

## АЛГОРИТМІЧНІ АСПЕКТИ АВТОМАТИЗОВАНОГО ПРОЕКТУВАННЯ НАВЧАЛЬНИХ ПЛАНІВ

В статті запропонований підхід до автоматизованого проектування основних документів для планування навчального процесу: освітніх програм, навчальних планів та робочих навчальних планів. Розглянуто методологічні та алгоритмічні аспекти побудови інструментальних комп'ютерних засобів для підтримки планування. Наведені відомості щодо реалізації запропонованих методів та алгоритмів.

This paper proposes an approach to computer-aided design of key documents for the planning of the educational process: educational programs, curriculum and working curricula. The methodological and algorithmic aspects of the development of computer tools for planning support are presented. It gives information about the implementation of the proposed methods and algorithms.

**Ключові слова:** алгоритм, автоматизація, граф, дисципліна, документ, запити, зв'язки, конфлікт, методика, міждисциплінарні, модель, модуль, навчальний, освітня, план, програма, проектування, процес, цикл

### 1. Вступ

Проектування та планування навчального процесу є трудомісткою багатокроковою справою і, загалом, включає наступні стадії: розробка освітнього стандарту, освітньої програми, навчального плану, робочих навчальних планів, складання розкладу занять, розрахунки навантаження викладачів тощо. У даній статті увагу зосереджено на фрагменті ланцюжку розробки: освітня програма – навчальний план – робочі навчальні плани. Розробка таких документів має комбінаторно-ітеративний характер зі зворотними зв'язками; потрібно аналізувати багато варіантів та обирати оптимальний. Як здається, у цьому можуть суттєво допомогти комп'ютерні засоби і технології.

### 2. Аналіз досліджень і публікацій

Розробці та використанню комп'ютерних технологій для підтримки навчального процесу присвячено вже достатньо багато досліджень та публікацій. На Україні створено і відомо декілька діючих автоматизованих систем такого типу, зокрема, АІС "Навчальні плани НГУУ "КПІ" [1], технології фірми ПОЛТЕК СОФТ [2] та інші [3].

В результаті аналізу літературних та рекламних джерел, присвячених системам підтримки автоматизації навчального процесу, можна зробити висновок, що ці системи будуються на платформах систем керування базами даних; і основна функція, яка реалізується у таких системах – зберігання та накопичення навчальних планів [4].

Особлива роль надається робочим навчальним планам, які є основою для розрахунків навантаження, штатів, складання розкладу занять [5].

У керівних документах та науково-методичних публікаціях [6,7], явно чи опосередковано виписана така послідовність: освітня програма (ОП) → навчальний план (НП) → робочий навчальний план (РНП). Загалом, така методологія "від загального до конкретного" – це достатньо популярний підхід, зручний для ієрархій керування, видання наказів. Проте, при складанні пакету планів для нових спеціальностей та спеціалізацій доцільнішою може бути зворотна послідовність: РНП → НП → ОП. Для цього можна навести такі аргументи:

- тільки після проектування на рівні кредитних модулів стають зрозумілими перелік, зв'язки та обсяги дисциплін. Від розуміння послідовності вивчення окремих модулів, їхніх обсягів, узгодженості з іншими спеціальностями та спеціалізаціями, які забезпечує кафедра та факультет, коректно виписуються міждисциплінарні зв'язки, а далі назви, обсяги та семестри викладання для усіх навчальних дисциплін;

- перехід від РНП до НП та ОП комп'ютерна програма може виконати повністю автоматично, що зменшує ризики багатьох людських помилок.

Циклічне, ітеративне повторення операцій проектування (фактично спіральна модель життєвого циклу розробки) дає можливість гарантовано розробити пакет з ОП, НП, РНП, які відповідають вимогам освітнього стандарту, вимогам сучасності та іншим вимогам.

