

Висновки

1. Запропонована класифікація задач розміщення об'єктів, що враховує як розміщення об'єктів на абстрактних структурах – графах, площині, поверхні, так і на природних поверхнях.

2. Проаналізовано формальні моделі задач розміщення об'єктів та алгоритми їх розв'язання, що дало змогу обґрунтувати перспективність застосування методів імітаційного моделювання до розв'язання практично значущих задач.

3. Модифіковано математичне формулювання узагальненої моделі розміщення об'єктів та розроблений алгоритм імітації процедури розміщення з урахуванням двох критеріїв якості.

4. Поданий у статті підхід до оптимального розміщення об'єктів за допомогою імітаційного моделювання є основою для подальших досліджень і створення системи підтримки прийняття рішення для задач розміщення.

1. Аристов С. А. Имитационное моделирование экономических систем: учеб. пособие / Аристов С. А. – Екатеринбург: Изд-во Урал. гос. экон. ун-та, 2004. – 123 с. 2. Харари Ф. Теория графов / Харари Ф. — М.: Мир, 1973. – 301 с. 3. Кристофидес Н. Теория графов. Алгоритмический подход / Кристофидес Н. – М.: Мир, 1978. – 432 с. 4. Забудский Г. Г. Алгоритм решения минимаксной задачи размещения объекта на плоскости с запрещенными зонами / Забудский Г. Г. // Автоматика и телемеханика, № 2, 2004. – С.93–100 5. Шеннон Р. Имитационное моделирование систем – искусство и наука / Шеннон Р. – М.: Мир, 1978. – 420 с. 6. Томашевський В. М. Моделювання систем / Томашевський В. М. – К.: Видавнична група ВHV, 2005. – 352 с. 7. Емельянов А. А. Имитационное моделирование экономических процессов: учеб. пособие / А. А. Емельянов, Е. А. Власова, Р. В. Дума. – М.: Финансы и статистика, 2002. – 368 с. 8. Ситник В. Ф. Імітаційне моделювання: навч.-метод. посібник для самост. вивч. дисц. / В. Ф. Ситник, Н. С. Орленко. – К.: КНЕУ, 1999. – 208 с. 9. Кобелев Н. Б. Основы имитационного моделирования сложных экономических систем / Кобелев Н. Б. – М.: Дело, 2003. – 336 с.

УДК 004

Я.П. Кісь, М. Коваленко, І. Яртим

Національний університет “Львівська політехніка”,
кафедра інформаційних систем та мереж

ІНТЕЛЕКТУАЛЬНО-ІНФОРМАЦІЙНА СИСТЕМА РОЗРОБЛЕННЯ НАВЧАЛЬНОГО ПЛАНУ У ВНЗ

© Кісь Я.П., Коваленко М., Яртим І., 2011

Описано методику збору, аналізу і опрацювання інформації на прикладі предметної області ВНЗ. Розроблено схеми функціонування алгоритму та методи опрацювання даних у заданій предметній області.

Ключові слова: аналіз, інформація, алгоритм, методика збору, схеми.

In the article the method of collection, analysis and working of information is described on the example of the high school. The charts of functioning of algorithm and methods of working of information are developed in the set subject domain.

Key words: analys, information, algorithm, methods of collection, charts.

Вступ

Розвиток вищої освіти України передбачає вирішення багатьох економічних і організаційних питань, зокрема оптимізації мережі вищих навчальних закладів (ВНЗ), створення потужних регіональних освітянських комплексів, розроблення та впровадження науково обґрунтованих

методів визначення обсягу і структури державного замовлення на підготовку фахівців, впровадження кредитно-модульної системи організації навчального процесу, модернізацію і розвиток матеріально-технічної бази, укріплення кадрового потенціалу тощо. Але центральною проблемою, від вирішення якої залежить розв'язання багатьох інших завдань, що стоять перед вищою освітою, є фінансування ВНЗ, забезпечення їхньої діяльності за вкрай обмежених державних коштів.

Вищий навчальний заклад сьогодні – це складний навчально-господарський комплекс із численними зовнішніми й внутрішніми зв'язками. Керування ВНЗ, його інформаційними потоками, навчальним процесом, документообігом й іншими процесами являє собою складну систему, дрібні й великі завдання якої тісно пов'язані.

В організаційній системі найбільш трудомісткими є процеси, пов'язані з обробкою інформації (збирання, перетворення, відображення, зберігання, передача та обробка). Прискорити ці процеси й полегшити працю персоналу кафедри допомагає автоматизована система управління. Людино-машинний характер такої системи керування зручний тим, що завдяки їй здійснюється управління навчальним процесом і пов'язаними з ним підрозділами ВНЗ. Програмно-математичні методи та обчислювальна техніка є лише засобами обробки інформації й підготовки відповідних управлінських рішень.

Постановка задачі

Навчальний процес планується з урахуванням науково-педагогічного потенціалу, матеріальної та навчально-методичної бази, сучасних інформаційних технологій. Він орієнтується на формування освіченої, гармонійно розвиненої особистості, здатної до постійного оновлення наукових знань, професійної мобільності та швидкої адаптації до динамічних процесів у освітній та соціально-культурній сферах, галузях техніки і технологій, системах управління й організації праці, умовах ринкової економіки.

