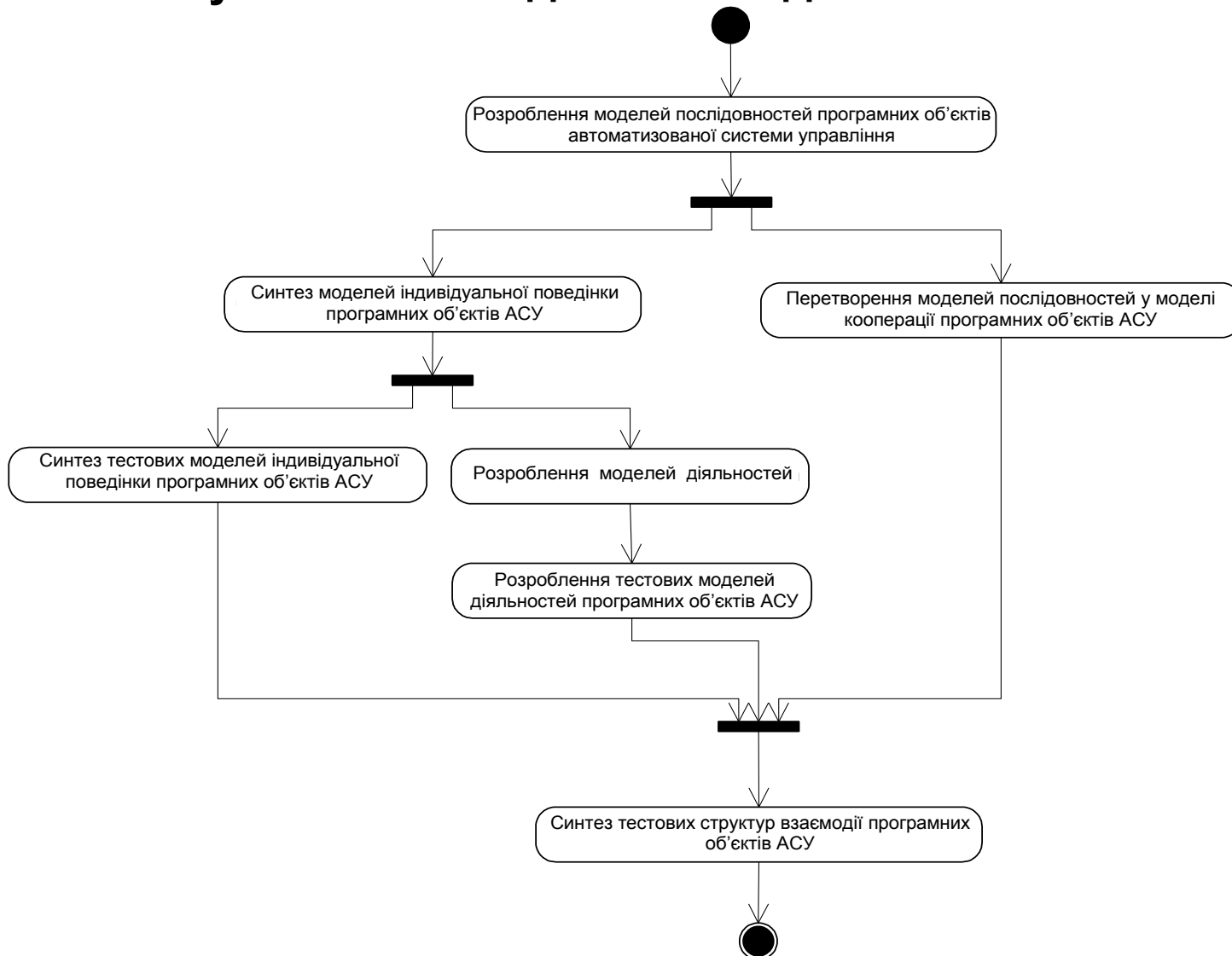


Рис. 7.1. Структура інформаційної технології

Інформаційна технологія синтезу тестових моделей поведінки об'єктів ООПЗ АСУ

Формалізоване представлення комплексної моделі та тестових моделей поведінки об'єктів програмного забезпечення АСУ:

Модель прецедентів:

$$MUC = ((SU \cup ST), SAC_{SU}, SDC_{SU}, SGC_{SU}, F_{SutSadgC}).$$

Модель класів:

$$MC = (SC, SAC_{SC}, SDC_{SC}, SGC_{SC}, F_{ScSadC}).$$

Модель послідовностей:

$$MM_{su} = (SO_{su}, ST_{su}, SL_{su}, SH_{su}, F_{SotSl}).$$

Модель кооперації:

$$MCO_{su} = (SO_{su}, ST_{su}, SL_{su}, SH_{su}, SR_{su}, F_{SotSl}, F_{SoSr}).$$

Модель станів:

$$MS_{sc} = (SS_{sc}, SE_{sc}, SH_{sc}, F_{SesbSs}).$$

Модель діяльностей:

$$MA_{sc}^{sh} = (SHA_{sc}^{sh}, SS_{sc}^{sh}, F_{SesbSs}).$$

Тестова структура індивідуальної поведінки об'єктів ООПЗ АСУ:

$$N_{Asc} = (B^A, Z^A, F_{BZ}^A, F_{ZB}^A, M_0^A).$$

Тестова структура колективної поведінки об'єктів ООПЗ АСУ:

$$N_E^{MVO} = (B^E, B_P^E, B_R^E, Z^E, F_{BZ}^E, F_{ZB}^E, M_0^E).$$

Висновки

- 1. Вдосконалено метод синтезу моделей станів**, у якому на відміну від існуючих вихідна проектна інформація представляється як алфавітне відображення на основі алгебри регулярних подій, а побудова безпосередньо скінченно-автоматної моделі станів здійснюється за допомогою метода абстрактного синтезу скінченного автомата Мура, що в цілому забезпечує формалізацію процесу визначення станів і їх взаємозв'язків у життєвому циклі об'єкта програмного забезпечення на стадії об'єктно-орієнтованого проектування.
- 2. Розроблено метод синтезу тестової структури станів програмного об'єкта АСУ** на стадії проектування, що забезпечує тестування індивідуальної поведінки об'єктів ПЗ АСУ, виявлення логічних помилок відповідної проектної моделі станів шляхом аналізу властивостей живучості (безпечності) й досяжності автоматної мережі Петрі.
- 3. Розроблено метод синтезу тестових структур взаємодії програмних об'єктів** на основі формалізації елементів моделі взаємодії програмних об'єктів АСУ із застосуванням формального подання тестової моделі взаємодії на базі E-мереж, що забезпечує врахування характеристик об'єктів програмного забезпечення АСУ в межах відповідної моделі, а також тестування їх колективної поведінки.
- 4. Розроблено інформаційну технологію синтезу тестових моделей поведінки об'єктів програмного забезпечення АСУ** на стадії об'єктно-орієнтованого проектування, яка забезпечує підвищення значення показника повноти тестування проектних рішень щодо ООПЗ АСУ, ступеня автоматизації процесу проектування й тестування, а також збільшення кількості виявлених помилок на стадії проектування ПЗ АСУ .



Дякую за увагу!