Суттєвим недоліком багатьох існуючих автоматизованих систем є те, що користувачу надаються можливості лише для введення вже готових РНП та НП. Розробник створює документ-план у якомусь середовищі (найчастіше Microsoft Excel), друкує, погоджує паперову копію, хтось вводить з паперу план у базу, потім паперовий документ затверджується. Наявність такої буцімто автоматизованої системи призводить до того, що усі "паперові" служби залишаються, а ще й додатково залучаються значні обсяги людських та матеріальних ресурсів на забезпечення функціонування комп'ютерної системи. Керівництву закладу та розробникам цієї системи користь є, а розробникам РНП, НП (ОП також) майже жодної користі немає – хіба що з'являється додаткова можливість контролю помилок (але при вводі в базу також можливі помилки). Робити вигляд, що автоматизована система, у яку можна тільки ввести (часто вручну) готові РНП та НП, щось автоматизує – для розробників РНП та НП це є фікцією. Проектування РНП та НП є творчим трудомістким ітеративним процесом, який вимагає високої кваліфікації, і може та повинен бути забезпечений спеціальними інструментами. Наприклад, якщо при проектуванні потрібно перенести дисципліну, або її частину, в інший курс (можливо зі змінами обсягів), то потрібен інструмент, який допоможе швидко, зручно і без помилок перебудувати, як мінімум, два РНП з відповідними змінами НП (а, можливо, й ОП).

### 3. Постановка задачі

Навчальні плани містять перелік дисциплін, розподілених по різних семестрам. Для створення коректного плану необхідно загалом дотримуватися багатьох вимог та правил. Виділимо такі:

1. Сума обсягів дисциплін для кожного семестру повинна дорівнювати деяким визначеним кількостям годин (кредитів).

2. Повинна забезпечуватися послідовність викладання-вивчення дисциплін (модулів) згідно мети та основних завдань навчання.

Послідовність дисциплін у часі можна спробувати описати орієнтованим графом. Вершинами будуть дисципліни, а стрілками – зв'язки між ними. Стрілка від вершини А до вершини В означатиме, що дисципліна В базується на деяких знаннях дисципліни А і повинна викладатися пізніше дисципліни А.

Проектувати такий граф (структурно-логічну схему), можна двома способами. Перший спосіб: для кожної дисципліни намагатися уявити, які дисципліни потім можна буде вивчати. Другий спосіб: для кожної дисципліни визначати дисципліни, які необхідно вивчити до того. Другий спосіб на думку автора доцільніший, оскільки набагато простіше шукати відповідь на питання "що необхідно для того, щоб зробити це" замість того щоб фантазувати на тему "де це може стати у нагоді".

Таким чином, для опису графу доцільно використовувати структури даних, які для кожної вершини містять список вершин-попередниць.

Одною з проблем при складанні таких графів є вірогідність циклів. Наприклад, викладач, який розробляє навчальну програму для дисципліни В, вважає, що для вивчення цієї дисципліни потрібні знання, навички, уміння, які є результатами засвоєння деяких інших дисциплін, зокрема, дисципліни А. Інший фахівець, який викладатиме дисципліну А, може вважати, що для вивчення цієї дисципліни необхідно у попередніх семестрах засвоїти декілька дисциплін, зокрема, дисципліну В. Це можна відобразити у вигляді графа зв'язків (рис. 1).

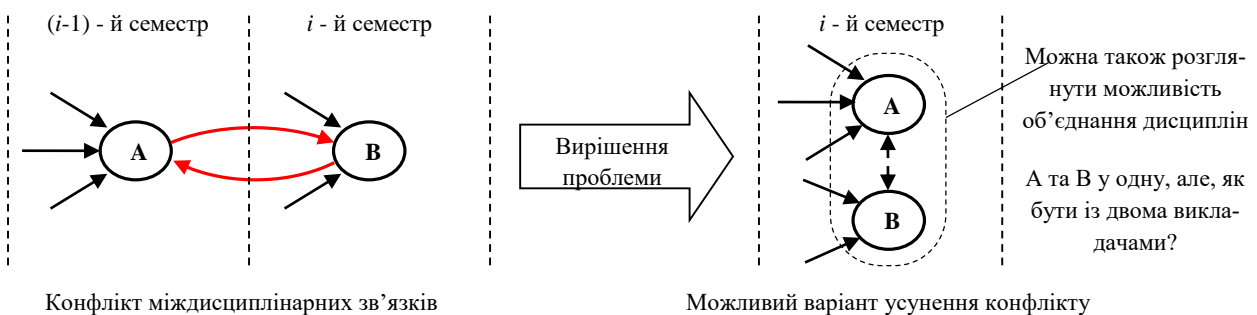


Рис. 1. Приклад простого конфлікту – мінімальний цикл на графі

Складніший варіант конфлікту, який важче помітити людині, яка планує навчальний процес – це замкнений шлях через багато дисциплін, розкиданих по різних семестрах.

