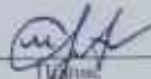



КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему Метод побудови розкладів занять за генетичним алгоритмом

Галузь знань 12 – Інформаційні технології
Шифр і назва галузі знань
Спеціальність 122 – Комп'ютерні науки
Шифр і назва спеціальності
Освітня програма Комп'ютерні науки
Назва освітньої програми

Виконав: студент 2 курсу, група КНМ-20-1  В.В. Богач
Курс, група виконавця Підпис Ініціали, прізвище

Керівник: к.т.н., доцент кафедри КН  О.В. Мазурець
Науковий ступінь, посада Підпис Ініціали, прізвище

Нормоконтроль: к.т.н., доцент кафедри КН  Р.О. Багрий
Науковий ступінь, посада Підпис Ініціали, прізвище

До захисту допускаю:

Зав. кафедри КН, д.т.н., професор  О.В. Бармак
Підпис Ініціали, прізвище

05 грудня 2021 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій

Кафедра комп'ютерних наук

Освітній ступінь магістр

Галузь знань 12 – Інформаційні технології

Спеціальність 122 – Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри комп'ютерних наук


(підпис)

д.т.н., професор О.В. Бармак

« 01 » вересня 2021 року

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА**

1. Тема кваліфікаційної роботи магістра: «Метод побудови розкладів занять за генетичним алгоритмом»

2. Завдання видано студенту Богачу Владиславу Валерійовичу
(прізвище, ім'я, по батькові)

3. Керівник роботи доцент кафедри КН Мазурець Олександр Вікторович
(прізвище, ім'я, по батькові)

4. Затверджені наказом університету від « 25 » серпня 2021 р. № 102

5. Зміст пояснювальної записки (перелік задач) та вихідні дані:

Мета роботи полягає у розробці методу побудови розкладів занять за генетичним алгоритмом, який дозволяє із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вищів за допомогою еволюційного генетичного алгоритму. В якості вхідних зразків розкладів слід передбачити як можливість введення вручну існуючих аналогів, так і можливість автоматизованого створення зразків розкладів. При оцінці сформованих розкладів занять у межах роботи генетичного алгоритму потрібно врахувати пріоритети викладачів щодо бажаної аудиторії проведення пари, днів тижня та власне номерів пар. Також слід врахувати пріоритети академічних груп щодо рекомендованої кількості пар в день і рекомендованої кількості пар в тиждень.

Реферат

Кваліфікаційна робота магістра розв'язує науково-технічну задачу автоматизації побудови розкладів занять за генетичним алгоритмом, який дозволяє із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму. В якості вхідних зразків розкладів слід передбачено як можливість введення вручну існуючих аналогів, так і можливість автоматизованого створення зразків розкладів.

Актуальність теми. Процес набуття знань для різних рівнів освіти називається навчальним процесом, який в свою чергу представлений тісною взаємодією вчителів та учнів. Метою навчального процесу є свідоме здобуття знань для отримання певної кваліфікації. Для того, щоб навчальний процес відбувався безперебійно та за планом, потрібні не лише викладач, учень та дисципліна. Так як навчальний процес частіше всього проходить між декількома групами здобувачів знань та викладачами, то його можна розбити на заняття. Заняття в свою чергу мають наступні компоненти: викладач, студент, предмет заняття, час проведення, день тижня та аудиторія, в якій заняття буде проходити.

Для забезпечення нормального функціонування всіх складових навчального процесу складаються розклади. Особливо важливо це для вищих навчальних закладів, так як групи студентів не мають закріплених за ними аудиторій як, наприклад, в школі. Також свою складність в процес розробки розкладу вносить і тип занять – лекція, лабораторна чи практична робота тощо. Відповідно до кількості студентів та типу занять потрібно підібрати вільну аудиторію та час.

В останні роки значна увага приділяється коректному формуванню розкладу вишів. З сьогоднішніми об'ємами надання вищої освіти розробка таких розкладів є дуже важкою і трудомісткою задачею. Оптимальним розклад є тоді, коли він повністю покриває зв'язок навчальних дисциплін під час процесу навчання, задовольняє певним умови та обмеження, а також враховує досвід викладачів. Якщо при складанні розкладу не враховані всі зв'язки між

дисциплінами, або не враховано всі обмеження, то це впливає на якість навчального процесу, що звісно погіршує рівень підготовки студентів. Тому розклад навчальних занять є документом, що засвідчує трудовий режим викладачів, тому оптимальний розклад здатний позитивно впливає на працездатність як викладачів, так і студентів.

Проте на сьогоднішній день все ще доволі поширена ручна побудова розкладів навчальних дисциплін у вищих навчальних закладах. Хоч вона і займає багато часу та є великий ризик виникнення помилок із-за людського фактора, проте не всі виші можуть дозволити собі використовувати програмне забезпечення, що автоматизовано будує навчальні розклади. Саме тому є актуальним проведення досліджень в цій області та розробка нових методів ефективної автоматизованої побудови розкладів занять для вишів.

Мета і задачі роботи. *Мета* кваліфікаційної роботи магістра полягає у розробці методу побудови розкладів занять за генетичним алгоритмом, який дозволяє із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму. В результаті виконання роботи були поставлені та вирішені наступні *завдання*:

1. Проведено аналіз відомих підходів до автоматизованої побудови розкладів занять й шляхів до їх удосконалення.

2. Вдосконалено інформаційну модель розкладу занять як вичерпного подання предметної області, достатнього для побудови розкладів занять за генетичним алгоритмом.

3. Удосконалено метод побудови розкладів занять за генетичним алгоритмом, який використовує математичну модель обрахунку відповідності розкладу занять вимогам.

4. Розроблено математичну модель обрахунку відповідності розкладу занять вимогам.

5. Розроблено інформаційну систему генерації розкладів занять за генетичним алгоритмом згідно створеного методу.

6. Проведено прикладне та функціональне дослідження методу побудови розкладів занять за генетичним алгоритмом.

Об'єкт дослідження – процес створення розкладів занять для вишів.

Предмет дослідження – інформаційні технології, моделі, методи та засоби автоматизації процесу створення розкладів занять.

Методи дослідження, застосовані для вирішення поставлених завдань: для розв'язання поставлених задач використовуються основні положення методів аналізу даних і теорії множин, а для реалізації інформаційної системи – методології проектування інформаційних систем і об'єктно-орієнтований підхід.

Наукова новизна одержаних результатів. В результаті роботи були отримані інновації й положення наукової новизни:

1. Вдосконалено інформаційну модель розкладу занять, яка відрізняється тим, що містить подання всіх необхідних сутностей для побудови розкладів занять за генетичним алгоритмом, включаючи параметри роботи генетичного алгоритму, обмеження на компоненти розкладу занять та критерії оцінки розкладів.

2. Вдосконалено метод побудови розкладів занять за генетичним алгоритмом, який використовує математичну модель обрахунку відповідності розкладу занять вимогам і відрізняється тим, що дозволяє за наявними в інформаційній моделі розкладу занять параметрами із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму.

3. Розроблено нову інформаційну систему генерації розкладів занять за генетичним алгоритмом, що дозволяє з використанням створених інформаційної моделі розкладу занять, методу побудови розкладів занять за генетичним алгоритмом та математичної моделі обрахунку відповідності розкладу занять вимогам виконувати ручне та автоматизоване створення нових розкладів, формувати нові похідні розклади із використанням генетичного алгоритму, оцінювати сформовані розклади занять із урахуванням вимог викладачів та

академічних груп до параметрів проведення пар й визначати кращі зразки розкладів занять за результатом порівняння їх оцінок відповідності вимогам.

Практичне значення одержаних результатів. Практичне використання розробленої системи підтвердило спроможність інформаційної системи забезпечити виконання функцій побудови розкладів занять за генетичним алгоритмом. Можливості розробленого методу побудови розкладів занять за генетичним алгоритмом і супутніх засобів забезпечують виконання ряду важливих функцій при створенні розкладів занять, що підкреслює практичну значущість одержаних результатів. Зокрема, забезпечується ручне створення розкладів занять, автоматизоване створення розкладів занять, автоматизована оцінка наявних розкладів занять, автоматизоване порівняння наявних розкладів занять, автоматизований пошук шляхів покращення наявних розкладів занять, врахування вимог викладачів до аудиторій, днів тижня та номерів пар у розкладі, врахування вимог підгрупи (групи) до кількостей пар в тиждень та поденно, баланс груп вимог до розкладів викладачів і підгруп, врахування вимог викладачів і підгруп до відсутності вікон у розкладі та врахування нормативних значень по кількості пар у день і тиждень при створенні розкладу занять.

Визначними рисами розробленого методу побудови розкладів занять за генетичним алгоритмом є врахування при оцінюванні зразків розкладів таких критеріїв, як пріоритети викладачів (аудиторії, дні тижня, пари), пріоритети потоків (рекомендовані кількості пар у день і в тиждень), відсутність вікон в викладача і підгрупи, а також врахування обмежень на роботу алгоритму таких як максимальна кількість пар в день і у тиждень у викладача і максимальна кількість пар в день і в тиждень в підгрупі.

Також в роботі розроблено математичну модель обрахунку відповідності розкладу занять вимогам, що дозволяє за наявними в інформаційній моделі розкладу занять параметрами автоматизовано обраховувати умовну оцінку відповідності визначеного зразка розкладу занять вимогам.

Апробація результатів кваліфікаційної роботи магістра та публікації. Основні наукові й практичні результати кваліфікаційної роботи магістра у

доповіді на тему «Метод побудови розкладів занять за генетичним алгоритмом» доповідались на XIII Всеукраїнській науково-практичній конференції «Актуальні проблеми комп'ютерних наук АПКН-2021» (15-16 жовтня 2021 року); за темою роботи автором виконано наукову публікацію:

Богач В. В., Шамрелюк В. В., Шпичко А. В., Мазурець О. В. Метод побудови розкладів занять за генетичним алгоритмом. Збірник наукових праць за матеріалами XIII Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2021». Хмельницький, 2021. с. 291-297.

Структура та обсяг роботи. Кваліфікаційна робота магістра складається із реферату, завдання, змісту, переліку скорочень, вступу, 4 розділів, висновків, переліку посилань із 39 найменувань та 5 додатків. Загальний обсяг кваліфікаційної роботи магістра становить 120 сторінок, з них 100 сторінок основного тексту та 20 сторінок додатків. У роботі наведений 32 рисунки та 29 таблиць.

Ключові слова: генетичний алгоритм, розкладів занять, штучний інтелект, еволюційні алгоритми, інформаційна система, інформаційна модель, інформаційна технологія, викладач, розклад проведення пар.

Зміст

Перелік скорочень	4
Вступ.....	5
Розділ 1	
Аналіз сучасного стану проблеми автоматизації формування розкладів занять вищих навчальних закладів	10
1.1 Аналіз сучасного стану предметної області.....	10
1.2 Аналіз існуючого програмного забезпечення для автоматизованого формування розкладів занять вищів	17
1.3 Аналіз сучасних наукових публікацій з проблеми автоматизації формування розкладів занять.....	24
1.4 Постановка задачі.....	25
Висновки до розділу 1	27
Розділ 2	
Метод побудови розкладів занять за генетичним алгоритмом	29
2.1 Інформаційна модель розкладу занять вищого навчального закладу	29
2.2 Структура вхідних даних та схема методу побудови розкладів занять за генетичним алгоритмом	30
2.3 Математична модель обрахунку відповідності розкладу занять вимогам ..	35
2.4 Функції інформаційної системи для побудови розкладів занять за генетичним алгоритмом	37
Висновки до розділу 2	38

Розділ 3

Інформаційна система генерації розкладів занять за генетичним алгоритмом....	40
3.1 Архітектура інформаційної системи генерації розкладів занять за генетичним алгоритмом	40
3.2 Структура та функціональне призначення складових системи	42
3.3 Аргументація вибору рекомендованих засобів розробки інформаційної системи.....	44
3.4 Проектування структури бази даних інформаційної системи.....	53
Висновки до розділу 3	66

Розділ 4

Дослідження ефективності методу побудови розкладів занять за генетичним алгоритмом.....	68
4.1 Розробка прикладних програмних компонентів інформаційної системи	68
4.2 Тестування інформаційної системи	79
4.3 Функціональне дослідження інформаційної системи генерації розкладів занять за генетичним алгоритмом.....	84
4.4 Прикладне використання інформаційної системи.....	89
Висновки до розділу 4	92
Загальні висновки.....	94
Перелік посилань.....	97
Додатки	

Перелік скорочень

Скорочення, термін, позначення	Пояснення
ІС	Інформаційна система
ІТ	Інформаційні технології
КРМ	Кваліфікаційна робота магістра
КН	Комп'ютерні науки
СКБД	Система керування базами даних
ПП	Програмний продукт
ХНУ	Хмельницький національний університет
ГА	Генетичні алгоритми
ПО	Початкова освіта
ВНЗ	Вищий навчальний заклад
ЗНО	Зовнішнє незалежне оцінювання
ЗУ	Закон України
IDE	Інтегроване середовище розробки
CLR	Common Language Runtime
FCL	Framework Class Library
SQL	Structured query language

Вступ

Кваліфікаційна робота магістра розв'язує науково-технічну задачу автоматизації побудови розкладів занять за генетичним алгоритмом, який дозволяє із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму. В якості вхідних зразків розкладів слід передбачено як можливість введення вручну існуючих аналогів, так і можливість автоматизованого створення зразків розкладів.

Актуальність теми. Процес набуття знань для різних рівнів освіти називається навчальним процесом, який в свою чергу представлений тісною взаємодією вчителів та учнів. Метою навчального процесу є свідоме здобуття знань для отримання певної кваліфікації. Для того, щоб навчальний процес відбувався безперебійно та за планом, потрібні не лише викладач, учень та дисципліна. Так як навчальний процес частіше всього проходить між декількома групами здобувачів знань та викладачами, то його можна розбити на заняття. Заняття в свою чергу мають наступні компоненти: викладач, студент, предмет заняття, час проведення, день тижня та аудиторія, в якій заняття буде проходити.

Для забезпечення нормального функціонування всіх складових навчального процесу складаються розклади. Особливо важливо це для вищих навчальних закладів, так як групи студентів не мають закріплених за ними аудиторій як, наприклад, в школі. Також свою складність в процес розробки розкладу вносить і тип занять – лекція, лабораторна чи практична робота тощо. Відповідно до кількості студентів та типу занять потрібно підібрати вільну аудиторію та час.

В останні роки значна увага приділяється коректному формуванню розкладу вишів. З сьогоднішніми об'ємами надання вищої освіти розробка таких розкладів є дуже важкою і трудомісткою задачею. Оптимальним розклад є тоді, коли він повністю покриває зв'язок навчальних дисциплін під час процесу навчання, задовольняє певним умови та обмеження, а також враховує досвід

викладачів. Якщо при складанні розкладу не враховані всі зв'язки між дисциплінами, або не враховано всі обмеження, то це впливає на якість навчального процесу, що звісно погіршує рівень підготовки студентів. Тому розклад навчальних занять є документом, що засвідчує трудовий режим викладачів, тому оптимальний розклад здатний позитивно впливає на працездатність як викладачів, так і студентів.