Навчальний процес організовується згідно з навчальним планом, який розробляється на весь період реалізації навчання відповідної до галузевих стандартів освітньо-кваліфікаційної програми підготовки і затверджується ректором.

На основі навчального плану приймають робочий навчальний план, який складають деканати факультетів на поточний навчальний рік. Він конкретизує форми навчального процесу, види навчальних занять і засоби поточного і підсумкового контролю. У навчальному плані відображається також обсяг часу, передбачений на самостійну роботу студентів.

Робочий навчальний план приймає Вчена рада університету і затверджує ректор.

Витяги з робочого навчального плану в установлені строки доводяться до кафедр для розподілу і закріплення навчальних дисциплін за викладачами.

Основним документом, у якому відображається навчальна, методична, наукова, організаційна та виховна робота викладача університету протягом навчального року, є його індивідуальний план.

Метою роботи є аналіз засобів формування навчального навантаження ВНЗ, а також розроблення власного засобу для формування навантаження.

Актуальність роботи пояснюється постійним зростанням кількості студентів, а отже, і кількості облікової інформації у начальному закладі. Також з'являється багато нових спеціальностей та напрямів, тому кафедра повинна здійснювати перерозподіл навантаження.

Мета роботи – розробити засоби створення попереднього навантаження з подальшим процесом його корегування у діалоговому режимі.

Основний матеріал

Концептуальна модель системи

Модель системи можна побудувати за допомогою CASE-засобів. Існує два напрями побудови моделі:

1. Моделювання процесів.
2. Моделювання потоків даних.

Для моделювання інформаційних систем взагалі й структурного аналізу зокрема використовують три групи засобів, що відображають

- функції, які система повинна реалізовувати;
- відношення між даними;
- поведінку системи, що залежить від часу.

Серед всієї множини засобів у методологіях структурного аналізу найчастіше застосовують такі:

- DFD (Data Flow Diagrams) – діаграми потоків даних спільно зі словниками даних та специфікаціями або міні-специфікаціями процесів;
- ERD (Entity-Relationship Diagrams) – діаграми “сутність-зв’язок”.

Вигляд контекстної даграми наведено на рис. 1.

Після вивчення і аналізу процесу розроблення навчального плану можна виділити зовнішні сутності, з якими взаємодіє система. Це “Міністерство освіти України”, “Методична рада ВНЗ”, “Навчально-методичне управління ВНЗ” та “кафедра ВНЗ” – вони будуть входами системи. Виходом системи буде результат її роботи, тобто “навчальний план”. Самою системою буде процес “Навчальний план”, який взаємодіє із зовнішніми сутностями.

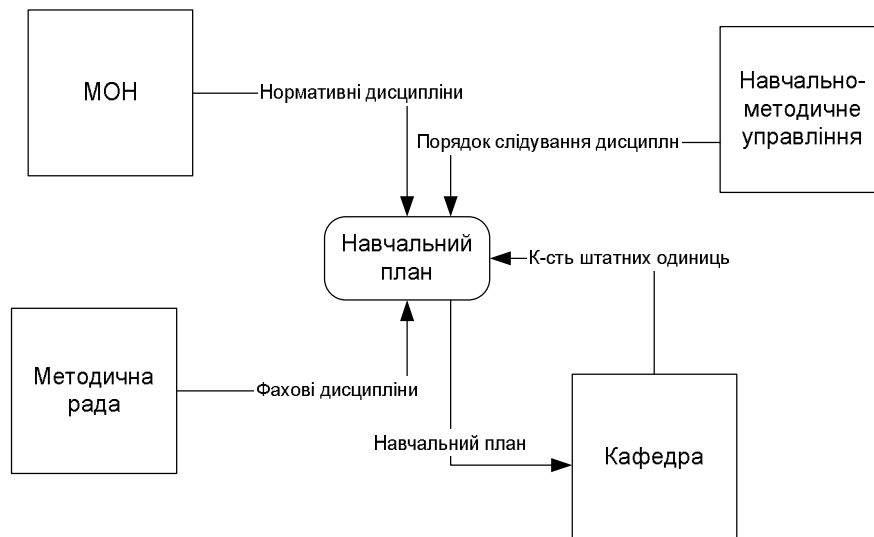


Рис. 1. Контекстна діаграма

DFD першого рівня будується як декомпозиція процесу, що зображений на контекстній діаграмі. Побудована діаграма першого рівня також має множину процесів, що, своєю чергою, можуть декомпонуватись у DFD нижнього рівня. Так будується ієрархія DFD з контекстною діаграмою в корені дерева. Цей процес декомпозиції триває доти, доки процеси не будуть ефективно описані за допомогою коротких специфікацій процесів.

Процес “Навчальний план” ділиться на декілька процесів: “Сформувати вхідні дані”, “Автоматично створити навчальний план”, “Коректувати план” та процес “Затвердити план”. Визначаємо всі інформаційні потоки, що циркулюють в системі, та сховища даних: “Обов’язкові предмети”, “Предмети ВНЗ”, “Предмети кафедри” та “Перелік обмежень”

На рис. 2 подано деталізовану діаграму потоків даних.