Для виявлення та усунення таких конфліктів потрібно уважно аналізувати усі міждисциплінарні зв'язки, які містяться у пропозиціях викладачів і відкидати зайві, оскільки зазвичай викладачі не обмежуються тільки критично необхідними дисциплінами-попередницями, а вказують багато ще інших (трапляється і до десятка) дисциплін, які, на їхню думку, необхідно попередньо засвоїти. Декілька десятків дисциплін (модулів), кожна з яких може бути пов'язана з десятком інших, утворюють громізде павутиння графа міждисциплінарних зв'язків опису проекту навчального (робочого) плану.

Необхідно зазначити, що наявність замкнених шляхів на графі проекту майбутніх НП-РНП зовсім не означає некомпетентність викладачів відповідних дисциплін. Цілком природно, що згодом дійсно може знадобитися знову вивчати якусь дисципліну з подібною назвою – по спіралі, вже на наступному вищому рівні навчання.

А тепер необхідно вказати на те, що деякі дисципліни можуть бути взагалі не пов'язані з іншими. Тоді – це не граф, а множина елементів. Загалом ця множина уявляється як декілька графів для груп зв'язаних елементів та декілька одиночних елементів. Як коректно розподілити у часі таку множину елементів?

У аналізі множин дисциплін, своєчасному виявленні проблем може допомогти комп'ютерна програма, яка втілює відповідні методи і алгоритми. Потрібно розробити такі алгоритми.

#### 4. Виклад основного матеріалу

Для розподілу елементів множини (дисциплін) по стадіям часу (семестрам) можна запропонувати наступний алгоритм.

##### 4.1. Алгоритм впорядкування

Нехай загальна кількість елементів буде  $N$ . Алгоритм складається з двох кроків.

**Крок 1.** Усі елементи розташовуються на першій стадії.

**Крок 2.** Виконується основний цикл впорядкування елементів у часі:

```
for (i=0; i<=N; i++)
```

```
{
  res=MoveToNextStages();
  if (res == 0) break;
}
```

У тілі циклу декілька разів виконується функція

```
int MoveToNextStages()
{
  int done = 0;
  for (i=0; i<N; i++)
  {
    mxstprev = MaxStagePrev(i);
    if (стадія i-го елемента <= mxstprev)
    {
      MoveToStage(i, mxstprev+1);
      done = 1;
    }
  }
  return done;
}
```

Функція **MaxStagePrev(i)** знаходить для  $i$ -го елемента максимальну стадію елемента-попередника. Процедура **MoveToStage(i,s)** розташовує  $i$ -й елемент на стадії  $s$ .

Основний цикл кроку 1 наведеного вище алгоритму виконується максимум  $N+1$  ітерацій. Чому  $N+1$  а не  $N$ ? Це зроблено для виявлення циклів у зв'язках елементів. Якщо елементів усього  $N$ , то їх можна розташувати максимум по  $N$  стадіям часу (це буде суцільний послідовний ланцюжок по одному елементу на кожному стадію). Якщо якийсь елемент займе  $N+1$  стадію, то це вже буде помилкою – розрив у часі. Відповідно цей елемент належить до циклу і його зв'язки необхідно проаналізувати для розв'язання конфлікту. При наявності циклів у зв'язках елементи будуть безкінечно пересуватися на наступні стадії. Тому зроблене обмеження кількості ітерацій – остання  $N+1$ -а ітерація призначена для контролю можливості впорядкування елементів.

Якщо циклів немає, то цей алгоритм може завершитися суттєво раніше до  $N$ -ї ітерації. Це залежить від порядку розташування елементів у робочих масивах.

Складність наведеного вище алгоритму оцінюється як  $O(N^2)$ . Це можна вважати недоліком, проте для навчальних планів з кількістю дисциплін менше сотні, такий алгоритм у програмі для звичайного офісного комп'ютера відпрацьовує за доли секунди. Позитивною рисою алгоритму є поєднання впорядкування множи-

ни елементів з пошуком циклів у графах груп зв'язаних елементів.