Проте на сьогоднішній день все ще доволі поширена ручна побудова розкладів навчальних дисциплін у вищих навчальних закладах. Хоч вона і займає багато часу та є великий ризик виникнення помилок із-за людського фактора, проте не всі виші можуть дозволити собі використовувати програмне забезпечення, що автоматизовано будує навчальні розклади. Саме тому є актуальним проведення досліджень в цій області та розробка нових методів ефективної автоматизованої побудови розкладів занять для вишів.

Мета і задачі роботи. *Мета* кваліфікаційної роботи магістра полягає у розробці методу побудови розкладів занять за генетичним алгоритмом, який дозволяє із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму. В результаті виконання роботи були поставлені та вирішені наступні *завдання*:

1. Проведено аналіз відомих підходів до автоматизованої побудови розкладів занять й шляхів до їх удосконалення.

2. Вдосконалено інформаційну модель розкладу занять як вичерпного подання предметної області, достатнього для побудови розкладів занять за генетичним алгоритмом.

3. Удосконалено метод побудови розкладів занять за генетичним алгоритмом, який використовує математичну модель обрахунку відповідності розкладу занять вимогам.

4. Розроблено математичну модель обрахунку відповідності розкладу занять вимогам.

5. Розроблено інформаційну систему генерації розкладів занять за генетичним алгоритмом згідно створеного методу.

6. Проведено прикладне та функціональне дослідження методу побудови розкладів занять за генетичним алгоритмом.

Об'єкт дослідження – процес створення розкладів занять для вишів.

Предмет дослідження – інформаційні технології, моделі, методи та засоби автоматизації процесу створення розкладів занять.

Методи дослідження, застосовані для вирішення поставлених завдань: для розв'язання поставлених задач використовуються основні положення методів аналізу даних і теорії множин, а для реалізації інформаційної системи – методології проектування інформаційних систем і об'єктно-орієнтований підхід.

Наукова новизна одержаних результатів. В результаті роботи були отримані інновації й положення наукової новизни:

1. Вдосконалено інформаційну модель розкладу занять, яка відрізняється тим, що містить подання всіх необхідних сутностей для побудови розкладів занять за генетичним алгоритмом, включаючи параметри роботи генетичного алгоритму, обмеження на компоненти розкладу занять та критерії оцінки розкладів.

2. Вдосконалено метод побудови розкладів занять за генетичним алгоритмом, який використовує математичну модель обрахунку відповідності розкладу занять вимогам і відрізняється тим, що дозволяє за наявними в інформаційній моделі розкладу занять параметрами із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму.

3. Розроблено нову інформаційну систему генерації розкладів занять за генетичним алгоритмом, що дозволяє з використанням створених інформаційної моделі розкладу занять, методу побудови розкладів занять за генетичним алгоритмом та математичної моделі обрахунку відповідності розкладу занять

вимогам виконувати ручне та автоматизоване створення нових розкладів, формувати нові похідні розклади із використанням генетичного алгоритму, оцінювати сформовані розклади занять із урахуванням вимог викладачів та академічних груп до параметрів проведення пар й визначати кращі зразки розкладів занять за результатом порівняння їх оцінок відповідності вимогам.

Практичне значення одержаних результатів. Практичне використання розробленої системи підтвердило спроможність інформаційної системи забезпечити виконання функцій побудови розкладів занять за генетичним алгоритмом. Можливості розробленого методу побудови розкладів занять за генетичним алгоритмом і супутніх засобів забезпечують виконання ряду важливих функцій при створенні розкладів занять, що підкреслює практичну значущість одержаних результатів. Зокрема, забезпечується ручне створення розкладів занять, автоматизоване створення розкладів занять, автоматизована оцінка наявних розкладів занять, автоматизоване порівняння наявних розкладів занять, автоматизований пошук шляхів покращення наявних розкладів занять, врахування вимог викладачів до аудиторій, днів тижня та номерів пар у розкладі, врахування вимог підгрупи (групи) до кількостей пар в тиждень та поденно, баланс груп вимог до розкладів викладачів і підгруп, врахування вимог викладачів і підгруп до відсутності вікон у розкладі та врахування нормативних значень по кількості пар у день і тиждень при створенні розкладу занять.

Визначними рисами розробленого методу побудови розкладів занять за генетичним алгоритмом є врахування при оцінюванні зразків розкладів таких критеріїв, як пріоритети викладачів (аудиторії, дні тижня, пари), пріоритети потоків (рекомендовані кількості пар у день і в тиждень), відсутність вікон в викладача і підгрупи, а також врахування обмежень на роботу алгоритму таких як максимальна кількість пар в день і у тиждень у викладача і максимальна кількість пар в день і в тиждень в підгрупи.

Також в роботі розроблено математичну модель обрахунку відповідності розкладу занять вимогам, що дозволяє за наявними в інформаційній моделі

розкладу занять параметрами автоматизовано обраховувати умовну оцінку відповідності визначеного зразка розкладу занять вимогам.

Апробація результатів кваліфікаційної роботи магістра та публікації.

Основні наукові й практичні результати кваліфікаційної роботи магістра у доповіді на тему «Метод побудови розкладів занять за генетичним алгоритмом» доповідались на XIII Всеукраїнській науково-практичній конференції «Актуальні проблеми комп'ютерних наук АПКН-2021» (15-16 жовтня 2021 року); за темою роботи автором виконано наукову публікацію.

Богач В. В., Шамрелюк В. В., Шпичко А. В., Мазурець О. В. Метод побудови розкладів занять за генетичним алгоритмом. Збірник наукових праць за матеріалами XIII Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2021». Хмельницький, 2021. с. 291-297.

Структура та обсяг роботи. Кваліфікаційна робота магістра складається із реферату, завдання, змісту, переліку скорочень, вступу, 4 розділів, висновків, переліку посилань із 39 найменувань та 5 додатків. Загальний обсяг кваліфікаційної роботи магістра становить 120 сторінок, з них 100 сторінок основного тексту та 20 сторінок додатків. У роботі наведений 32 рисунки та 29 таблиць.

Розділ 1

Аналіз сучасного стану проблеми автоматизації формування розкладів занять вищих навчальних закладів

1.1 Аналіз сучасного стану предметної області

Система управління якістю освіти у вищому навчальному закладі має забезпечувати контроль показників якості освіти, а також на їх основі проводити підготовку рекомендацій, що покращують всі складові підготовки фахівців, а також підвищують якість освітніх програм. Кожна країна, незалежно від рівня її розвитку просто зобов'язана надати своєму населенню можливість отримати якісну вищу освіту. Вищу освіту надають вищі навчальні заклади, які зацікавлені в тому, щоб їх заклад мав високий престиж. Тому для всіх вищих навчальних закладів та країни в цілому мати привабливу, гнучку та якісну систему освіти [1].

До основних ланок освітньої системи відносяться:

- поточний рівень підготовки абітурієнтів;
- методичне забезпечення процесу навчання;
- рівень кваліфікації професорсько-викладацького складу;
- якість процесу навчання;
- рівень наукових досліджень, що проводяться в університеті;
- рівень забезпечення навчального процесу не обхідним оснащенням.

Важливим чинником, що підтримує зв'язок ланок в тісному зв'язку та дає максимально якісний результат є оптимально складений розклад занять. Освіта є однією з важливих цінностей людини та суспільства, саме вона є передумовою для існування та розвитку люди, а відтак і суспільства в цілому. Чим вищий рівень освіти у людини, тим вищі її прагнення у житті, а також ширші можливості для забезпечення найкращих умов життя та праці. Високий рівень освіти у людини, це не лише її особисте надбання, але і проекція на високий рівень освіти суспільства [2].

Кожна людина, яка є громадянином України має право на освіту згідно конституції України. При чому під поняттям освіти є більш широке визначення, ніж просто здобуття певних навичок у вузькій предметній області. Поняття освіти включає в себе – виховання людини як особистості, набуття навичок спілкування та взаємовідносин з іншими людьми, фізичний розвиток, розвиток людини як майбутнього працівника та важливої ланки в суспільних процесах. Через пізнання своєї історії, ознайомлення з культурною спадщиною, архітектурою, людина долучається до збереження та розвитку системи культурних цінностей. Досягнення пізнавального характеру є сукупністю матеріального і духовного багатства суспільства [3].

Однією з переваг освіти є те, що освітня система вчить отримувати та розвивати критичне та логічне мислення та приймати самостійні рішення. Коли діти стають дорослими, вони стикаються з багатьма непростими питаннями – оплата навчання, створення своєї сім'ї, забезпечення її, сплата комунальних послуг, пошук роботи. Якщо людина витратила роки на навчання, то вона має бути в змозі приймати обґрунтовані рішення щодо цих різноманітних труднощів.

В Україні, відповідно до вікової категорії, існують такі рівні освіти: початкова загальна освіта, базова загальна середня освіта, повна загальна середня освіта, професійно-технічна освіта, неповна вища освіта, базова вища освіта, повна вища освіта.

Початкова освіта (ПО) – це найперший етап загальної освіти. Коли діти отримують початкову освіту, то набувають перші знання про світ, що їх оточує, також отримують важливі навички спілкування та вирішення різного роду життєвих завдань. На даному етапі формується і розвивається особистість, саме це підкреслює важливість початкової освіти для суспільства.

ПО в Україні є обов'язковою, безкоштовною та загальнодоступною. Навчання дітей у початкових школах починається з віку шести років та шести місяців, якщо відсутні протипоказання за станом здоров'я. Проте, навчання у початковій школі не варто відкладати пізніше, ніж досягнення дітьми віку восьми років [4].

Загальна середня освіта – процес оволодіння загальними знаннями про природу, людину, суспільство, культуру. Результатом навчання у середній школі є розвиток особистості в плані інтелекту, соціальності, а також фізичного здоров'я, що забезпечує подальший розвиток працездатності та адаптації в суспільстві [5].

Професійно-технічна освіта – в системі освіти України представляється цілим комплексом педагогічних та управлінських заходів. Ці заходи спрямовані на оволодіння знаннями та навичками в обраній професійній діяльності, а також виховання професіоналізму та загальної професіональної культури, становлення людини як працівника, що має свої права та обов'язки [6].

Вища освіта – набувається у закладах вищої освіти тільки якщо здобувач має повну загальну середню освіту. Вища освіта необхідна працівникам, що мають вищу кваліфікацію в економіці, культурі, науці, тощо. Для того, щоб вступити до ВНЗ необхідно успішно скласти зовнішнє незалежне оцінювання (ЗНО). В процесі послідовного та системного навчання студент отримує необхідні знання та навички для складання державної атестації [7].

Освітній процес (ОП), задля забезпечення його високої якості, регулюється різними законодавчими документами та внутрішнім регламентом кожного учбового закладу незалежно від його рівня.

До таких документів відносяться документи конвенцій та законів України (ЗУ):

- Конвенція про права дитини;
- ЗУ «Про освіту»;
- ЗУ «Про вищу освіту»;
- ЗУ «Про охорону дитинства»;
- ЗУ “Про загальну середню освіту” та інші.

В загальному можна сказати, що процес набуття знань для різних рівнів освіти називається навчальним процесом, який в свою чергу представлений тісною взаємодією вчителів та учнів. Метою навчального процесу є свідоме здобуття знань для отримання певної кваліфікації.

В навчальному процесі важливу роль відіграють всі його компоненти. До компонентів навчального процесу можна віднести: навчальний предмет, який пропонується, процес викладання предмету вчителем, вчителі та учні. Також сюди можна віднести розумові дії учнів, наявність засобів навчання (ЗН) (посібники та підручники, інші методичні матеріали) [8].

Для того, щоб навчальний процес відбувався безперебійно та за планом потрібні не лише викладач, учень та підручник. Так як навчальний процес частіше всього проходить між декількома групами здобувачів знань та викладачами, то його можна розбити на заняття. Заняття в свою чергу мають наступні компоненти: викладач, студент (якщо ми говоримо про виші), предмет заняття, час проведення, день тижня та аудиторія, в якій заняття буде проходити.

Викладач – це людина, в обов'язки якої входить навчання студентів певній дисципліні.

Студент – це людина, що здобуває вищу освіту. Студенти об'єднуються в групи відповідно до спеціальності на яку вони вступили. Група має назву, шифр, що відповідає номеру курсу та назві спеціальності, номеру групи.

Предмет заняття – це дисципліна, що викладається викладачем для студентів, кожна спеціальність має свій набір дисциплін.

Час та день тижня – конкретний фіксований часовий проміжок в який викладач викладає дисципліну студентам.

Аудиторія – місце, в якому проходить навчальний процес. Має свої властивості, наприклад, тип (лекційна аудиторія, лабораторія чи комп'ютерний клас), кількість людей, що може поміститися в середині.

Згідно статті 50 ЗУ «Про вищу освіту» у закладах вищої освіти виділяють наступні форми організації ОП:

- навчальні;
- практична підготовка;
- самостійна робота;
- контрольні.

Також у цій статті НЗ поділяється на наступні види:

- лекція;
- лабораторна робота, практична робота, семінарське заняття та індивідуальне;
- консультація [9].

Для забезпечення нормально функціонування всіх складових навчального процесу складаються розклади. Особливо важливо це для вищих навчальних закладів, так як групи студентів не мають закріплених за ними аудиторій як, наприклад, в школі. Також свою складність в процес розробки розкладу вносить і тип занять – лекція, лабораторна чи практична робота тощо. Відповідно до кількості студентів та типу занять потрібно підібрати вільну аудиторію та час.

Розклад – це план взаємодії компонентів навчального процесу, що чітко розписаний у часі. Розклад можна віднести до документів, що регламентують внутрішній порядок кожного навчального закладу [10].

ЗУ «Про вищу освіту» у статті 10 надає повноваження кожному навчальному закладу на підставі відповідної освітньої програми складати графік навчального процесу (розклад) для забезпечення здобувачами відповідного ступеня освіти [9].

Розклад повинен відповідати враховувати наступні нюанси:

- розклад, що формується повинен створюватися на один тиждень, або на два тижні(якщо розклад суттєво відрізняється по тижням). Можливо також вказувати чи за чисельником чи знаменником на даному тижні відбувається навчання.

- НП відбувається протягом 5 днів (якщо довжина робочого тижня інша, то це також має враховуватись при складанні розкладу);

- НП може відбуватися у декілька змін;
- часовою одиницею при складанні розкладу є урок або пара;
- за певними дисциплінами можуть бути закріплені відповідні аудиторії;
- розклад повинен враховувати допустимі навантаження для здобувачів освіти, а також викладачів;

- за різними викладачами можуть бути закріплені різні типи занять однієї і тієї ж дисципліни;
- необхідно враховувати віддаленість аудиторії, в яких проходять заняття;
- важливо враховувати кількість здобувачів, що може вмістити аудиторія, тощо.

Базовими ж вимогами до складання розкладу є наступні положення:

- жодна група не може знаходитися в один і той же час, в одній аудиторії, з одним викладачем і тим же викладачем, більше, ніж на одному занятті;
- жоден викладач в один і той же час не може проводити будь-який урок більше, ніж в однієї групи, в одному кабінеті (за винятком занять для об'єднаних груп чи потоків);
- в жодній з аудиторій не може проводитися більше одного заняття для однієї групи одним і тим же викладачем (за винятком занять для об'єднаних груп чи потоків).

Як бачимо, задача створення розкладів є доволі складною, адже необхідно враховувати багато важливих моментів, аби це не вплинуло на якісь надання знань.

Ще декілька десятків років тому розклади складалися вручну. Але з часом це стало занадто складно, адже ускладнювався сам освітній процес, освіта ставала все доступніша, що призводило до збільшення потоку охочих на здобуття освіти. Тому для вирішення проблеми складання адекватного розкладу почали залучатися нові технології та здобуття науки.