DFD забезпечують зручний опис функціонування компонентів системи, але не дають аналітику засобів описання деталей цих компонентів, тобто не визначають, яка інформація перетворюється процесами. Для розв’язання цієї задачі призначені текстові засоби моделювання, які слугують для описання структури інформації, що перетворюється. Це у певний спосіб організований список всіх елементів даних системи з їхніми точними визначеннями, що забезпечить загальне розуміння різноманітними користувачами всіх вхідних та вихідних потоків і компонентів сховищ.

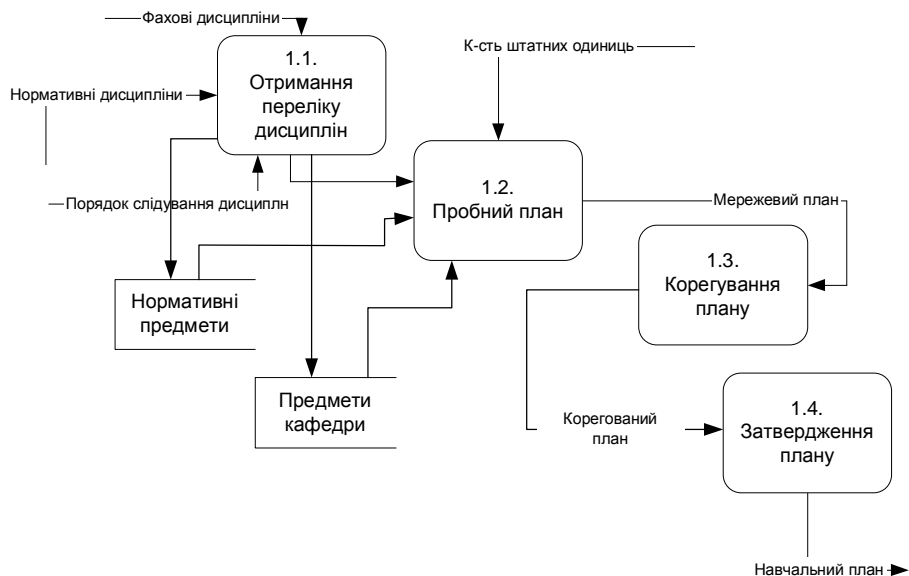


Рис. 2. DFD першого рівня



Рис. 3. DFD другого рівня

Список словників подано на діаграмі потоків даних другого рівня.

Для кожного із сховищ, які містять дані про предмети, що вивчаються на кафедрі, визначаємо таку інформацію:

- Назву предмета.
- Кількість годин, упродовж яких читається предмет.
- Вид контролю.
- Кафедру, яка читає предмет.
- Групи, для яких читають предмет.

Сховище, в якому є дані про порядок слідування, які накладаються на навчальний план, містить таку інформацію:

- Сумарна кількість кредитів протягом всього навчання не повинна перевищувати заданого значення (120 –130 кредитів).
- Сумарне навантаження за семестр не може бути нескінченно великим (якщо прийняти, що є 5 пар на день і на тиждень – 6 навчальних днів, то сумарне навантаження на семестр – 30 кредитів).
- Обмеження на кількість контрольних контролів – це 5 заліків, 5 іспитів, 2 курсові роботи на семестр.
- Обмеження на відношення передування.

ERD подано на рис. 4.