### 4.2. Традиційна методика розробки пакету документів

Можна сказати, що традиційна методика складання документів є послідовним ланцюжковим процесом розробки-затвердження документів – документ на основі документа.

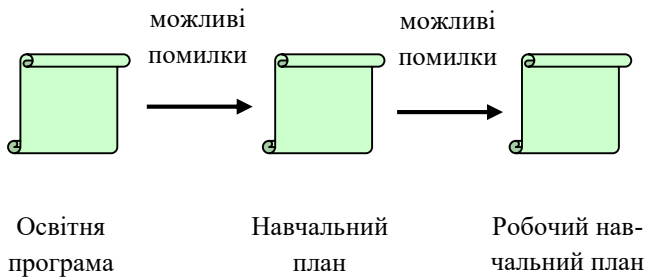


Рис. 3. Традиційне складання документів

Якщо людина бере, наприклад, освітню програму і на її основі складає навчальний план, то може припускатися помилок. На основі документа з помилками, які не помітні, далі може створюватися інший документ, у який можуть додатися нові помилки. Це може призвести до накопичення-розмноження помилок при створенні пакету документів. Досвід показує, що таке трапляється, проте, може бути мінімізовано на рівні виконавців та відповідних процедур контролю.

Набагато гірше, коли проектування пакетів документів пов'язане із частими змінами вихідних даних або(та) вимог. Наприклад, коли вже складені робочі навчальні плани, а потім хтось вимагає змінити назву дисципліни, обсяг, перемістити у інший семестр або курс, або навіть змінити набір варіативних дисциплін у освітній програмі. Це призводить до необхідності змінювати усі документи. З'являється стрес. Різко зростає вірогідність помилок. Розробка пакету документів у визначені терміни стає практично неможливою. Автор статті повною мірою відчув це на собі.

Як запобігти помилок, або хоча б зменшити ризику їхнього виникнення та розмноження? Традиційний шлях – фіксація документоджерела, заборона внесення змін у нього, і тільки після цього дозвіл створювати підпорядковані документи. Іншими словами – каскадна (водоспадна) модель розробки. Проте, така модель

непридатна для розробки чогось суттєво нового – люди не можуть одразу правильно спланувати на майбутнє усе, що треба.

Як створити гнучку технологію, менш чутливу до частих змін вихідних даних та вимог? Вирішити це можна у наступний спосіб.

### 4.3. Методика автоматизованої розробки пакету документів на основі єдиної моделі

Важливе правило: вихідна інформація повинна бути у єдиному екземплярі і вставлятися у багато документів шляхом посилань, а не копіюванням-повторенням. Тоді зміна назви дисципліни у освітній програмі буде автоматично відпрацьована у навчальному та робочому навчальному планах та в усіх інших підпорядкованих документах. Але зміна назви – це тривіальний випадок. А як бути зі змінами обсягів, термінів викладання дисциплін?

Потрібно створити таку модель навчального процесу, яка дозволить автоматично породжувати усі потрібні документи зі змінами. Іншими словами, усі документи повинні з'являтися шляхом відпрацювання запитів до бази даних. Якщо потрібно, вносяться зміни до моделі (редагування бази), а потім одною командою формуються (друкуються) усі потрібні оновлені документи (рис.4).

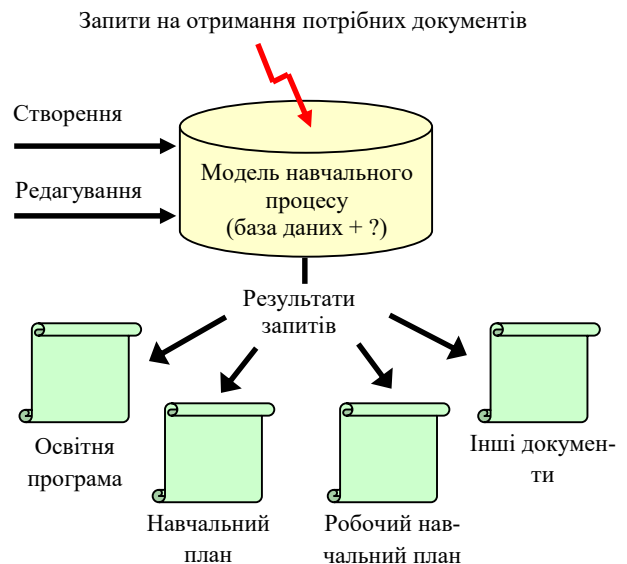


Рис. 4. Документи як результати запитів до бази (моделі)