В загальному вирішенням задачі побудови розкладу сьогоdnішній час займається розділ дискретної математики, який називається «Теорія розкладів». Для цього ставиться задача дискретної оптимізації: потрібно побудувати розклад, який економить час проведення робіт, їх вартість [11].

Проте, ще одним потужним інструментом для складання розкладів генетичний алгоритм.

Генетичний алгоритм (ГА) – це алгоритм пошуку, який застосовується для рішення задач оптимізації і створення моделі шляхом послідовного підбору, та створенні комбінацій параметрів, що шукаються з використанням механізмів, що за виглядом нагадують біологічну еволюцію.

ГА має акцент на використання оператора «схрещення», що виконує функцію рекомбінації рішень-кандидатів, роль якої має аналог – роль схрещення в природі. Книга «Адаптація в природних і штучних системах» Джона Голланда, який вважається засновником генетичних алгоритмів, є фундаментальною в цій сфері [12].

ГА виникли внаслідок спостережень та спроб скопіювати природні процеси, що відбуваються у світі живих організмів, зокрема, еволюцію та пов'язаний з нею природний відбір. Генетичні алгоритми виникли внаслідок спостережень та спроб скопіювати природні процеси, що відбуваються у світі живих організмів, зокрема, еволюцію та пов'язаний з нею відбір (природний відбір) популяції живих істот. Звісно, що за такого порівняння нейронних мереж і генетичних алгоритмів слід звернути увагу на різну тривалість цих природних процесів, тобто, дуже швидко обробку інформації в нервовій системі та дуже повільний процес природної еволюції. Проте, у комп'ютерному моделюванні ці відмінності не є суттєвими [13].

Схема розробки ГА для складання розкладу занять включає декілька кроків. На першому кроці – складається структура та вигляд хромосоми, що буде зберігати зв'язок. Визначена структура повинна включати всі особливості, які вимагаються до розкладу, який потрібно створити. Після цього необхідно створити початкову популяцію такого розміру, який відповідає розмірності задачі. Щоб організувати процес оптимізації, необхідно задати напрямок розвитку популяції. Мінімізація цільової функції може виступати у якості цього напрямку. Також ця функція повинна вміти розрізняти два варіанти створеного розкладу. Відповідно до значень хромосоми далі переходимо до відбору хромосом у батьківський пул та застосування операторів схрещення та мутації. Коли помістимо початкову популяцію у штучне середовище, яке створили, та

реалізуємо процеси селекції, схрещення та мутації, можна прийти до ітераційного алгоритму пошуку оптимального рішення [14].

Отже, як ми бачимо, освіта є одною із головних складових людського життя. При чому, чим вищий рівень освіти людина має, тим більше можливостей перед нею відкривається для досягнення особистих цілей. Освіта – це не тільки здобуття знань в певній області, а й становлення людини як особистості. Також варто зазначити, що рівень освіченості кожної окремої особистості становить в сукупності і рівень освіченості соціуму певної країни.

Для забезпечення високого рівня освіти необхідно правильно організувати НП. Це правило стосується усіх рівнів освіти від початкового до вищого. Навчальний процес повинен включати в себе усі компоненти, а також вони повинні бути у міцному взаємозв'язку один з одним. Саме для забезпечення цього взаємозв'язку навчальні заклади складають розклади. Часто це ще робиться вручну, з тих чи інших причин. Тому питання автоматизованої побудови розкладу занять є відкритим питанням і вартує уваги.

1.2 Аналіз існуючого програмного забезпечення для автоматизованого формування розкладів занять вишів

Завдання створення розкладу є доволі популярною задачею в силу своєї важливості в процесі забезпечення надання ОП. Тому на сьогоднішній день є багато популярних та потужних інструментів для генерації розкладу. Основними системами для автоматичного складання розкладу, на ринку програмного забезпечення України, є:

- комплексна система розкладу університету UniTime;
- програма 1С: Автоматизоване складання розкладів. Університет;
- програма Ректор-ВНЗ;
- програма складання розкладу занять «НІКА».

Комплексна система розкладу університету UniTime – це комплексна система створення навчального розкладу. Вона підтримує розробку розкладів,

керування змінами в цих розкладах, спільне використання аудиторій декількома групами. Ця система є розподіленою, що в свою чергу дозволяє кільком операторам зі складання розкладу університету об'єднувати зусилля для створення та внесення змін до розкладу, який відповідає їхнім різноманітним організаційним потребам, одночасно дозволяючи звести до мінімуму накладки в розкладі.

Ця система розповсюджується безкоштовно за ліцензією з відкритим вихідним кодом [15]. Скріншот програми зображено на рисунку 1.1.

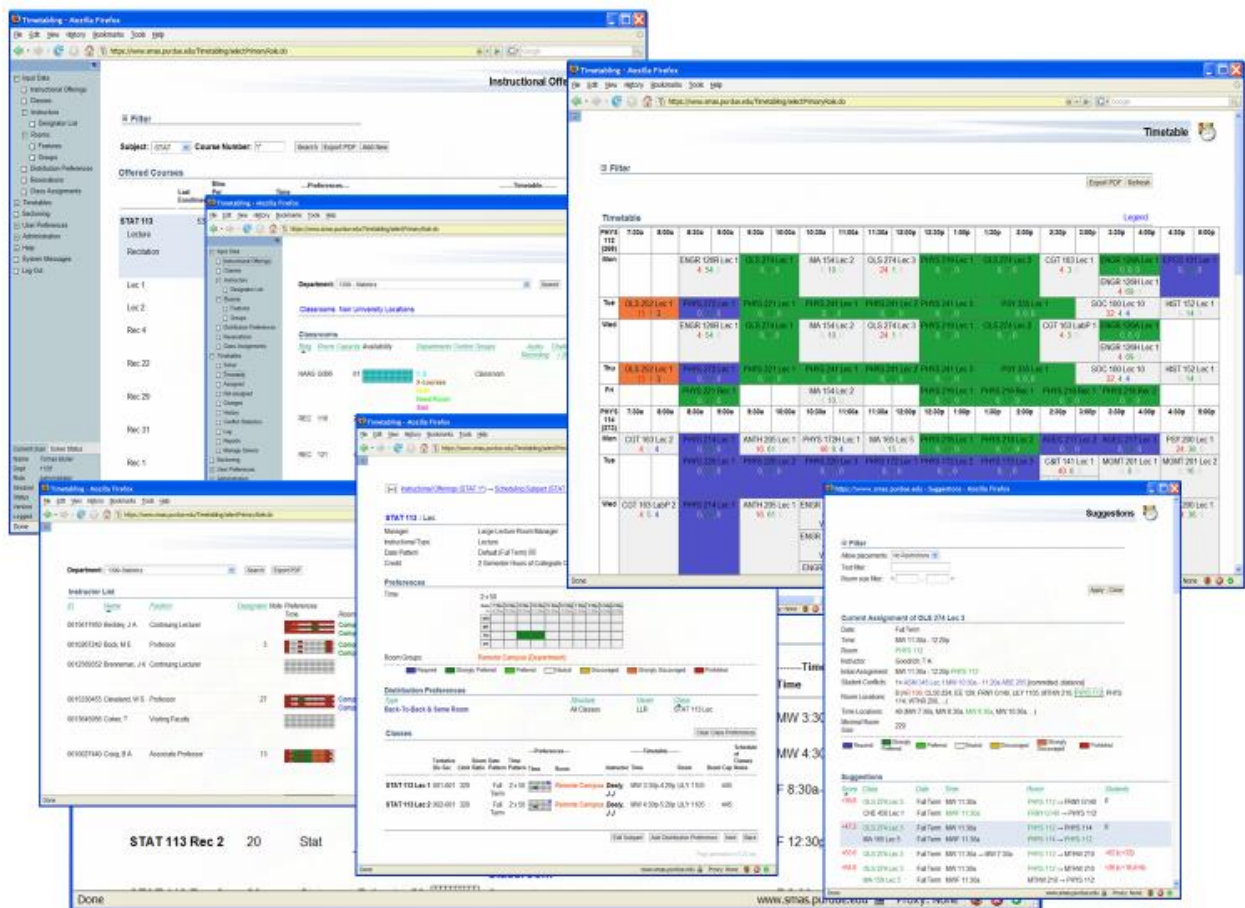


Рисунок 1.1 – Комплексна система розкладу університету UniTime [15]

1С: Автоматизоване складання розкладу. Університет – програма, що призначена для автоматизованого складання розкладу навчальних дисциплін у вищих навчальних закладах, а також для швидкого керування аудиторіями вищих навчальних закладів [16]. За допомогою програми можна скласти

розклад в автоматичному, ручному та змішаному режимах. При цьому є можливість враховувати обмеження, що виникають в процесі планування розкладу. Програма пропонує автоматично оптимізовану версію розкладу з мінімальною кількістю вікон між заняттями. Скріншот вікна програмного продукту у режимі створення розкладу зображено на рисунку 1.2

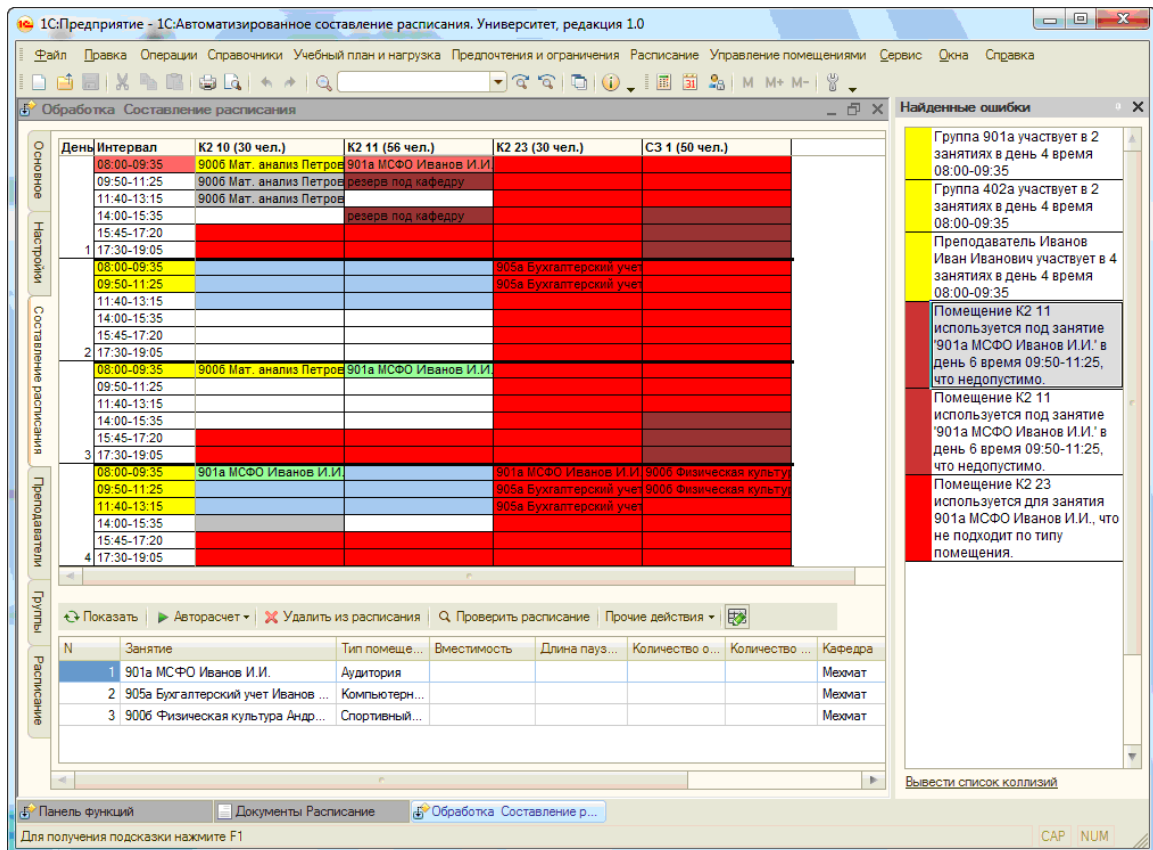


Рисунок 1.2 – Режим створення розкладу в 1С: Автоматизоване складання розкладу. Університет [16]

У програмі реалізований наступний функціонал:

- можливість складання розкладу в трьох режимах;
- інтеграція з системами з «1С: Університет» та «1С: Університет ПРОФ»;
- можливість швидкої ручної модифікації створеного розкладу;
- складання декількох варіантів розкладу та вибір най підходящого з них;

- складання розкладу відносно різних сутностей, таких як група, викладач, аудиторія тощо;
- врахування обмежень, побажань викладачів або студентів;
- об'єднання розкладів.

В об'єднаному розкладі прибираються накладки, що пов'язані, наприклад, зі спільним використанням аудиторій. У розклад заносяться заняття, які проводяться для груп, що належать різним кафедрам, заняття, що мають зарезервовані для певного факультету аудиторії (наприклад, комп'ютерні класи чи класи хімічних лабораторій).

Програма «Ректор ВНЗ» призначена для складання розкладу занять у вузах. Даний програмний продукт має чотири складових: «Списки», «Навантаження», «Розклад», «Заміни», головне вікно розкладу зображено на рисунку 1.3.

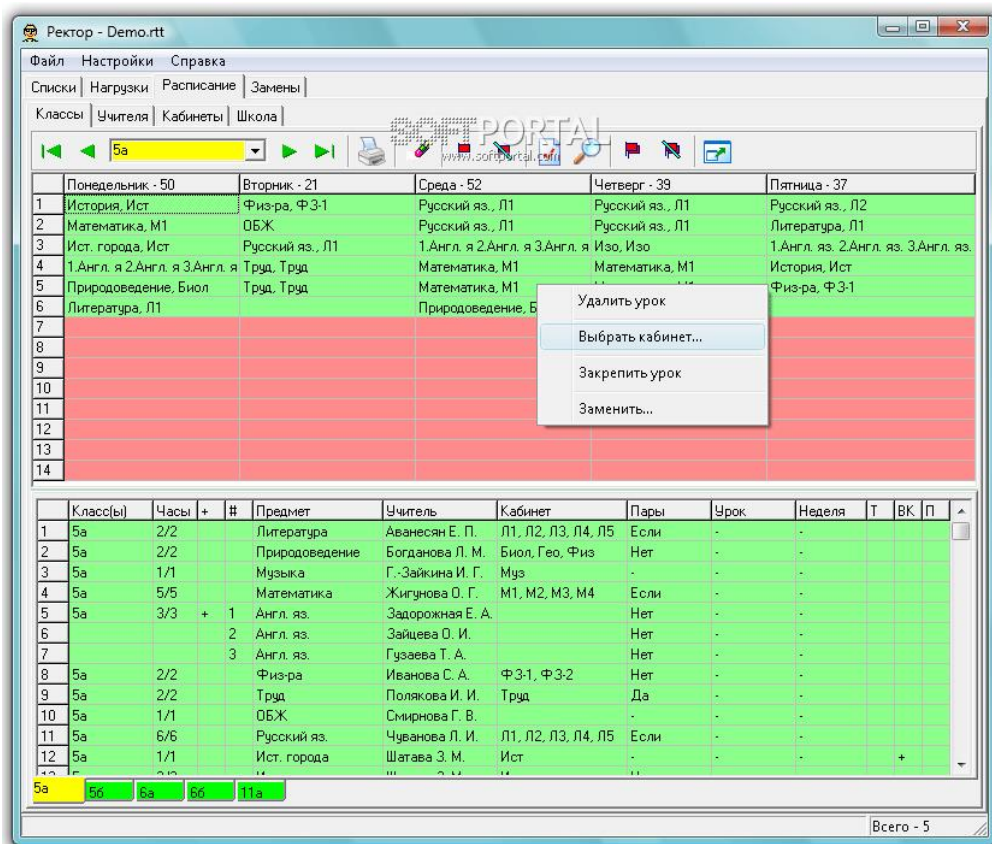


Рисунок 1.3 – Режим створення розкладу в програмі «Ректор ВНЗ» [17]

Розділ «Списки» реалізовує функції для заповнення, редагування і друку різних списків (наприклад, груп, занять, аудиторій, викладачів). Для роботи з навчальними планами з спеціальностями, навантаженнями викладачів, різноманітних звітів в цілому використовується розділ «Навантаження». «Розклад» реалізовує функціональні можливості складання розкладу відповідно до груп, викладачів та аудиторій. А у розділі «Заміни» вносяться дані щодо замін викладачів.