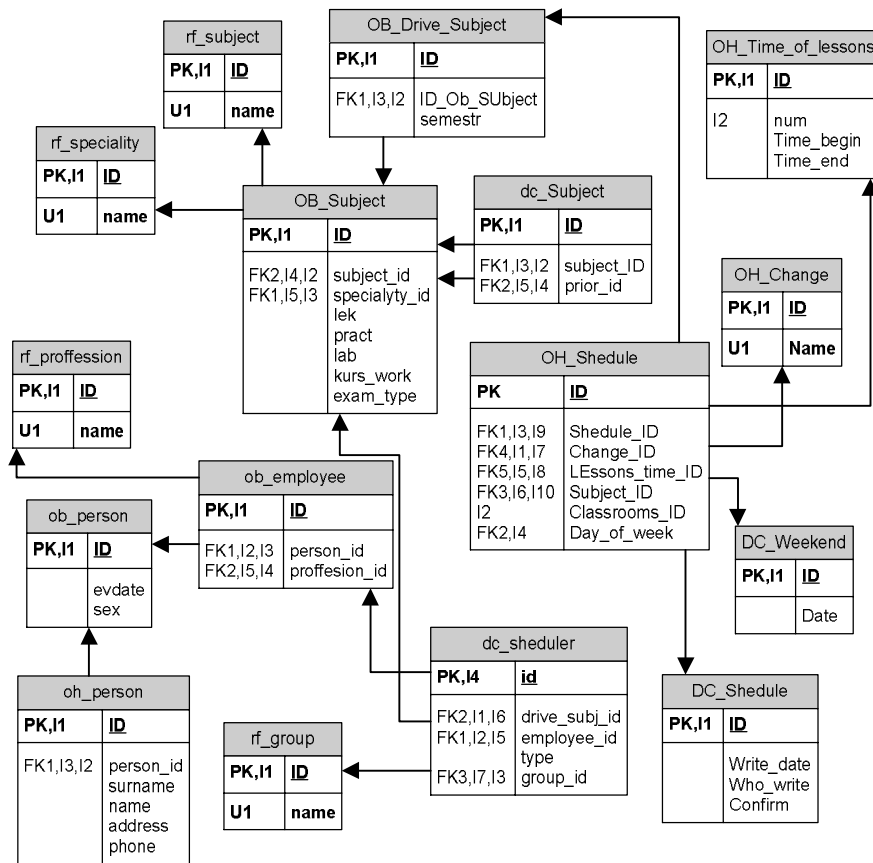


Рис. 4. ERD

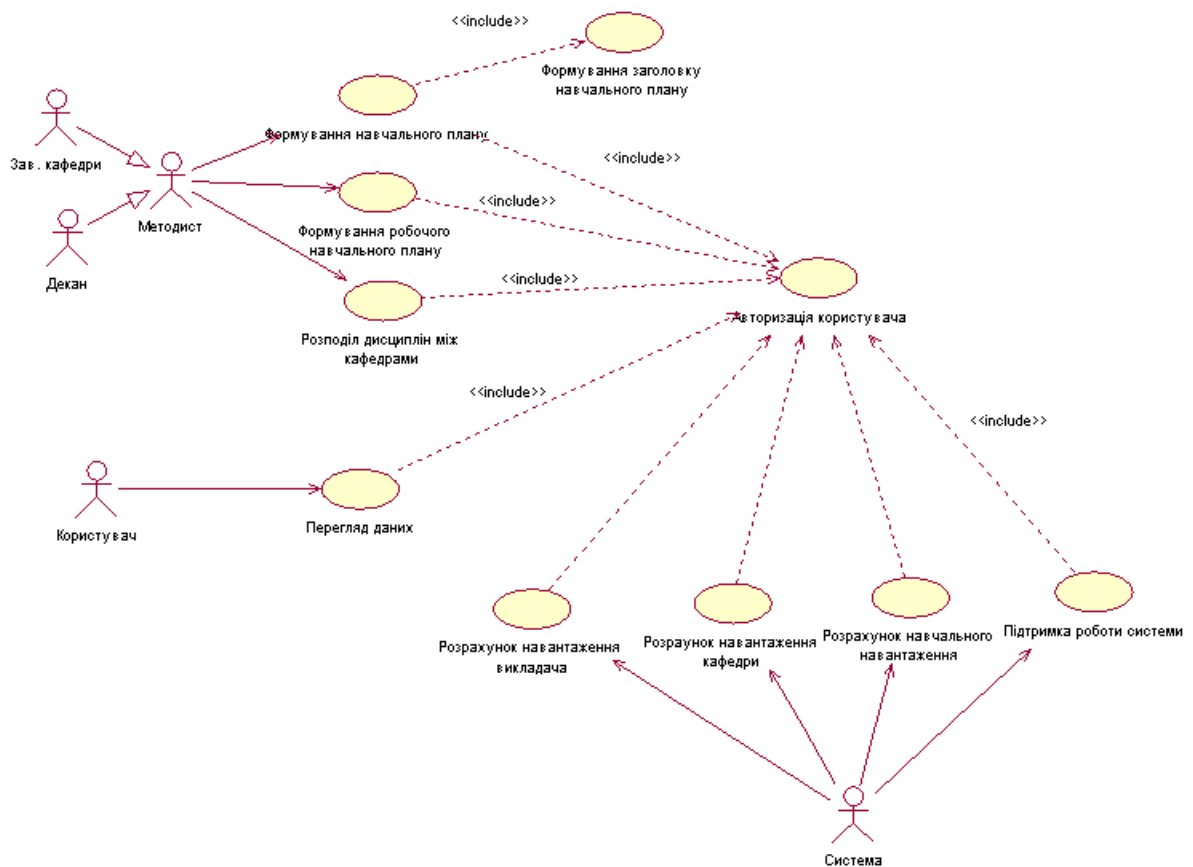


Рис.5. Діаграма прецедентів

На рис. 5 подано головну діаграму прецедентів для проектованої системи. Спільною операцією для всіх акторів є авторизація.

Наступним кроком під час побудови UML моделі є формування потоку подій – послідовності подій, потрібних для забезпечення необхідної поведінки. Цей потік описується в термінах того, “що” система повинна робити, а не “як” вона має це робити. Наведемо приклад потоку подій для прецеденту “Формування навчального плану”. Обов’язковою умовою під час формування потоку подій є визначення передумов, за яких потік може виникнути, умов розгалуження й результатів виконання тих чи інших дій [4].

Потік подій для прецеденту “Формування заголовка навчального плану”

1.1. Передумови

Підпотік створення масивів з різними типами базових освітньо-кваліфікаційних рівнів, кваліфікацією, спеціалізацією, формою навчання, термінами навчання, підпорядкуванням, спеціальністю система має виконати раніше. Методист використовує відповідні дані з цих масивів для формування заголовка навчального плану.