Якою має бути модель навчального процесу? Це може бути звичайна таблиця дисциплін плюс деяка структура даних, яка описує міждисциплінарні зв'язки. Більше того, окрім відповідних структур даних, модель може містити також і

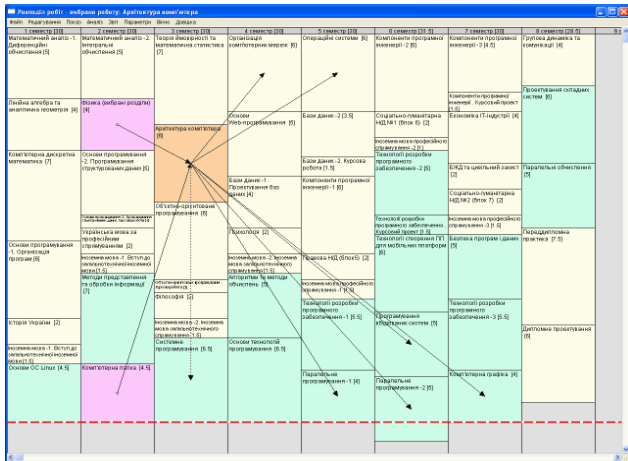
набір процедур – методів обробки внутрішніх даних, які викликаються при редагуванні та запитах. Автор статті реалізував таку модель мовою C++ у відповідному програмному модулі.

### 4.4. Реалізація

На рис. 5 проілюстровано, яка інформація є первинною для формування моделі навчального

процесу. Для кожної дисципліни у першу чергу зберігається обсяг та зв'язки з іншими дисциплінами, які визначають розташування дисципліни на осі часу. Кожну з дисциплін також можна примусово розмістити у часі (відповідному семестрі), якщо це не призведе до протиріччя у міждисциплінарних зв'язках.

Решта інформації зберігається у таблиці атрибутів дисциплін.