Можна в будь-який момент переключатись між трьома режимами створення розкладу – ручному, автоматизованому та змішаному. У автоматичному режимі створення враховуються усі вимоги щодо створеного розкладу, а при складанні в ручному режимі програма пропонує варіанти розміщення занять для відповідних груп у певні аудиторії та слідку, щоб не було великих проміжків між заняттями.

Розклад є можливість зберегти у декількох форматах, що зручні для використання: Microsoft Word, Excel або HTML [17].

Програма складання розкладу занять «НІКА» повністю автоматизує процес складання розкладу занять в освітніх закладах різного статусу та профілю (школи, гімназії, коледжі, училища), а також враховує особливості навчальних планів та організації навчального процесу. Програма легка у розумінні та зручна у роботі. Результатом роботи програми є розклад занять, складений з мінімальною кількістю вікон для викладачів. Програма «НІКА» враховує:

- наявність другої зміни, п'яти-або шестиденної форм навчання;
- особливості наповнення аудиторії;
- методичні дні та години, що не бажані для роботи вчителів;
- граничні рівні навчального навантаження з урахуванням складності предметів [18]. Скріншот програми зображений на рисунку 1.4.

Ф.И.О.	Предмет	5а	5в	5г	6а	6г	7а	7г	8в	8г	9в	9г	11а	Всего	Методич. и часы
Абакумова	Алгебра										4	4	4	12/20	
Барклай	Алгебра							4						4/6	
Воробьева	Алгебра												4	4/10	Пт,Сб
Пронина	Алгебра							4						4/7	Ср,Сб
Трусенев	Алгебра								4					4/6	
Ибралимова	Биология	2	2	2		2			2	2			2	14/14	
Титова	Биология				2						2	2		8/9	Ср,Сб
Соснина	Биология				2									2/2	
Труфанова	География				2	2	2	2	2	2	2	2		16/16	
Абакумова	Геометрия								2	2	2			6/20	
Барклай	Геометрия						2							2/6	
Воробьева	Геометрия											2		2/10	Пт,Сб
Пронина	Геометрия							2						2/7	Ср,Сб
Трусенев	Геометрия								2					2/6	
Тихонова	ИЗО	1	1	1	1	1	1	1						7/20	Ср
Иванова	Ин.яз.	3	3	3	3	3	3	3	3	3	3	3		21/21	
Курская	Ин.яз.	3	3	3	3	3	3	3	3	3	3	3		24/24	
Товиева	Ин.яз.	3	3	3	3	3	3	3	3	3	3	3		24/24	
Устинова	Ин.яз.		3											3/3	Сб
Абакумова	Информатика											2/1		2/20	
Скворцова	Информатика	1	1	2/1	1	2/1	1	2/1					2/1	10/10	Чт,Сб
Рам	Информатика	1	1	2/1	1	2/1	1	2/1	1	2/1	1	2/1	2/1	12/12	
Тушина	Информатика					2/1	2/1	1	2/1	1	2/1	2/1	2/1	12/12	
Алтуфьева	История										2	2		4/4	
Вьрина							2	2	2	2			3	9/9	
Ев лампиева							2	2	2	2				10/10	

Рисунок 1.4 – Створення розкладу в програмі автоматизованого складання розкладу занять «НИКА» [18]

Розглянуті програмні продукти є доволі потужними інструментами в генерації оптимального розкладу занять у вузах. Недоліком є те, що вони не є безкоштовними. Якщо ж придбати ліцензію, то можливо не весь функціонал потрібен для конкретного вищого навчального закладу за повну вартість ліцензії.

Програма «АВТОРозклад» працює з розкладами будь-якого ступеня складності. Розмірність навчального закладу не обмежена. Можна складати тижневі, двотижневі, семестрові розклади, а також розклади сесій заочників з плаваючим графіком навчання та розкладу разових заходів [19]. Є 8 версій даного програмного продукту. В залежності від типу навчального закладу та потреб користувач може обрати для себе потрібну версію.

«АВТОРозклад» дозволяє максимально полегшити і автоматизувати складну роботу укладачів розкладу. Система допомагає легко будувати, змінювати і роздруковувати у вигляді зручних та наочних документів:

- розклад занять класів (навчальних груп);
- розклади викладачів;
- розклад зайнятості аудиторій (кабінетів);
- навчальні навантаження.

Має дружній до користувача інтерфейс та проста у використанні. Має довідку, в якій користувач може знайти відповіді на запитання щодо процесу створення розкладу. Скріншот програми «АВТОРозклад» зображено на рисунку 1.5.

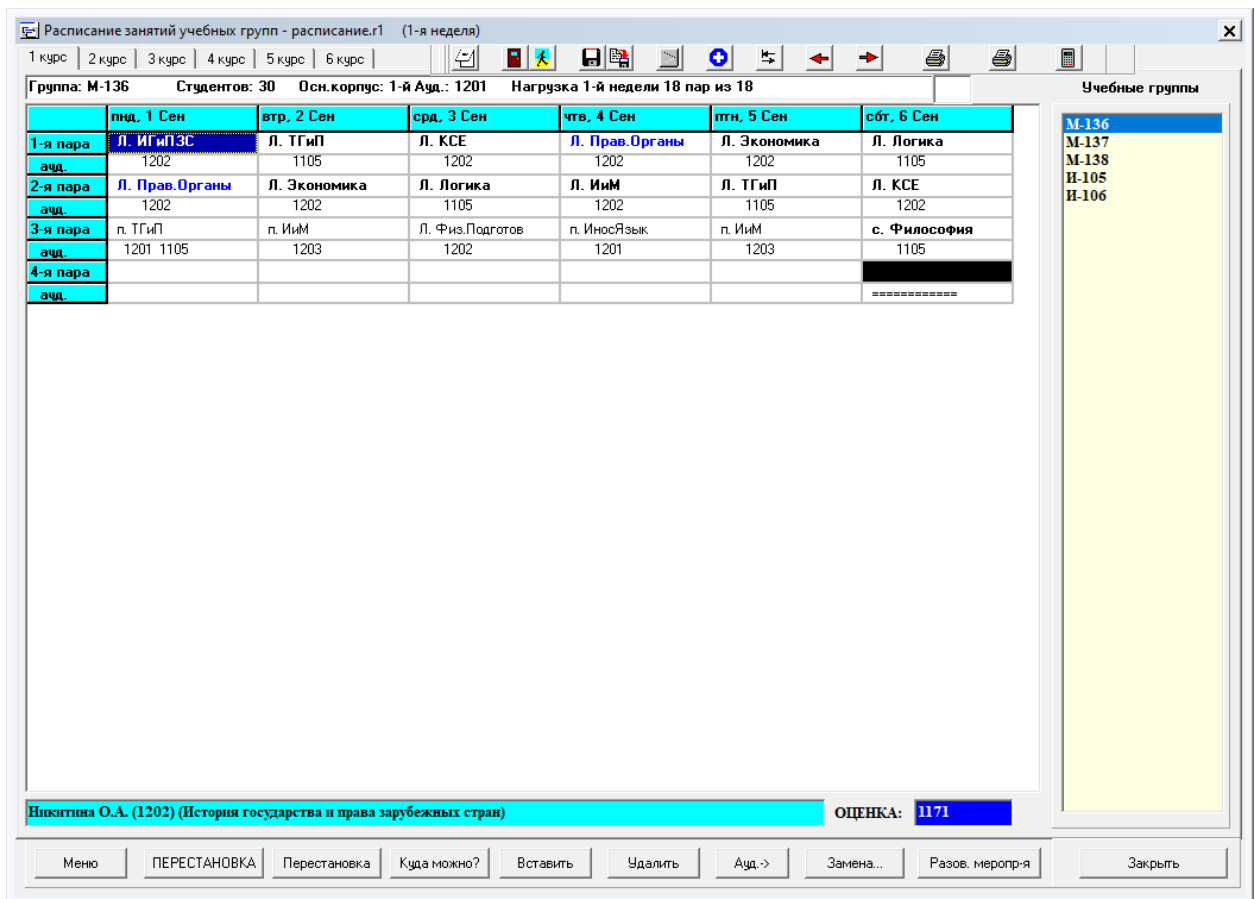


Рисунок 1.5 – Вікно з розкладом програми «АВТОРозклад» [19]

Таким и чином, є актуальним проведення досліджень в області автоматизованої побудови розкладів занять та розробка нових методів ефективної автоматизованої побудови розкладів занять для вишів. Є підстави стверджувати, що для вирішення означеної задачі є перспективним

використання генетичних алгоритмів, тому розробка засобів автоматизованої побудови розкладів занять за генетичним алгоритмом є актуальною.

1.3 Аналіз сучасних наукових публікацій з проблеми автоматизації формування розкладів занять

В останні роки значна увага приділяється коректному формуванню розкладу вищів. З сьогоднішніми об'ємами надання вищої освіти розробка таких розкладів є дуже важкою і трудомісткою задачею. Тому ця проблема всебічно розглядається вже декілька років та пропонуються нові рішення для розв'язання даної задачі.

Наприклад у публікації «Математична постановка задачі складання розкладу занять» [20] автор розгляну математичний підхід до вирішення задачі на основі різних побажань викладачів та студентів до розкладу у вищих навчальних закладах. Автор ввів поняття узагальнених та віртуальних груп. Також проаналізував та виокремив групи обмежень для задачі створення розкладів вищих навчальних закладів.

У публікації «Стандартизація даних для складання розкладу в навчальних закладах» [21] автори пропонують для вирішення задачі складання розкладу вищих навчальних закладів використати XML-технології. А також розглядають процес стандартизації вимог до складання розкладів.

Доволі популярним є алгоритм побудови розкладів на основі генетичного алгоритму. Саме він описаний у роботах «Складання розкладів занять у дистанційних системах навчання» [22] та «Exam Timetabling Problem using Genetic algorithm [23]. Генетичні алгоритми застосовують також і в публікації «Дослідження генетичних алгоритмів та їх застосування в автоматизованій системі розподілу навантаження для викладачів і студентів» [24].

В останній час доволі часто застосовуються також гібридні методи складання розкладу у вищих навчальних закладах. Наприклад, у роботі Optimal Time-Table Generation by Hybridized Bacterial Foraging and Genetic Algorithm [25]

автор пропонує метод генерації розкладу за допомогою комбінованого підходу методом підживлення бактерій та генетичного алгоритму. Даний підхід є доволі складний та ресурсозатратний.

У публікації [26] запропоновано використовувати ідеї системного підходу при складанні розкладів навчальних занять для освітніх систем масового навчання, які є складними організаційними системами. Запропоновано композиційний генетичний алгоритм, особливістю якого є:

- структуроване уявлення об'єктів (особі та хромосом) генетичного алгоритму;
- модифікація процедур схрещування та мутації стосовно структурованого представлення об'єктів генетичного алгоритму.

У публікації [27] автор досліджує ефективність паралельних ГА для вирішення задачі складання оптимального розкладу занять. При чому розклад повинен відповідати певним вимогам. А саме, певні викладачі, групи та аудиторії не можуть брати участь в навчальному процесі у певні проміжки часу, не повинно бути вікон, для деяких занять можуть створюватись підгрупи або об'єднуватись у великі групи, викладачі та групи не повинні протягом дня пересуватись між заняттями на великі відстані, заняття бажано повинні починатись зранку. Провівши ряд досліджень, автор зазначає, що використання паралельних ГА є ефективним рішенням для пошуку оптимального розв'язку для задачі створення розкладу занять.

Отже, як видно з аналізу публікацій, проблема автоматизованої генерації розкладу занять вишів має багато розв'язків. Доволі часто для розв'язку даної задачі використовуються генетичні алгоритми. Усі вони мають свої переваги та недоліки, проте цей напрям однозначно є актуальним.

1.4 Постановка задачі

Мета кваліфікаційної роботи магістра полягає у розробці методу побудови розкладів занять за генетичним алгоритмом, який дозволяє із

урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму. В якості вхідних зразків розкладів слід передбачити як можливість введення вручну існуючих аналогів, так і можливість автоматизованого створення зразків розкладів.

При оцінці сформованих розкладів занять у межах роботи генетичного алгоритму потрібно врахувати пріоритети викладачів щодо бажаної аудиторії проведення пари, днів тижня та власне номерів пар. Також слід врахувати пріоритети академічних груп щодо рекомендованої кількості пар в день і рекомендованої кількості пар в тиждень. Окрім урахування наведених пріоритетів викладачів та пріоритетів академічних груп, слід забезпечити врахування показників:

- важливості врахування пріоритетів викладачів;
- важливості врахування пріоритетів потоків;
- важливості відсутності вікон у викладачів;
- важливості відсутності вікон у групах.

Обмеженнями на роботу методу побудови розкладів занять за генетичним алгоритмом мають бути:

- максимальна кількість пар в день у викладача;
- максимальна кількість пар в тиждень у викладача;
- максимальна кількість пар в день у групи;
- максимальна кількість пар в тиждень у групи.

Таким чином, для досягнення поставленої мети розробки методу побудови розкладів занять за генетичним алгоритмом необхідно розв'язати наступні *задачі дослідження*:

1. Провести аналіз відомих підходів до автоматизованої побудови розкладів занять й шляхів до їх удосконалення.

2. Вдосконалити інформаційну модель розкладу занять як вичерпного подання предметної області, достатнього для побудови розкладів занять за генетичним алгоритмом.

3. Удосконалити метод побудови розкладів занять за генетичним алгоритмом, який використовує математичну модель обрахунку відповідності розкладу занять вимогам.

4. Розробити математичну модель обрахунку відповідності розкладу занять вимогам.

5. Розробити інформаційну систему генерації розкладів занять за генетичним алгоритмом згідно створеного методу.

6. Провести прикладне та функціональне дослідження методу побудови розкладів занять за генетичним алгоритмом.