1.2. Головний потік

Прецедент починає виконуватися, коли Методист під’єднується до системи реєстрації і вводить свій пароль. Система перевіряє правильність пароля (E-1) і просить Методиста вибрати потрібний підрозділ (E-2). Система пропонує потрібну операцію – створити новий навчальний план (Add), видалити (Delete), редагувати (Change), надрукувати (Print) чи вийти (Quit). Якщо вибрано операцію додати (Add), S-1: виконується потік – додати навчальний план. Якщо вибрана операція видалити (Delete), S-2: виконується потік – видалити навчальний план. Якщо вибрана операція редагувати (Change), S-3: виконується потік – редагувати навчальний план. Якщо вибрана операція надрукувати (Print), S-4: виконується потік – надрукувати навчальний план. Якщо вибрана операція видалити (Quit), S-5: виконується потік – вийти.

1.3. Підпотоки S-1: додати навчальний план

Система відображає вікно діалогу, яке містить поля для введення підпорядкування, спеціальності, спеціалізації, форми навчання, базового ОКР, терміну навчання й кваліфікації. Методист вибирає відповідні варіанти (E-3). Система створює навчальний план (E-4). Потім прецедент розпочинається спочатку.

S-2: Видалити навчальний план

Система відображає вікно діалогу, яке містить перелік навчальних планів. Методист вибирає відповідний навчальний план (E-5). Система його видаляє (E-6). Потім прецедент розпочинається спочатку.

S-3: Змінити навчальний план

Система відображає вікно діалогу, яке містить поля з переліком навчальних планів. Методист вибирає потрібний навчальний план. Система відкриває меню з параметрами навчального плану. Методист вибирає потрібні характеристики (E-7). Система зберігає внесені зміни. Прецедент розпочинається спочатку.

S-4: Надрукувати навчальний план

Система отримує навчальний план і друкує його (E-8). Прецедент розпочинається спочатку.

1.4. Альтернативні потоки

E-1: введено хибний пароль. Методист повинен повторити введення чи завершити прецедент.

E-2: введено хибний підрозділ. Користувач повинен повторити введення чи завершити прецедент.

E-3: залишилися незаповнені поля чи формат даних не відповідає передбаченому системою. Методист повинен відредагувати дані чи завершити прецедент.

E-4: заголовок навчального плану не може бути відображений. Методисту повідомляють, що ця команда на поточний момент є недоступною. Прецедент розпочинається знову.

E-5: вибраний план неможливо відредагувати. Прецедент розпочинається спочатку.

E-6: система не може видалити відповідний навчальний план. Інформація зберігається. Система може видалити відповідний об’єкт пізніше. Виконання прецеденту продовжується.

Е-7: залишилися незаповнені поля чи формат даних не відповідає передбаченому системою. Методист повинен відредагувати дані чи завершити прецедент.

Е-8: система не може виконати друк даних. Методисту повідомляють, що опція на поточний момент недоступна. Прецедент розпочинається заново. Документи з описом потоку подій формуються і зберігаються окремо від даних програми Rational Rose, але вони пов'язані з прецедентами.

Діаграми діяльності – відображають динаміку проекту, є схемами потоків управління в системі від дії до дії, а також паралельні дії й альтернативні потоки. Діаграми діяльності ілюструють дії, переходи між ними, елементи вибору й лінії синхронізації [3].

На рис. 6 подано діаграму діяльності для дій зі створення навчального плану.

Діаграма містить дії й переходи між ними. Також на діаграмі діяльності можна вказувати умовні переходи, які визначають дії системи під час виконання чи невиконання певної умови. За допомогою секцій, що поділяють діаграму на декілька ділянок, можна встановити, хто відповідає за виконання дій на кожній з них.

Обґрунтування методів реалізації

Для автоматичного складання навчального плану вибрано обмеження на відношення передування предметів та на навантаження в семестрах.

Автоматичне формування навчального плану полягає у побудові мережі слідування предметів. Оскільки фіксовані предмети (визначені обов'язковими) також розташовані у послідовності певного вивчення, то задача побудови графік призначення предметів і буде головною у процесі автоматичного формування навчального плану.

Для реалізації розподілу вибрано алгоритм топологічного сортування, який був модифікований і розширений під задачу [1, 2].

Сортуються елементи, для яких визначений частковий порядок, тобто впорядкування задане не на всіх, а тільки на деяких елементах (відношення передування предметів задані попарно). Задачі, які розв'язуються за допомогою цього методу, можна сформулювати так:

1. У тлумачному словнику слова визначаються за допомогою інших слів, якщо слово v визначене за допомогою іншого слова w ($v \mathbf{P} w$). Топологічне сортування слів у словнику означає розташування їх в такій послідовності, щоб всі слова, що беруть участь у визначенні певного слова, містилися перед ним у словнику.

2. Задачу ділять на ряд підзадач. Виконання одних підзадач повинно передувати виконанню інших підзадач. Топологічне сортування означає виконання підзадач у такій послідовності, щоб перед початком виконання кожної підзадачі всі необхідні для цього підзадачі були виконані.

3. В університетській програмі одні предмети вивчають на основі матеріалу інших, тому деякі курси студенти повинні прослухати раніше від інших. Топологічне сортування означає читання курсів у такій послідовності, щоб жоден курс не читали раніше від того, на матеріалі якого він оснований.