№	Назва дисципліни	Семестр	Курс	Тип	Після	Перед	Умови	УРС	Клас	Додаток	Замови	МКТ
1	Математична лінгвістика	6	1	Загальна	Передч. март	150	30	36	60	1	1	Математична
2	Математична лінгвістика	6	2	Загальна	Передч. март	150	30	36	60	2	1	Математична
3	Лінгвістична антропологія та аналітична геометрія	3	1	Загальна	Передч. март	72	36	36	18	3	1	Математична
4	Курси курсу дисципліни математика	6	1	Загальна	Передч. март	210	102	36	180	1	1	Математична
5	Математична лінгвістика	6	1	Загальна	виступає	30	36	36	36	1	1	Математична
6	Математична лінгвістика	6	2	Загальна	виступає	30	36	36	36	2	1	Математична
7	Математична лінгвістика	6	3	Загальна	виступає	30	36	36	36	3	1	Математична
8	Математична лінгвістика	6	4	Загальна	виступає	30	36	36	36	4	1	Математична
9	Математична лінгвістика	6	5	Загальна	виступає	30	36	36	36	5	1	Математична
10	Математична лінгвістика	6	6	Загальна	виступає	30	36	36	36	6	1	Математична
11	Математична лінгвістика	6	7	Загальна	виступає	30	36	36	36	7	1	Математична
12	Математична лінгвістика	6	8	Загальна	виступає	30	36	36	36	8	1	Математична
13	Математична лінгвістика	6	9	Загальна	виступає	30	36	36	36	9	1	Математична
14	Математична лінгвістика	6	10	Загальна	виступає	30	36	36	36	10	1	Математична
15	Математична лінгвістика	6	11	Загальна	виступає	30	36	36	36	11	1	Математична
16	Математична лінгвістика	6	12	Загальна	виступає	30	36	36	36	12	1	Математична
17	Математична лінгвістика	6	13	Загальна	виступає	30	36	36	36	13	1	Математична
18	Математична лінгвістика	6	14	Загальна	виступає	30	36	36	36	14	1	Математична
19	Математична лінгвістика	6	15	Загальна	виступає	30	36	36	36	15	1	Математична
20	Математична лінгвістика	6	16	Загальна	виступає	30	36	36	36	16	1	Математична
21	Математична лінгвістика	6	17	Загальна	виступає	30	36	36	36	17	1	Математична
22	Математична лінгвістика	6	18	Загальна	виступає	30	36	36	36	18	1	Математична
23	Математична лінгвістика	6	19	Загальна	виступає	30	36	36	36	19	1	Математична
24	Математична лінгвістика	6	20	Загальна	виступає	30	36	36	36	20	1	Математична
25	Математична лінгвістика	6	21	Загальна	виступає	30	36	36	36	21	1	Математична
26	Математична лінгвістика	6	22	Загальна	виступає	30	36	36	36	22	1	Математична
27	Математична лінгвістика	6	23	Загальна	виступає	30	36	36	36	23	1	Математична
28	Математична лінгвістика	6	24	Загальна	виступає	30	36	36	36	24	1	Математична
29	Математична лінгвістика	6	25	Загальна	виступає	30	36	36	36	25	1	Математична
30	Математична лінгвістика	6	26	Загальна	виступає	30	36	36	36	26	1	Математична
31	Математична лінгвістика	6	27	Загальна	виступає	30	36	36	36	27	1	Математична
32	Математична лінгвістика	6	28	Загальна	виступає	30	36	36	36	28	1	Математична
33	Математична лінгвістика	6	29	Загальна	виступає	30	36	36	36	29	1	Математична
34	Математична лінгвістика	6	30	Загальна	виступає	30	36	36	36	30	1	Математична
35	Математична лінгвістика	6	31	Загальна	виступає	30	36	36	36	31	1	Математична
36	Математична лінгвістика	6	32	Загальна	виступає	30	36	36	36	32	1	Математична
37	Математична лінгвістика	6	33	Загальна	виступає	30	36	36	36	33	1	Математична
38	Математична лінгвістика	6	34	Загальна	виступає	30	36	36	36	34	1	Математична
39	Математична лінгвістика	6	35	Загальна	виступає	30	36	36	36	35	1	Математична
40	Математична лінгвістика	6	36	Загальна	виступає	30	36	36	36	36	1	Математична
41	Математична лінгвістика	6	37	Загальна	виступає	30	36	36	36	37	1	Математична
42	Математична лінгвістика	6	38	Загальна	виступає	30	36	36	36	38	1	Математична
43	Математична лінгвістика	6	39	Загальна	виступає	30	36	36	36	39	1	Математична
44	Математична лінгвістика	6	40	Загальна	виступає	30	36	36	36	40	1	Математична
45	Математична лінгвістика	6	41	Загальна	виступає	30	36	36	36	41	1	Математична
46	Математична лінгвістика	6	42	Загальна	виступає	30	36	36	36	42	1	Математична
47	Математична лінгвістика	6	43	Загальна	виступає	30	36	36	36	43	1	Математична
48	Математична лінгвістика	6	44	Загальна	виступає	30	36	36	36	44	1	Математична
49	Математична лінгвістика	6	45	Загальна	виступає	30	36	36	36	45	1	Математична
50	Математична лінгвістика	6	46	Загальна	виступає	30	36	36	36	46	1	Математична

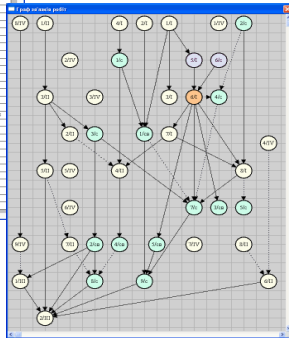
У головному вікні діаграма розподілу дисциплін по семестрам. Виділено одну з дисциплін – показують її зв'язки з іншими дисциплінами

Вікно таблиці атрибутів дисциплін. Це вікно для вводу та редагування текстових та числових атрибутів дисциплін

Рис. 5. Ілюстрація елементів моделі навчального процесу бакалаврів

Вторинною є інформація, яка надається у результаті виконання запитів до моделі.