Висновки до розділу 1

В розділі за результатом дослідження сучасного стану проблеми автоматизації побудови розкладів занять визначено, що завдання оптимізації розкладів занять є актуальним. Встановлено, що в останні роки значна увага приділяється коректному формуванню розкладу вишів. З сьогоднішніми об'ємами надання вищої освіти розробка таких розкладів є дуже важкою і трудомісткою задачею. Оптимальним розклад є тоді, коли він повністю покриває зв'язок навчальних дисциплін під час процесу навчання, задовольняє певним умови та обмеження, а також враховує досвід викладачів. Якщо при складанні розкладу не враховані всі зв'язки між дисциплінами, або не враховано всі обмеження, то це впливає на якість навчального процесу, що звісно погіршує рівень підготовки студентів. Тому розклад навчальних занять є документом, що засвідчує трудовий режим викладачів, тому оптимальний розклад здатний позитивно впливає на працездатність як викладачів, так і студентів.

Проте на сьогоднішній день все ще доволі поширена ручна побудова розкладів навчальних дисциплін у вищих навчальних закладах. Хоч вона і займає багато часу та є великий ризик виникнення помилок із-за людського фактора, проте не всі виші можуть дозволити собі використовувати програмне забезпечення, що автоматизовано будує навчальні розклади. Саме тому є

актуальним проведення досліджень в цій області та розробка нових методів ефективної автоматизованої побудови розкладів занять для вишів.

Аналіз сучасних наукових публікацій з проблеми автоматизації формування розкладів занять виявив, що для вирішення означеної задачі багато авторів схиляються до перспективності використання генетичних алгоритмів, тому розробка засобів автоматизованої побудови розкладів занять за генетичним алгоритмом є актуальною.

Відповідно, метою кваліфікаційної роботи магістра визначено розробку методу побудови розкладів занять за генетичним алгоритмом, який дозволяє із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму. При цьому у якості вхідних зразків розкладів доцільно передбачити як можливість введення вручну існуючих аналогів, так і можливість автоматизованого створення зразків розкладів.

Розділ 2

Метод побудови розкладів занять за генетичним алгоритмом

2.1 Інформаційна модель розкладу занять вищого навчального закладу

Інформаційна модель розкладу занять потрібна для формального подання сутностей предметної області в вигляді множин, що забезпечує подальше використання цих відомостей для розробки й реалізації методів та засобів автоматизації побудови розкладів занять за генетичним алгоритмом. Тобто розроблена модель містить подання всіх необхідних сутностей для побудови розкладів занять за генетичним алгоритмом, включаючи параметри роботи генетичного алгоритму, обмеження на компоненти розкладу занять та критерії оцінки розкладів.

Інформаційна модель розкладу занять вищого навчального закладу R має наступний вигляд:

$$\{V \cup A \cup T \cup M \cup G \cup Y \cup P \cup U \cup D \cup E \cup F \cup L \cup B \cup C \cup H \cup Q \cup K\} \subset R, \quad (2.1)$$

де V – множина викладачів, A – множина аудиторій, T – множина типів аудиторій і занять, M – множина входжень підгруп у потоки, G – множина груп, Y – множина підгруп, P – множина потоків, U – множина кафедр як включень викладачів і потоків до аудиторного фонду, D – множина днів тижня, E – множина вимог на планування пар (як комбінацій викладача, потоку, дисципліни та типу заняття), F – множина запланованих у розкладі пар (як комбінацій вимог на планування пари, аудиторії, дня тижня та номеру пари), L – множина пріоритетів викладачів до пар, B – множина пріоритетів викладачів до аудиторій, C – множина пріоритетів викладачів до днів тижнів, H – множина оцінок розкладів для груп, Q – множина оцінок розкладів для викладачів, K – множина вагових коефіцієнтів для оцінки розкладів.

Таким чином, розроблено інформаційну модель розкладу занять, яка відрізняється тим, що містить подання всіх необхідних сутностей для побудови

розкладів занять за генетичним алгоритмом, включаючи параметри роботи генетичного алгоритму, обмеження на компоненти розкладу занять та критерії оцінки розкладів.

2.2 Структура вхідних даних та схема методу побудови розкладів занять за генетичним алгоритмом

Метод побудови розкладів занять за генетичним алгоритмом призначений для формування розкладу занять з максимальним значенням оцінки відповідності розкладу занять поставленим вимогам за інформацією по елементах розкладів, наявними зразками розкладів, параметрами роботи генетичного алгоритму, параметрами обмеження на роботу методу та критеріями оцінки розкладів. На Рисунку 2.1 зображено схему етапів роботи методу побудови розкладів занять за генетичним алгоритмом.

Вхідними даними методу побудови розкладів занять за генетичним алгоритмом є масив даних (Рисунок 2.2), що групуються за ознаками у інформацію по елементах розкладів, наявні зразки розкладів, параметри роботи генетичного алгоритму, параметри обмеження на роботу методу та критерії оцінки розкладів. У розгорнутому вигляді масив вхідних даних має наступний вміст:

- інформація по елементах розкладів (викладачі, дисципліни, потоки, групи, аудиторії, дні тижня, типи занять, типи аудиторій);
- наявні зразки розкладів;
- параметри роботи генетичного алгоритму (дозволи на наскрізну та точкову мутації);
- обмеження на роботу методу (максимальна кількість пар в день у викладача, максимальна кількість пар в тиждень у викладача, максимальна кількість пар в день у групи, максимальна кількість пар в тиждень у групи);

– критерії оцінки розкладів (важливість врахування пріоритетів викладачів, важливість врахування пріоритетів потоків, важливість відсутності вікон у викладача, важливість відсутності вікон у підгрупи).

На Етапі 1 виконується ініціація нульової епохи роботи генетичного алгоритму. Для цього спершу проводиться перевірка повноти відомостей у вхідних даних. Далі здійснюється встановлення значень по замовчуванню (середніх) замість відсутніх у блоках параметрів роботи генетичного алгоритму. Також здійснюється встановлення обмежень на роботу методу та критеріїв оцінки розкладів.

Наступним кроком виконується аналіз достатньої кількості зразків розкладів для еволюції й перевірка можливості формування розкладу і несуперечності вхідних даних. У випадку потреби автоматизовано чи шляхом клонування існуючих здійснюється створення додаткових зразків розкладів за потреби. Результатом цих дій є сформована популяція зразків розкладів.

При роботі генетичного алгоритму на Етапі 2.1 відбувається створення нових зразків розкладів шляхом наскрізної мутації, одночасно на Етапі 2.2 відбувається створення нових зразків розкладів заняття шляхом точкової мутації. При наскрізній мутації в зразках розкладів відбувається повна заміна двох днів тижня чи аудиторій однакового призначення. При точковій мутації відбувається заміна одного параметру (аудиторія, номер пари, день тижня) на випадковий із числа доступних і несуперечливих наявним позиціям розкладу.

Як наслідок, на Етапі 3 здійснюється одержання нових зразків розкладів. Для цього спершу відбувається формування поточної популяції зразків розкладів. Далі виконується оцінка зразків розкладів методом обрахунку відповідності розкладу заняття вимогам. Це дозволяє виконати формування результуючої вибірки з поточного покоління зразків розкладів.

Етап 4 відповідає за перевірку умов виходу з еволюції епох. Такими умовами виходу з еволюції епох можуть бути одержання кращого ніж існуючі зразка розкладу, досягнення заданої оцінки відповідності розкладу заняття вимогам або вичерпання заданої кількості епох.

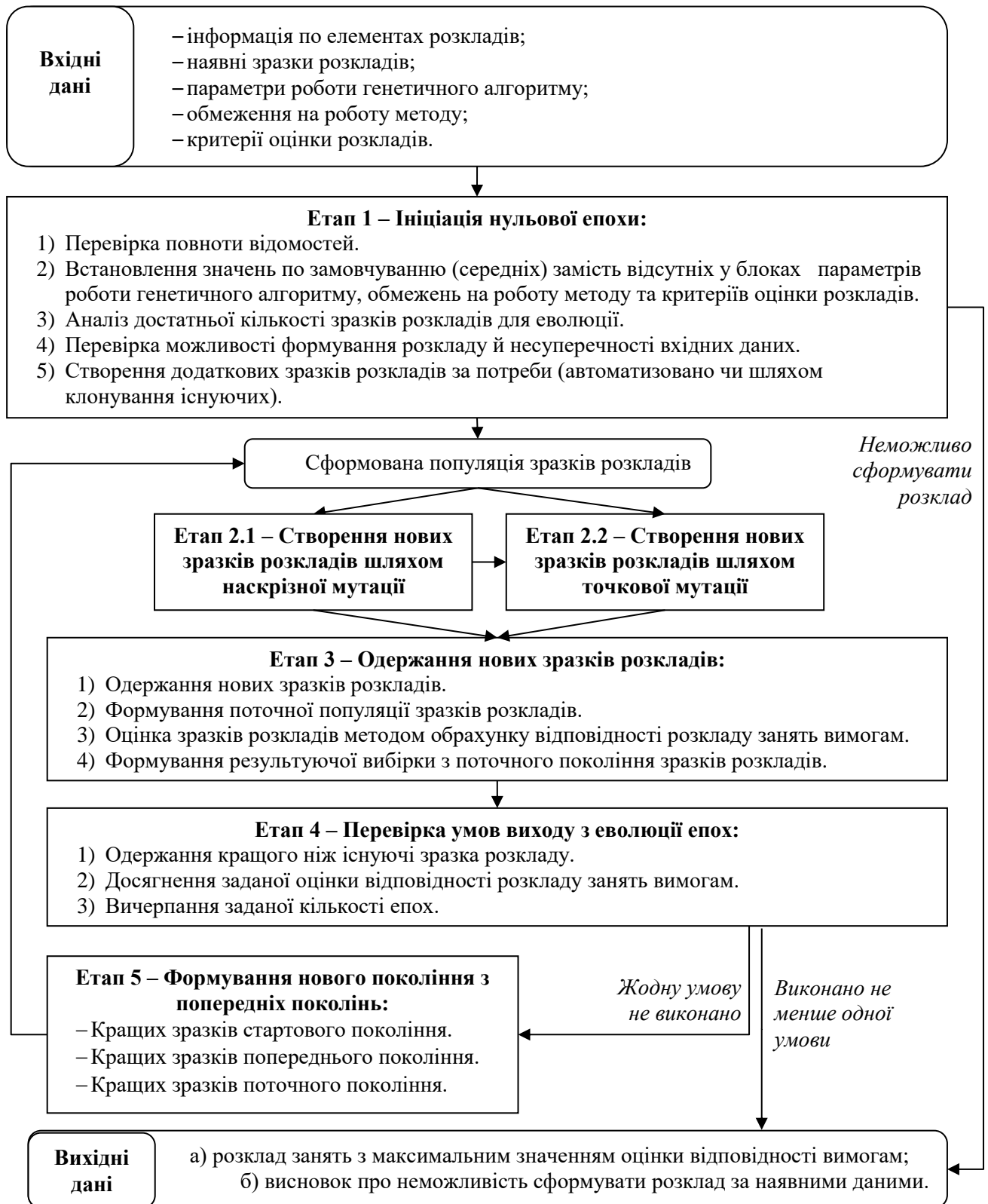


Рисунок 2.1 – Схема методу побудови розкладів занять за генетичним алгоритмом

Якщо при перевірці умов виходу з еволюції епох жодну умову не виконано, то відбувається перехід на Етап 5 для формування нового покоління з

зразків розкладів із попередніх поколінь. При цьому для формування нового покоління використовується визначені кількості кращих зразків стартового покоління, кращих зразків попереднього покоління та кращих зразків поточного покоління.

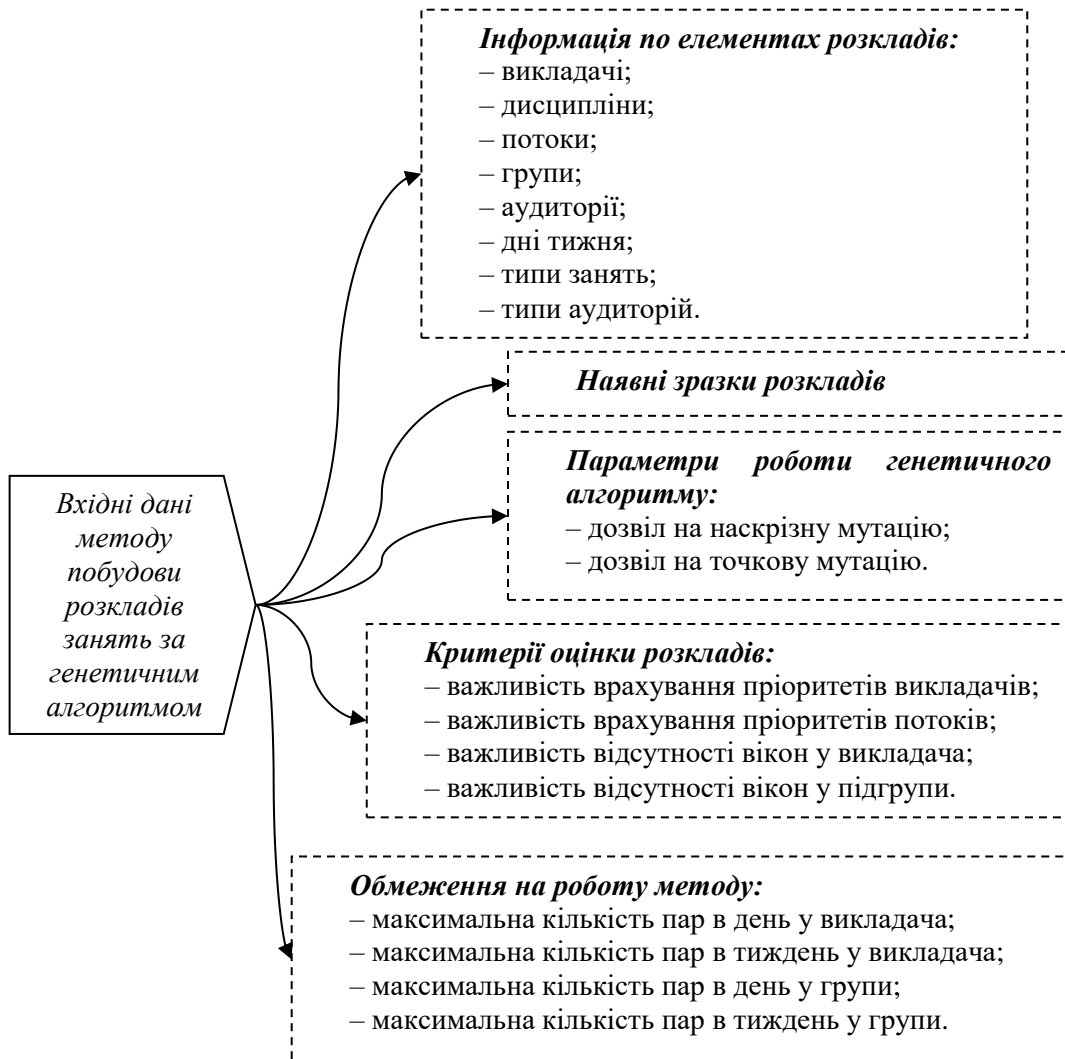


Рисунок 2.2 – Структура вхідних даних методу побудови розкладів занять за генетичним алгоритмом

Якщо ж при перевірці умов виходу з еволюції епох виконано не менше одної умови, то відбувається формування вихідних даних методу у вигляді розкладу занять із максимальним значенням оцінки відповідності розкладу занять вимогам.