У загальному вигляді частковий порядок на множині S – це відношення між елементами цієї множини і задовольняє три властивості (аксіоми) для будь-яких різних елементів x, y, z із S :

1) якщо $x \mathbf{P} y$ та $y \mathbf{P} z$, то $x \mathbf{P} z$ (транзитивність);

2) якщо $x \mathbf{P} y$, то не $y \mathbf{P} x$ (асиметричність);

3) не $x \mathbf{P} x$ (іррефлексивність).

Вважаємо, що множина S , яку необхідно розсортувати, є скінченною. Тому відношення часткового порядку можна проілюструвати за допомогою діаграми або графу, в якому вершини позначають елементи S , а стрілки зображають відношення порядку. Мета топологічного сортування – перетворити частковий порядок на лінійний.

Щоб знайти одне з можливих лінійних впорядкувань дисциплін, починаємо з того, що вибираємо будь-який елемент, якому не передують ніякий інший (хоча б один такий елемент існує, інакше був би цикл). Цей елемент поміщається на початок списку і виключається зі списку S .

Множина, що залишилась, частково впорядкована; отже, можна ще раз застосувати той самий алгоритм, поки множина не стане порожньою. Впорядковуємо всі предмети, і вони будуть по порядку розташовані у лінійному списку (рис. 7).

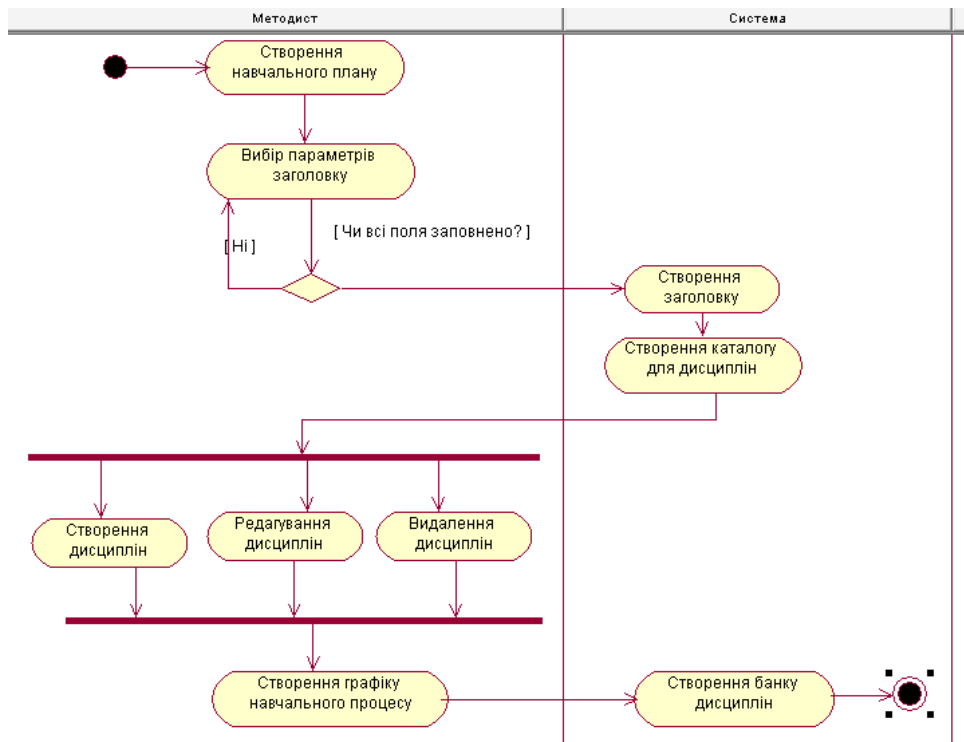


Рис. 6. Діаграма діяльності зі створення навчального плану



Рис. 7. Алгоритм формування мережі слідування

Для реалізації наведеної схеми скористаємося засобами об'єктно-орієнтованого проектування. Для цього дещо модифікуємо традиційну реляційну структуру бази даних. Уведемо відношення слідування предметів dc-Drive_subject, яке дасть змогу будувати мережу:

Dc_drive_subject
Id
Subject_id
Prior_id

У ньому атрибут prior_id вказує на код предмета, який викладається перед предметом subject_id. Зазначимо, що ця структура враховує особливості кредитно-модульної системи, у якій предмет може ділитися на логічні шматки, та вказує порядок їх слідування.

Тоді послідовність побудови мережі предметів складається з таких кроків:

1. Виділення предметів, перед яким не стоїть жоден предмет (prior_id IS NULL).
2. Отримання відношення безпосереднього слідування на множині предметів для кожного семестру з повтором кортежів по предмету: який слідує за множиною дисциплін.