На рис. 6 проілюстровані деякі різновиди результатів запитів – документи-звіти



Елементи освітньої програми: перелік дисциплін та структурно-логічна схема

Навчальний план

Робочий навчальний план

Рис. 6. Документи – результати запитів до моделі навчального процесу

Комп'ютерна програма, яка усе це виконує, міститься у файлі формату EXE розміром декілька сотень кілобайтів. Програма має декілька рівнів. Внутрішній рівень – ядро, яке реалізує структури даних, що зберігають атрибути кре-

дитних модулів дисциплін, міждисциплінарні зв'язки та розподіл у часі. До базових функцій ядра належать функції-примітиви створення моделі навчального процесу, редагування, контролю цілісності даних, відстеження конфліктів

зв'язків кредитних модулів тощо. На зовнішньому рівні реалізуються операції запитів для формування документів-звітів. Для того, щоб забезпечити максимальну гнучкість, щоб можна було б створити будь-який документ потрібного вигляду, запити оформлюються як виконання (інтерпретація ядром програми) скрипту, який є програмним кодом, записаним у текстовому файлі або модульно у декількох файлах. Автором розроблена проста мова скриптів для запитів. Обробку скриптів виконує вбудований парсер та ядро програми.

## 5. Висновки

Запропонований підхід до автоматизованого проектування освітніх програм, навчальних пла-

нів та робочих навчальних планів, який дозволяє побудувати інструментальні комп'ютерні засоби для підтримки планування навчального процесу, для ефективної розробки пакету документів для різноманітних спеціальностей та спеціалізацій. Висвітлені та проаналізовані деякі проблеми, які постають перед розробниками навчальних планів. Надані методичні рекомендації та алгоритми вирішення таких проблем. Одною з рекомендацій є складання моделі навчального процесу, яка враховує міждисциплінарні зв'язки та атрибути навчальних дисциплін. Така модель дозволяє зручно та ефективно проектувати навчальні плани та освітні програми в умовах часткової первісної невизначеності вихідних даних та змінах вимог.

## Список посилань

1. АІС "Навчальні плани". Конструкторське бюро інформаційних систем НТУУ "КПІ". [Електронний ресурс]. – Режим доступу: [http://kbis.kpi.ua/kbis/index.php?option=com\\_content&task=view&id=84&Itemid=91](http://kbis.kpi.ua/kbis/index.php?option=com_content&task=view&id=84&Itemid=91)
2. ПОЛІТЕК СОФТ. Програмне забезпечення для вищих навчальних закладів України. [Електронний ресурс]. – Режим доступу: <http://www.politek-soft.kiev.ua/>
3. Медовая Е.В. Автоматизация процесса создания учебного плана в ОНАПТ // Автоматизация технологических і бизнес-процесів. – 2014, Volume 6, Issue 4, С. 151-154. [Електронний ресурс]. – Режим доступу: <http://www.journal-atbp.com/journals/atbp20.pdf>
4. В.Г. Гриценко, Г.В. Луценко. UML-моделювання інформаційно-аналітичної системи "Навчальний план" // Інформаційні технології і засоби навчання. 2011. №2 (22). [Електронний ресурс]. – Режим доступу: <http://www.journal.iitta.gov.ua>
5. Мелкумян Е.Ю., Савицький А.Й., Стасюк Н. Розробка автоматизованої інформаційної системи роботи з робочими навчальними планами ВНЗ. Національний технічний університет України „Київський політехнічний інститут”, КБ інформаційних систем. [Електронний ресурс]. – Режим доступу: [http://kbis.kpi.ua/kbis/index.php?option=com\\_content&task=view&id=29&Itemid=46&limit=1&limitstart=2](http://kbis.kpi.ua/kbis/index.php?option=com_content&task=view&id=29&Itemid=46&limit=1&limitstart=2)
6. Про вищу освіту. Закон України. Відомості Верховної Ради, 2014, № 37-38, ст. 2004.
7. Рашкевич Ю.М. Болонський процес та нова парадигма вищої освіти: монографія / Ю. М. Рашкевич. – Львів: Видавництво Львівської політехніки, 2014. – 168 с.