Якщо висновок на Етапі 1 було встановлено, що неможливо сформувати жоден зразок розкладу, то в такому випадку відбувається формування вихідних даних методу у вигляді повідомлення про неможливість сформувати розклад за наявними даними

Таким чином, розроблено метод побудови розкладів занять за генетичним алгоритмом призначений для автоматизованого формування зразків розкладів занять й пошуку серед них розкладу з максимальним значенням оцінки відповідності розкладу занять поставленим вимогам за використання генетичного алгоритму. Формування зразків розкладів занять виконується за інформацією по елементах розкладів, наявними зразками розкладів, параметрами роботи генетичного алгоритму, параметрами обмеження на роботу методу та критеріями оцінки розкладів.

Визначними рисами розробленого методу побудови розкладів занять за генетичним алгоритмом є врахування при оцінюванні зразків розкладів таких критеріїв, як пріоритети викладачів (аудиторії, дні тижня, пари), пріоритети потоків (рекомендовані кількості пар в день і в тиждень), відсутність вікон у викладача і підгрупи, а також врахування обмежень на роботу алгоритму таких як максимальна кількість пар в день і в тиждень у викладача й максимальна кількість пар в день і в тиждень у підгрупи.

Створений метод побудови розкладів занять за генетичним алгоритмом використовує математичну модель обрахунку відповідності розкладу занять вимогам і дозволяє за наявними в інформаційній моделі розкладу занять параметрами із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму.

2.3 Математична модель обрахунку відповідності розкладу занять вимогам

При роботі методу побудови розкладів занять за генетичним алгоритмом враховуються обмеження на роботу алгоритму (максимальна кількість пар в день у викладача, максимальна кількість пар в тиждень у викладача, максимальна кількість пар в тиждень у підгрупи та максимальна кількість пар в день у підгрупи), у випадку порушень яких визначається неможливість створення розкладу. При цьому інша частина параметрів роботи методу (важливість врахування пріоритетів викладачів (аудиторії, дні тижня, пари), важливість врахування пріоритетів потоків (рекомендовані кількості пар в день і в тиждень), важливість відсутності вікон у викладача, важливість відсутності вікон у підгрупи) визначає критерії оцінки розкладів.

Математична модель обрахунку відповідності розкладу занять вимогам дозволяє за цими наявними в інформаційній моделі розкладу занять параметрами автоматизовано обраховувати умовну оцінку відповідності визначеного зразка розкладу занять вимогам.

Умовна оцінка відповідності визначеного зразка розкладу занять вимогам визначеного викладача OV обраховується наступним чином:

$$OV = VA_{poz} - VA_{neg} + VD_{poz} - VD_{neg} + VP_{poz} - VP_{neg}, \quad (2.2)$$

де VA_{poz} – кількість випадків пар у аудиторіях в розкладі згідно позитивних пріоритетів викладача, VA_{neg} – кількість випадків пар у аудиторіях в розкладі згідно негативних пріоритетів викладача, VD_{poz} – кількість випадків пар у дні тижня в розкладі згідно позитивних пріоритетів викладача, VD_{neg} – кількість випадків пар у дні тижня в розкладі згідно негативних пріоритетів викладача, VP_{poz} – кількість випадків пар з номером проведення в розкладі згідно позитивних пріоритетів викладача, VP_{neg} – кількість випадків пар з номером проведення в розкладі згідно негативних пріоритетів викладача.

Умовна оцінка відповідності визначеного зразка розкладу занять вимогам визначеної підгрупи OP обраховується наступним чином:

$$OP = PD_{poz} - PD_{neg} + PT_{poz} - PT_{neg}, \quad (2.3)$$

де PD_{poz} – кількість днів в розкладі у яких кількість пар не відрізняється від рекомендованої, PD_{neg} – кількість днів в розкладі у яких кількість пар відрізняється від рекомендованої, PT_{poz} – кількість тижнів (розглядаються окремо чисельник та знаменник) в розкладі у яких кількість пар не відрізняється від рекомендованої, PT_{neg} – кількість тижнів (розглядаються окремо чисельник та знаменник) в розкладі у яких кількість пар відрізняється від рекомендованої.

Загальна умовна оцінка відповідності визначеного зразка розкладу занять вимогам O визначається наступним чином:

$$O = k_{OV} \sum_{i=1}^n OV_i + k_{OP} \sum_{j=1}^m OP_j - k_{VV} \sum_{i=1}^n VV_i - k_{VP} \sum_{j=1}^m VP_j, \quad (2.4)$$

де n – кількість викладачів у розкладі який аналізується, m – кількість підгруп у розкладі який аналізується, OV_i – умовна оцінка відповідності визначеного зразка розкладу занять вимогам визначеного викладача i , OP_j – умовна оцінка відповідності визначеного зразка розкладу занять вимогам визначеної підгрупи j , VV_i – кількість вікон в розкладі який аналізується у визначеного викладача i , VP_j – кількість вікон в розкладі який аналізується у визначеної підгрупи j , k_{OV} – ваговий коефіцієнт важливості врахування пріоритетів викладачів за розкладом, k_{OP} – ваговий коефіцієнт важливості врахування пріоритетів потоків за розкладом, k_{VV} – ваговий коефіцієнт важливості відсутності вікон у викладача за розкладом, k_{VP} – ваговий коефіцієнт важливості відсутності вікон у підгрупи за розкладом.

Таким чином, розроблено математичну модель обрахунку відповідності розкладу занять вимогам, що дозволяє за наявними в інформаційній моделі розкладу занять параметрами автоматизовано обраховувати умовну оцінку відповідності визначеного зразка розкладу занять вимогам.

2.4 Функції інформаційної системи для побудови розкладів занять за генетичним алгоритмом

Для прикладного використання методу побудови розкладів занять за генетичним алгоритмом й створення відповідної інформаційної системи, потрібно визначити необхідний обсяг функцій для її реалізації. Це дозволить у подальшому сформулювати її структуру та множину функцій для дослідження її функціональності.

Таким чином, інформаційна система генерації розкладів занять за генетичним алгоритмом в випадку успішної реалізації має виконувати наступні обов'язкові функції:

- робота з даними викладачів (викладачі, кафедри, прив'язка аудиторій);
- робота з даними академічних груп (групи, підгрупи, потоки, включення підгруп у потоки);
- робота з аудиторним фондом (аудиторії, типи аудиторій);
- створення зразків розкладів занять в редакторі розкладів;
- автоматизоване створення розкладів занять без оцінювання результатів;
- коригування створених зразків розкладів занять в редакторі розкладів;
- робота з параметрами обмежень на генерацію розкладів;
- робота з пріоритетами викладачів до елементів розкладу;
- робота з ваговими коефіцієнтами важливості при оцінюванні розкладу;
- запуск автоматизованої генерації розкладів занять;
- автоматизована перевірка повноти й несуперечності даних та їх доповнення;
- автоматизований аналіз достатньої кількості зразків розкладів для еволюції;
- автоматизоване додаткове створення додаткових зразків розкладів за потреби;
- автоматизоване створення нових зразків розкладів шляхом мутацій;

- автоматизована оцінка зразків розкладів методом обрахунку відповідності розкладу занять вимогам;
- автоматизована перевірка досягнення оцінки розкладів занять вимогам;
- автоматизована селекція для нових епох вибірок розкладів;
- автоматизоване виведення результатів роботи системи користувачеві.

Висновки до розділу 2

В розділі було розглянуто створені метод і засоби для побудови розкладів занять за генетичним алгоритмом. Зокрема, розроблено інформаційну модель розкладу занять, яка містить подання всіх необхідних сутностей для побудови розкладів занять за генетичним алгоритмом, включаючи параметри роботи генетичного алгоритму, обмеження на компоненти розкладу занять та критерії оцінки розкладів.

Також в розділі розроблено метод побудови розкладів занять за генетичним алгоритмом призначений для автоматизованого формування зразків розкладів занять й пошуку серед них розкладу з максимальним значенням оцінки відповідності розкладу занять поставленим вимогам за використання генетичного алгоритму. Формування зразків розкладів занять виконується за інформацією по елементах розкладів, наявними зразками розкладів, параметрами роботи генетичного алгоритму, параметрами обмеження на роботу методу та критеріями оцінки розкладів.

Визначними рисами розробленого методу побудови розкладів занять за генетичним алгоритмом є врахування при оцінюванні зразків розкладів таких критеріїв, як пріоритети викладачів (аудиторії, дні тижня, пари), пріоритети потоків (рекомендовані кількості пар в день і в тиждень), відсутність вікон у викладача і підгрупи, а також врахування обмежень на роботу алгоритму таких як максимальна кількість пар в день і в тиждень у викладача й максимальна кількість пар в день і в тиждень у підгрупи. Створений метод побудови розкладів занять за генетичним алгоритмом використовує математичну модель обрахунку

відповідності розкладу занять вимогам і дозволяє за наявними в інформаційній моделі розкладу занять параметрами із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вищів за допомогою еволюційного генетичного алгоритму.

При роботі методу побудови розкладів занять за генетичним алгоритмом враховуються обмеження на роботу алгоритму, у випадку порушень яких визначається неможливість створення розкладу. При цьому інша частина параметрів роботи методу визначає критерії оцінки розкладів. Тому було розроблено математичну модель обрахунку відповідності розкладу занять вимогам, що дозволяє за наявними в інформаційній моделі розкладу занять параметрами автоматизовано обраховувати умовну оцінку відповідності визначеного зразка розкладу занять вимогам.

Також в розділі наведено обов'язкові функції інформаційна система генерації розкладів занять за генетичним алгоритмом, яка потрібна для прикладного використання методу побудови розкладів занять за генетичним алгоритмом. Це дозволить у подальшому сформувати її структуру та множину функцій для дослідження її функціональності в випадку успішної реалізації.

Розділ 3

Інформаційна система генерації розкладів занять за генетичним алгоритмом

3.1 Архітектура інформаційної системи генерації розкладів занять за генетичним алгоритмом

Відповідно до визначених згідно методу побудови розкладів занять за генетичним алгоритмом функцій інформаційної системи, було спроектовано відповідну структуру, зображену на Рисунку 3.1. Інформаційна система генерації розкладів занять за генетичним алгоритмом включає в себе базу даних, в якій зберігаються наступні сутності й їх властивості: викладачі, аудиторії, типи аудиторій і занять, входження підгруп у потоки, групи, підгрупи, потоки, кафедри як включення викладачів і потоків до аудиторного фонду, дні тижня, вимоги на планування пар (як комбінацій викладача, потоку, дисципліни й типу заняття), заплановані у розкладі занять пари, пріоритети викладачів до пар, пріоритети викладачів до аудиторій, пріоритети викладачів до днів тижнів, оцінки розкладів для груп, оцінки розкладів для викладачів, вагові коефіцієнти для оцінки розкладів.

Крім БД, інформаційна система генерації розкладів занять за генетичним алгоритмом має підсистеми: підсистема роботи з відомостями бази даних по розкладах, підсистема користувацького створення зразків розкладів, підсистема роботи з налаштуваннями генерації розкладів та підсистема автоматизованої генерації розкладів.

Підсистема роботи з відомостями бази даних по розкладах виконує функції роботи з даними викладачів (викладачі, кафедри, прив'язка аудиторій), роботи з даними академічних груп (групи, підгрупи, потоки, включення підгруп у потоки) та роботи з аудиторним фондом (аудиторії, типи аудиторій).

Підсистема користувацького створення зразків розкладів виконує функції створення зразків розкладів занять в редакторі розкладів, автоматизованого

створення розкладів занять без оцінювання результатів та коригування створених зразків розкладів занять в редакторі розкладів.

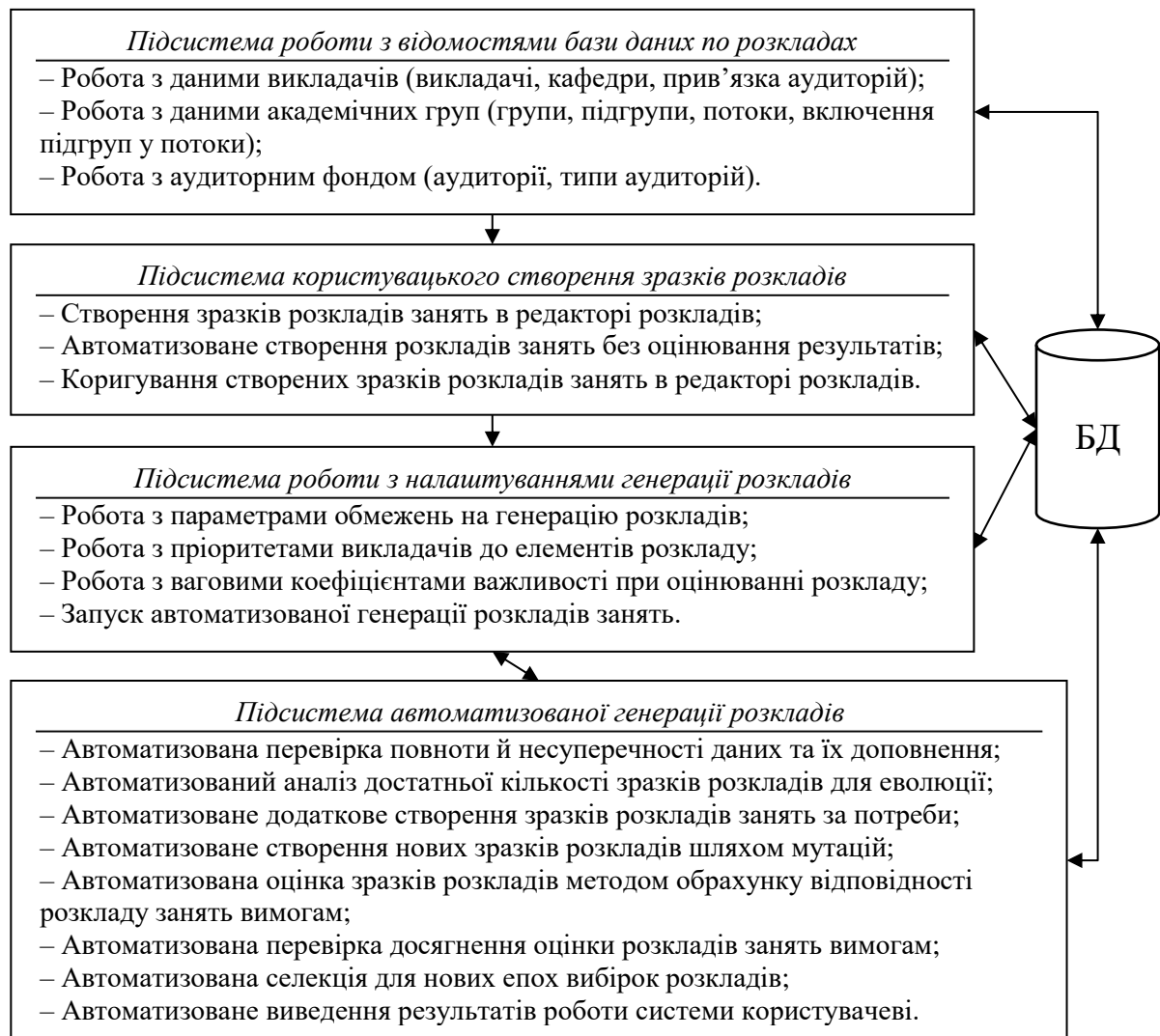


Рисунок 3.1 – Схема інформаційної системи генерації розкладів занять за генетичним алгоритмом

Підсистема роботи з налаштуваннями генерації розкладів виконує функції роботи з параметрами обмежень на генерацію розкладів, роботи з пріоритетами викладачів до елементів розкладу, роботи з ваговими коефіцієнтами важливості при оцінюванні розкладу та запуску автоматизованої генерації розкладів занять.