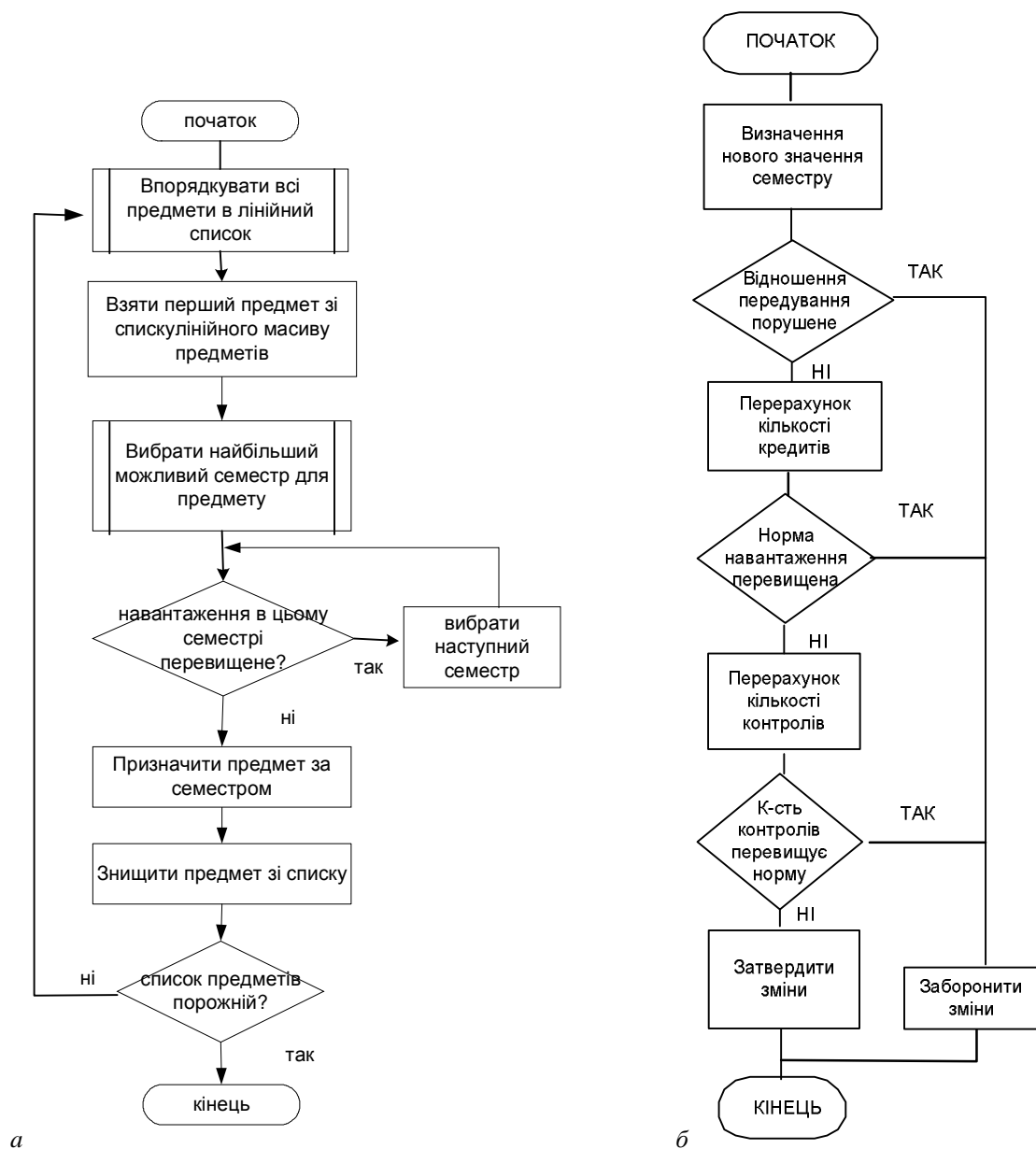


Рис. 8. Алгоритм розподілу предметів (а) та процедури перевірки коректності змін у навчальному плані (б)

3. Визначення сумарного навантаження у кожному семестрі.
4. Якщо сумарне навантаження перевищує максимально дозволене, перенести предмет у наступний предмет. Перейти на крок 3, інакше перейти на крок 5.
5. Групування кортежів отриманого відношення за предметами.
6. Визначення максимального семестру викладання предмета (виходимо з того факту, що предмет можна почати викладати лише тоді, коли завершилися усі предмети, що йому передують).
7. Видалення кортежів у групі, у яких номер семестру менший за максимальний у групі.
8. Формування тимчасового навантаження та виведення його у формі з метою погодження з користувачем.

Схематично це можна зобразити, як на рис. 8, а.

Отже, навчальний план будується в першому наближенні. Звичайно, він не буде оптимальним і не задовольнятиме необхідні вимоги. Для того, щоб навчальний план гарантовано був створений, виконуються лише два найважливіші обмеження: відношень передування та рівномірний розподіл навантаження в семестрах.

Склавши навчальний план у першому наближенні, необхідно його довести у діалоговому режимі до придатного вигляду. Всі зміни і перестановки у навчальному плані контролюють за допомогою відповідних процедур. Для цього впорядкуємо критерії допустимості певної зміни за важливістю:

1. Відношення передування.
2. Рівномірність навантаження у семестрах.
3. Кількість контролів у семестрі.

Отже, у разі будь-якого намагання щось змінити у навчальному плані першим буде перевірятись обмеження на відношення передування (рис. 8, б).

1. Якщо відношення передування не порушується або узалежнені предмети читаються в одному семестрі, то ця зміна допускається.