Підсистема автоматизованої генерації розкладів виконує функції автоматизованої перевірки повноти й несуперечності даних та їх доповнення,

автоматизованого аналізу достатньої кількості зразків розкладів для еволюції, автоматизованого додаткового створення зразків розкладів занять за потреби, автоматизованого створення нових зразків розкладів занять шляхом мутацій, автоматизованої оцінки зразків розкладів методом обрахунку відповідності розкладу занять вимогам, автоматизованої перевірки досягнення оцінки розкладів занять вимогам, автоматизованої селекції для нових епох вибірок розкладів та автоматизованого виведення результатів роботи системи користувачеві. При наскрізній мутації в зразках розкладів відбувається повна заміна двох днів тижня чи аудиторій однакового призначення. При точковій мутації відбувається заміна одного параметру (аудиторія, номер пари, день тижня) на випадковий із числа доступних і несуперечливих наявним позиціям розкладу.

Таким чином, розроблено інформаційну систему генерації розкладів занять за генетичним алгоритмом, що дозволяє з використанням створених інформаційної моделі розкладу занять, методу побудови розкладів занять за генетичним алгоритмом та математичної моделі обрахунку відповідності розкладу занять вимогам виконувати ручне та автоматизоване створення нових розкладів, формувати нові похідні розклади із використанням генетичного алгоритму, оцінювати сформовані розклади занять із урахуванням вимог викладачів й академічних груп до параметрів проведення пар й визначати кращі зразки розкладів занять за результатом порівняння їх оцінок відповідності вимогам.

3.2 Структура та функціональне призначення складових системи

Структура складових інформаційної системи генерації розкладів занять за генетичним алгоритмом у вигляді класів та їх властивостей має забезпечити можливість системи виконувати ручне та автоматизоване створення нових розкладів, формувати нові похідні розклади із використанням генетичного алгоритму, оцінювати сформовані розклади занять із урахуванням вимог

викладачів й академічних груп до параметрів проведення пар й визначати кращі зразки розкладів занять за результатом порівняння їх оцінок відповідності вимогам. Прикладна структура складових інформаційної системи генерації розкладів занять продається у вигляді діаграми класів. Діаграма класів – діаграма, що демонструє класи системи, їх атрибути, методи та взаємозв'язки між ними. Реалізовану діаграму класів зображено на рисунку 3.1.

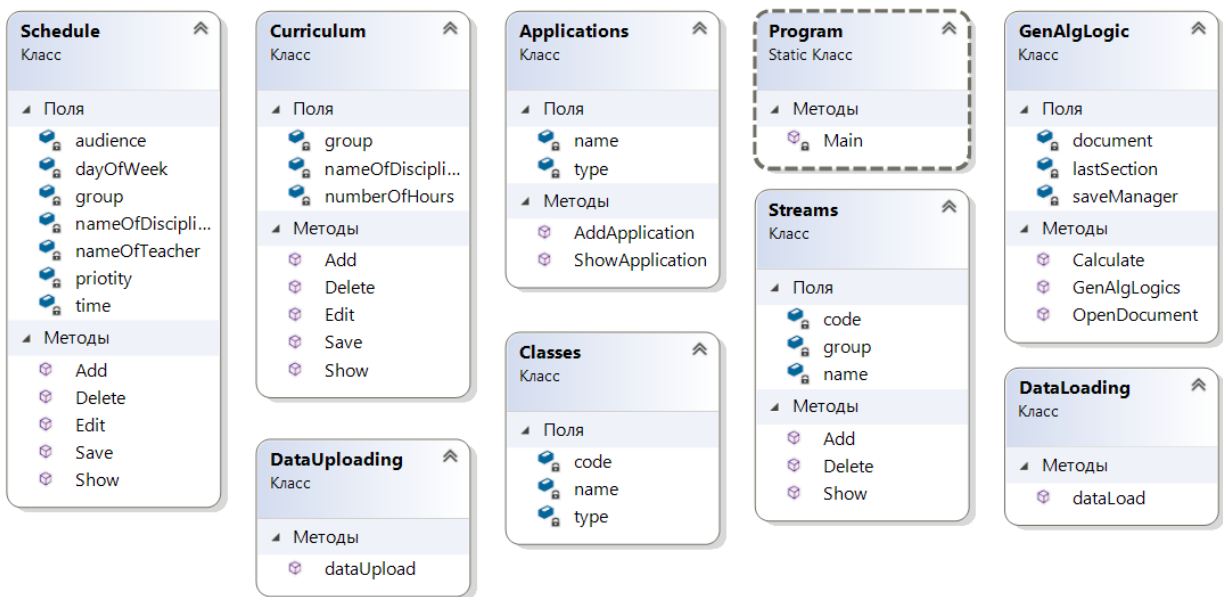


Рисунок 3.1 – Діаграма класів інформаційної системи генерації розкладів занять за генетичним алгоритмом

На представленій діаграмі зображено 8 класів, а саме:

- Curriculum;
- Schedule;
- GenAlgLogic;
- DataLoading;
- Streams;
- Classes;
- DataUploading.

Клас «Curriculum» відповідає за операції, що проводимуться із навчальними планами, а саме їх додання, редагування, завантаження та видалення.

Клас «DataUploading» призначений для вивантаження згенерованих розкладів на носій інформації.

Клас «DataLoading» створений для завантаження навчальних планів та розкладів із комп'ютеру користувача для проведення подальших операцій над ним.

Клас «Schedule» створений для проведення операцій, що стосуватимуться розкладів, визначення їх оцінки, додання нових розкладів та їх генерації, редагування та видалення.

Клас «Applications» створений для генерації, перегляду, видалення та проведення інших операцій із заявками до розкладів.

Клас «Streams» має методи, що забезпечують створення, редагування видалення та перегляд потоків студентів та груп, що їх формують.

Створені класи та методи у них дозволяють у повному обсязі виконати завдання кваліфікаційної роботи та створити систему генерації розкладів занять шляхом використання генетичного алгоритму.

3.3 Аргументація вибору рекомендованих засобів розробки інформаційної системи

Для вибору СКБД при розробці інформаційної системи формування розкладів занять за генетичним алгоритмом навчання було проаналізовано PostgreSQL та SQL Server.

PostgreSQL – це СКБД класу корпоративного використання з відкритим вихідним кодом. Підтримує як SQL, так і JSON для реляційних і нереляційних запитів згідно розширюваності та відповідності SQL. PostgreSQL підтримує розширені типи даних і функції оптимізації продуктивності, які доступні лише в дорогих комерційних базах даних, таких як Oracle і SQL Server. PostgreSQL

підтримує досвідчене співтовариство розробників, які зробили величезний внесок, щоб зробити його високонадійною системою СКБД [28].

PostgreSQL підтримує географічні об'єкти, тому його можна використовувати як сховище геопросторових даних для сервісів на основі розташування та геоінформаційних систем.

Недоліки PostgreSQL [29]:

- PostgreSQL не дуже хороший, коли справа доходить до продуктивності.
- Має нижчу популярність, ніж інші СКБД.
- Мало кваліфікованих фахівців.
- Що стосується швидкості, PostgreSQL дуже повільне СКБД.
- Непростий процес установлення.

SQL Server. SQL Server – це СКБД, або RDBMS, розроблена та продана Microsoft. Подібно до іншого програмного забезпечення СКБД, SQL Server побудований на основі SQL, стандартної мови програмування для взаємодії з реляційними базами даних. Сервер SQL пов'язаний з Transact-SQL або T-SQL, реалізацією SQL від Microsoft, яка додає набір власних програмних конструкцій. SQL Server працює виключно в середовищі Windows понад 20 років. У 2016 році Microsoft зробила його доступним для Linux. SQL Server 2017 став загальнодоступним у жовтні 2016 року, який працював як на Windows, так і на Linux [30].

Основним компонентом SQL Server є Database Engine. Database Engine складається з реляційного механізму, який обробляє запити, і механізму зберігання, який керує файлами бази даних, сторінками, індексом тощо. Об'єкти бази даних, такі як збережені процедури, представлення та тригери, також створюються та виконуються Database Engine [30].

Переваги SQL Server [31]:

- Хороша продуктивність: виклики процедур швидкі й ефективні, оскільки збережені процедури компілюються один раз і зберігаються у виконуваний формі. Виконуваний код автоматично кешується, що зменшує вимоги до пам'яті.

– Вища продуктивність: тому що можна повторно використовувати без необхідності знову і знову переписувати оператори SQL, тому продуктивність вища.

– Масштабування та обслуговування: процедури зберігання підвищують масштабованість за рахунок ізоляції обробки додатків на сервері. Якщо потрібно внести будь-які зміни, просто треба замінити збережену процедуру, не торкаючись коду програми.

– Хороша безпека: коли потрібно обмежити прямий доступ до даних певного користувача кількома таблицями, можна написати збережену процедуру для доступу до даних і дозволити цьому користувачеві використовувати лише збережені.

Недоліки SQL Server [31]:

– Перевірка: бізнес-логіка інкапсульована в збережені процедури, тому її важко перевірити. Написання тестів для будь-якої бізнес-логіки в збереженій процедурі неможливо, оскільки немає способу чітко відокремити бізнес-логіку;

– Можливість налагодження: залежно СКБД, налагодження процедур буде неможливим або надзвичайно складним для розуміння;

– Одним із найбільших недоліків збережених процедур є те, що надзвичайно важко знати, які частини системи їх використовують, а які ні.

Зважаючи на предметну область КРМ та на вимоги до СКБД по розробці програмного забезпечення, також враховуючи усі вищеописані недоліки та переваги, було прийнято рішення використовувати SQL Server у якості СКБД.

Оскільки розроблювана програма буде windows-застосуванням, вибір платформи зводиться до двох конкурентних платформ у цьому напрямі: Java та .NET Framework.

Java. Ніхто не може заперечувати внесок Java у трансформацію Інтернету. Завдяки тисячам вебсайтів і вебзастосувань, що працюють на платформі ява, вона залишається популярною протягом багатьох років. Але на популярність Java впливає поява нових платформ та інструментів розробки [32].

Оскільки розробник нового покоління має можливість вибирати з кількох платформ розробки застосувань, він може легко оцінити плюси та мінуси кожної платформи. Також важливо вибрати платформу розробки, яка повністю доповнює характер і вимоги проєкту.

Переваги Java як платформи розробки [32]:

- Відкритий код: на відміну від інших платформ розробки, Java є платформою з відкритим вихідним кодом;
- Сумісність із різними браузерами: розробники часто вибирають платформу для розробки вебзастосувань, враховуючи останні тенденції в технологіях;
- Простота розробки складних застосунків: через часті зміни в вимогах користувачів компаніям важливо запускати динамічні та потужні застосунки;
- Можливість повторного використання коду та сторонніх бібліотек: кожен розробник хоче створити потужний застосунок без написання довгих рядків коду.

Обмеження Java як платформи розробки [32]:

- Підтримка обмежених мов програмування: Як платформа розробки, Java підтримує додаткові мови програмування, такі як JRuby та Groovy. Але платформа не дозволяє користувачам скористатися перевагами цих мов для створення інноваційних сервісів;
- Інтерфейси користувача до Ajax: Використання AJAX змінило спосіб розробки вебзастосувань. Але більшість фреймворків Java не розроблено з повною підтримкою AJAX;
- Комплексні фреймворки: ряд платформ розробки застосувань зробили програмування простішим і швидшим. Крім того, оскільки Java заснована на C++, багато програмістів вважають, що створення корпоративних бізнес-застосувань є занадто складним.

.NET Framework. Це популярна безплатна платформа, яка в даний час використовується для багатьох різних типів застосунків, оскільки забезпечує

середовище програмування для більшості етапів розробки програмного забезпечення [33].

Запуск фреймворку .NET відбувся в 2002 році. Дотепер компанія розширила та інтегрувала багато речей у нього. Останнім доповненням був .NET 5 у 2019 році [33].

.NET Framework надає можливість перепроектувати програму. Це одна з найкорисніших функцій платформи .NET. Розробнику не потрібно починати з нуля, коли потрібно будь-яке коригування або новий продукт.

Широкий спектр інструментів: розробка .NET включає в себе надзвичайно потужні інструменти. Microsoft надає цілий ряд інструментів, які можна використовувати для спрощення роботи та підвищення ефективності.

Переваги .NET Framework [33]:

- Надійна система кешування. Платформа .NET має одну з найпростіших і найнадійніших систем кешування;
- Середовище розробки Visual Studio. Visual Studio є одним із найкращих інструментів, які присутні в платформі .NET.
- Кросплатформена функція. Спочатку ця функція була недоступна, але з запуском основного фреймворку ASP.NET була представлена кросплатформенна функція.
- Автоматичний моніторинг. Кодування – це складний процес, який неминуче призводить до помилок у коді. Якщо випадково щось піде не так, страждає вся робота. Отже, щоб уникнути цього, платформа .Net підтримує автоматичний моніторинг.
- Велика бібліотека. Сімейство .NET має численні бібліотеки, вбудовані в середину. Розробникам легко створити нове застосування, яке не потрібно будувати «з нуля» заново.
- Велика спільнота. .NET був створений Microsoft як фреймворк з відкритим вихідним кодом, що значно збільшило кількість користувачів.
- Гнучкість розгортання та просте обслуговування. .NET має чудові інструменти розробки, які полегшують розгортання та обслуговування.

Проте, цей фреймворк все ж має деякі недоліки [33]:

– Обмежена об'єктно-реляційна підтримка: оновлення є дуже поширеними і необхідні для будь-яких продуктів технологічного сектору. Проблема виникає, коли оновлення не дозволяють з'явитися попередній версії.

– Блокування постачальника: платформу .NET купує Microsoft. Це означає, що весь проєкт знаходиться в руках Microsoft. Тут розробники не отримують повний і єдиний контроль над програмою, а повинні дотримуватися правил Microsoft.

– Вартість ліцензування: .NET Framework є платформою з відкритим кодом, але іноді, коли розмір проєкту великий і складний, ліцензування стає дорогим.

Розробка під платформу .NET користується великим попитом через свої численні переваги. Отож враховуючи що у рамках теми КРМ буде створення саме windows-застосування, у якості фреймворку буде використовуватись .NET Framework.

У рамках фреймворку .NET Framework потрібно обрати мову для написання застосування. Для вибору мови було проаналізовано F#, Visual Basic та C#.

F# – функціональна мова для платформи .NET Framework, розроблена командою Microsoft Research. Розробляється з 2005 року і перейшла до версії 3.1 у грудні 2013 року. Вона розпочала як академічний дослідницький проєкт і протягом кількох років дозріла до мови, готової до виробництва, якою користуються багато комерційних компаній, особливо у фінансовому секторі [34].

Перевагами F# є [34]:

– Зосередження на вирішенні загальних проблем замість конкретних;

– F# заохочує використовувати загальні типи даних замість конкретних.