2. Якщо це обмеження порушується, то зміна не може відбутися ні в якому разі; в цьому випадку програма видає повідомлення про те, що корегування недопустиме. Процедура перевіряє, які предмети передують заданому і які слідують за ним.

3. Після цього для всіх предметів, які залежать від заданого або від яких залежить заданий предмет, знаходять номери семестрів, в яких їх читають.

4. Якщо є декілька предметів, що передують, то серед них є той, номер семестру якого найбільший, тобто який читається найпізніше. Його порівнюють зі зміненим номером семестру і якщо він є меншим (тобто предмет буде читатись раніше, ніж предмет, що передує, то зміна не допускається.

5. Якщо ж для заданого предмета були знайдено ті, які слідують за ним, то серед них знаходять такий, що має найменший номер семестру, тобто який читається найраніше після заданого. Значення семестру заданого предмета порівнюється зі значенням знайденого предмета, і якщо воно є більшим (тобто предмет буде читатись пізніше, ніж ті, яким він передує), то така зміна не буде дозволеною.

6. Після таких перевірок програма видає повідомлення, інформуючи користувача про те, яке саме відношення передування порушене, в якому семестрі читаються предмети, що передують, та наступні, а також яке максимальне і мінімальне значення предмета є допустимим.

У тому випадку, коли відношення передування не порушується, програма перевіряє зміну на наступне обмеження – на кількість кредитів у семестрі. Вона обчислює сумарне навантаження в семестрі, порівнює кількість кредитів з максимально можливою в цьому семестрі і видає повідомлення про те, наскільки воно відхилятиметься від заданого при такій зміні. Зміна не забороняється: користувач сам вирішує, відмовлятися від неї чи ні. Якщо користувач все ж вирішив підтвердити зміну, то відбувається наступна перевірка – на кількість контрольних заходів у семестрі. Обчислюється сума екзаменів, заліків і курсових робіт у семестрі і програма видає повідомлення про їх кількість.

Якщо жодне з перерахованих обмежень не порушується, то для всіх предметів, що слідує за предметом, який намагаються пересунути, номер семестру збільшується на різницю між новим та старим семестром базового предмета. Схема цієї процедури наведена на рис. 8, б.

Висновки

У роботі побудовано систему планування навчального процесу кафедри вищого навчального закладу. Основна увага приділяється формуванню навчального плану, який складається на основі освітньо-професійної програми і містить перелік навчальних предметів, які передбачають вивчати при підготовці фахівців. Навчальний план складається для певного напрямку або спеціальності підготовки фахівців відповідного освітньо-кваліфікаційного рівня (бакалавр, спеціаліст, магістр).

1. Топологічне сортування – [Електронний ресурс]. - [Режим доступу]: http://uk.wikipedia.org/wiki/Топологічне_сортування 2. Ломоносова О.Е. Управління трудомісткістю навчальної роботи у вищих технічних навчальних закладах; [Електронний ресурс]. - [Режим доступу]: <http://www.lib.ua-ru.net/diss/cont/240280.html> 3. Шишко Л.С., Черненко І.Є., Ткачук І.М., Кобицький Є.О. Інформаційні технології в управлінні кафедрою, [Електронний ресурс]. - [Режим доступу]: http://gisau.org.ua/conf2/2/16_Shishko.doc 4. Ломоносов А.В. Автоматизовані розрахунки основних трудових показників, що припадають на окремі спеціальності у вищих навчальних закладах [Електронний ресурс]. - [Режим доступу]: http://www.confcontact.com/2008dec/6_lomonosov.htm 5. Тесля Ю. Нові технології управління у вищих навчальних закладах; [Електронний ресурс]. - [Режим доступу]: <http://www.ualogos.kiev.ua/fulltext.html?id=246>.

УДК 004.052.32

М.Б. Крамаренко
ТзОВ «Елекс»

ДОСЛІДЖЕННЯ СИМЕТРИЧНОЇ МОДЕЛІ ДІАГНОСТУВАННЯ НА СИСТЕМНОМУ РІВНІ

© Крамаренко М.Б., 2011

Систематизовано результати досліджень паралельного, послідовного та локального t -діагностування однієї моделі діагностування на системному рівні, що завжди генерує симетричний синдром. Запропоновано діагностичні структури для паралельного, послідовного та локального t -діагностування. Запропоновано алгоритми дешифрації синдрому мінімальної складності.

Ключові слова: t -діагностування, локальне самодіагностування, моделі діагностики відмов на системному рівні, симетричний синдром.

Parallel, sequential, and local t -diagnosis for a certain system-level diagnosis model which always generates symmetrical syndrome has been researched. Previous researches results were systemized. Minimal complexity algorithms and related diagnosis structures have been suggested.

Key words: t -diagnosis, local self-diagnosis, system-level fault diagnostic models, symmetrical syndrome.

Вступ. Постановка проблеми у загальному вигляді

Моделі діагностики відмов на системному рівні (system-level fault diagnosis [2]) походять з класичної праці [1] й ґрунтуються на поданні багатопроцесорної обчислювальної системи у вигляді певної множини модулів $U=\{u_0, u_1, \dots, u_n\}$, елементи якої визначаються глибиною діагностичних процедур. Вважається, що модулі системи виконують елементарні перевірки (ЕП) один одного. Під