Той самий алгоритм, написаний для генериків, працюватиме з різними конкретними типами, такими як числа, рядки чи складні об'єкти. Наявність лише

однієї загальної реалізації для різних типів робить код більш придатним для повторного використання і залишає менше місця для помилок;

- F# має унікальний механізм роботи з різнорідними джерелами даних зручним уніфікованим способом, який називається постачальниками типів;

- У F# є 2 стандартні інструменти FsLex і FsYacc для генерування синтаксичних аналізаторів на основі формальної граматики. Ці інструменти названі на честь Lex і Yacc, популярних утиліт генератора аналізаторів у світі Unix. FsLex і FsYacc мають багаті діагностичні можливості і можуть бути включені в процес складання, допомагаючи оптимізувати розробку;

- F# заохочує використовувати трансформації, а не мутації. Трансформація не змінює об'єкт, отож, не змінює стан об'єкту. Він просто створює новий об'єкт і передає його, зберігаючи оригінальний об'єкт недоторканим;

- F# має розширену систему виведення типів, яка виводить типи значень із того, як ці значення використовуються, тому в більшості випадків рекомендується опускати типи в коді, дозволяючи F# визначити їх.

Недоліки F# [34]:

- Для людини, яка не мала досвіду функціонального програмування, може бути складно пристосуватися до нового способу мислення під час кодування на F#;

- З підходом трансформації над мутацією потрібно більш просунуті структури даних для ефективного маніпулювання з ними;

- F# не має функції перевантаження функцій, яку має C# для методів. Таким чином, дві функції в F#, які знаходяться в одному модулі, не можуть мати однакові імена, що створює труднощі з їх унікальним іменуванням;

- Microsoft вкладає багато коштів у створення найкращих інструментів для програмістів C#. На жаль, для F# не так багато інструментів, які роблять кодування менш зручним. F# не має базових інструментів рефакторингу.

Visual Basic. Visual Basic є мовою програмування, що пропонує загальну простоту використання в поєднанні з простотою реалізації графічного інтерфейсу користувача [35].

VB є інструментом розробки на основі графічного інтерфейсу, який пропонує швидший RAD, ніж більшість інших мов програмування. VB також має синтаксис, який є більш простим, ніж інші мови, візуальне середовище, яке легко зрозуміти.

Структура VB розроблена, щоб дозволити програмістам використовувати середовище для запису виконуваних файлів. Крім того, використовуючи VB, розробники можуть створювати програми, які можна використовувати як інтерфейс для баз даних. Інструменти VB можуть допомогти програмістам розробляти застосування або завершувати програмне забезпечення, при цьому дозволяючи їм відповідно змінювати та переглядати свою роботу.

Загалом, VB забезпечує швидкий розвиток програм на базі Windows, а також допомагає в доступі до баз даних за допомогою об'єктів даних ActiveX (ADO), дозволяючи програмістам використовувати елементи керування ActiveX і різні об'єкти.

Недоліки [36]:

VB.NET не може обробляти покажчики безпосередньо. Це значний недолік, оскільки вказівники дуже необхідні для програмування. Будь-яке додаткове кодування призведе до багатьох циклів процесора, що вимагає більше часу обробки.

Мова програмування C# являється об'єктно-орієнтованою мовою програмування. C# автоматично керує пам'яттю віддалених об'єктів за допомогою збирача сміття, що усуває турботи розробників і витрати пам'яті [37].

Тип C# є більш безпечним, ніж C++, і має лише надійні перетворення за замовчуванням (наприклад, цілочисельне розширення), які виконуються під час компіляції або під час виконання.

Синтаксис C# надзвичайно виразний, але він також простий та легкий для вивчення. Синтаксис C# буде безпосередньо ідентифікований будь-яким, хто добре знайомий із C, C++ або Java. Розробники, що знайомі з будь-якою із цих мов, зазвичай можуть почати ефективно працювати на C# за менший час. Синтаксис C# спрощує безліч складнощів C++ і пропонує унікальні функції, такі як нульові, перераховані, делеговані, лямбда-вирази та прямий доступ до пам'яті, що недоступні в Java. C# підтримує стандартні методи та типи, які пропонують кращу безпеку та продуктивність, а також ітератори, що дозволяють реалізаторам класів збирання описувати користувацькі ітераційні поведінки, які клієнтський код може просто використовувати. Вирази запиту, інтегровані в мову (LINQ), роблять запит із потужною типізацією першокласною мовною конструкцією.

Хоча конструкції C# безпосередньо слідують традиційним різним мовам високого рівня, C++ і C# є об'єктно-орієнтованою мовою програмування. Вона має величезну схожість з Java, має численні функції програмування, які роблять її привабливою для різних програмістів у всьому світі.

Список деяких важливих функцій C# [37]:

- Автоматичний збір сміття.
- Універсальна проста у використанні.
- Стандартна бібліотека.
- Монтажні версії.
- Керування делегатами та подіями.

Недоліки [37]:

- Програміст не може робити нічого низького рівня, наприклад, безпосередньо взаємодіяти з обладнанням через драйвери та мікропрограму.
- Не поставляється з незалежним компілятором, який може прямо інтерпретувати максимальні рівні мови для базової апаратної архітектури чистого асемблера. Використовує свій байт-код і компілятор, який значною мірою інтегрований в структуру .Net і являється основою структури .Net як

посередник між машинним кодом замість безпосереднього зв'язку з апаратним забезпеченням.

Враховуючи недоліки та переваги описаних мов програмування, для написання програми інформаційної системи формування розкладів занять за генетичним алгоритмом була обрана мова програмування C#.

3.4 Проєктування структури бази даних інформаційної системи

Створення бази даних є одним із основних етапів при розробці будь-якого програмного продукту. Саме тому для формування таблиць БД необхідно спершу розробити структуру, прослідкувати зв'язки БД та відповідно відтворити їх у програмному середовищі розробки баз даних.

База даних є сукупністю певним чином впорядкованих даних, що зберігаються відповідно до схеми цих даних [38]. Відповідно до поставлених завдань, для реалізації інформаційної системи автоматизованої генерації розкладів занять за генетичним алгоритмом була розроблена та спроектована структура бази даних. Розроблена структура дозволяє організувати проведення ефективних обчислень параметрів множин тестових завдань. Структура розробленої бази даних зображена на рисунку 3.2.

Перейдемо до створення таблиць відповідно до розробленої структури БД. Таблиця «Departments» (таблиця 3.1) призначена для збереження інформації про кафедри, а саме їх назви.

Таблиця 3.1 – Атрибути таблиці «Departments»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Назва кафедри

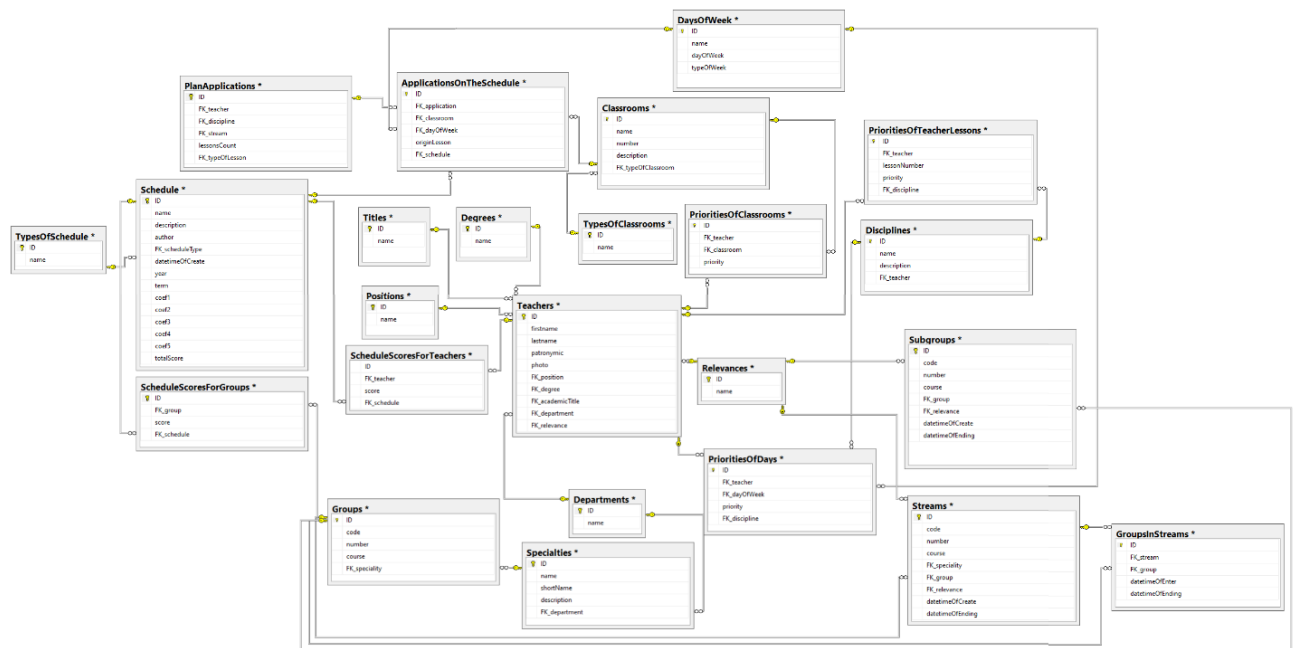


Рисунок 3.2 – Структура бази даних системи генерації розкладів занять за генетичним алгоритмом

Таблиця «Specialties» (таблиця 3.2) призначена для збереження даних про спеціальності, на яких навчаються студенти.

Таблиця 3.2 – Атрибути таблиці «Specialties»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Повна назва спеціальності
3.	shortName	varchar(50)	Короткий запис назви спеціальності
4.	description	varchar(500)	Опис спеціальності
5.	FK_department	int	Вторинний ключ, посилання на запис таблиці «Departments» для співставлення з відповідною кафедрою

Таблиця «DaysOfWeek» (таблиця 3.3) призначена для збереження даних щодо днів тижня, містить поля із назвою, типом та номером тижня.

Таблиця 3.3 – Атрибути таблиці «DaysOfWeek»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Назва дня тижні
3.	dayOfWeek	int	Порядковий номер дня тижня
4.	typeOfWeek	varchar(50)	Тип тижня

Таблиця «Relevances» (таблиця 3.4) призначена для збереження даних про актуальності.

Таблиця 3.4 – Атрибути таблиці «Relevances»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Назва типу актуальності

Таблиця «Groups» (таблиця 3.5) призначена для збереження інформації про групи студентів. Містить поля номер, шифру, курсу групи та спеціальності.

Таблиця 3.5 – Атрибути таблиці «Groups»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	code	varchar(50)	Шифр групи
3.	number	varchar(50)	Номер групи
4.	course	int	Порядковий номер курсу, на якому навчаються студенти
5.	FK_speciality	int	Вторинний ключ, посилання на запис таблиці «Specialities» для співставлення з відповідною спеціальністю

Таблиця «Subgroups» (таблиця 3.6) призначена для збереження інформації щодо підгруп студентів.

Таблиця 3.6 – Атрибути таблиці «Subgroups»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	code	varchar(50)	Шифр групи
3.	number	varchar(50)	Номер групи
4.	course	int	Порядковий номер курсу, на якому навчаються студенти
5.	FK_group	int	Вторинний ключ, посилання на запис таблиці «Groups» для співставлення з відповідною групою
6.	FK_relevance	int	Вторинний ключ, посилання на запис таблиці «Relevances» для співставлення з відповідною актуальністю
7.	datetimeOfCreate	datetime	Дата й час створення підгрупи
8.	datetimeOfEnding	datetime	Дата й час завершення підгрупи

Таблиця «TypesOfClassrooms» (таблиця 3.7) призначена для збереження інформації про типи аудиторій, а саме їх назви.

Таблиця 3.7 – Атрибути таблиці «TypesOfClassrooms»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Назва типу аудиторії

Таблиця «TypesOfLessons» (таблиця 3.8) призначена для збереження інформації про типи занять, а саме їх назви.

Таблиця 3.8 – Атрибути таблиці «TypesOfLessons»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Назва типу заняття

Таблиця «Streams» (таблиця 3.9) призначена для збереження назв потоків груп.

Таблиця 3.9 – Атрибути таблиці «Streams»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	code	varchar(50)	Шифр групи
3.	number	varchar(50)	Номер групи
4.	course	int	Порядковий номер курсу групи
5.	FK_speciality	int	Вторинний ключ, посилання на запис таблиці «Specialities» для співставлення з відповідною спеціальністю
6.	FK_group	int	Вторинний ключ, посилання на запис таблиці «Groups» для співставлення з відповідною групою
7.	FK_relevance	int	Вторинний ключ, посилання на запис таблиці «Relevances» для співставлення з відповідною актуальністю
8.	datetimeOfCreate	datetime	Дата й час створення потоку
9.	datetimeOfEnding	datetime	Дата й час завершення потоку

Таблиця «Classrooms» (таблиця 3.10) призначена для збереження детальної інформації про аудиторії, в яких проводяться заняття.

Таблиця 3.10 – Атрибути таблиці «Classrooms»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Назва навчальної аудиторії
3.	number	varchar(50)	Номер навчальної аудиторії
4.	description	varchar(500)	Опис навчальної аудиторії
5.	FK_typeOfClassroom	int	Вторинний ключ, посилання на запис таблиці «TypesOfClassrooms» для співставлення з відповідним типом групи

Таблиця «Disciplines» (таблиця 3.11) призначена для збереження необхідної інформації про навчальні дисципліни.

Таблиця 3.11 – Атрибути таблиці «Disciplines»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Назва дисципліни
3.	description	varchar(500)	Опис дисципліни
4.	FK_teacher	int	Вторинний ключ, посилання на запис таблиці «Teachers» для співставлення з відповідним викладачем

Таблиця «Teachers» (3.12) створена для збереження інформації щодо викладачів університету.

Таблиця 3.12 – Атрибути таблиці «Teachers»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	firstname	varchar(50)	Ім'я викладача
3.	lastname	varchar(50)	Прізвище викладача
4.	patronymic	varchar(50)	По батькові викладача
5.	photo	varbinary(50)	Фото викладача
6.	FK_position	int	Вторинний ключ, посилання на запис таблиці «Positions» для співставлення з відповідною посадою викладача
7.	FK_degree	int	Вторинний ключ, посилання на запис таблиці «Degrees» для співставлення з відповідним вченим ступенем
8.	FK_academicTitle	int	Вторинний ключ, посилання на запис таблиці «Titles» для співставлення з відповідним вченим званням
9.	FK_department	int	Вторинний ключ, посилання на запис таблиці «Departaments» для співставлення з відповідною кафедрою
10.	FK_relevance	int	Вторинний ключ, посилання на запис таблиці «Relevances» для співставлення з відповідною актуальністю

Таблиця «SubgroupsInStreams» (таблиця 3.13) призначена для збереження інформації про підгрупи в потоках, а саме відповідні групи, відповідні потоки та дати й час створення та завершення їх роботи.

Таблиця 3.13 – Атрибути таблиці «SubgroupsInStreams»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	FK_stream	int	Вторинний ключ, посилання на запис таблиці «Streams» для співставлення з відповідним потоком
3.	FK_group	int	Вторинний ключ, посилання на запис таблиці «Groups» для співставлення з відповідною групою
4.	datetimeOfEnter	datetime	Дата й час створення групи в потоці
5.	datetimeOfEnding	datetime	Дата й час завершення групи в потоці

Таблиця «Degrees» (таблиця 3.14) містить дані про назви вчених ступенів.

Таблиця 3.14 – Атрибути таблиці «Degrees»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Назва вченого ступеня

Таблиця «Titles» (таблиця 3.15) містить дані про назви вчених звань.

Таблиця 3.15 – Атрибути таблиці «Titles»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Назва вченого звання

Таблиця «Positions» (таблиця 3.16) містить дані про назви посад.

Таблиця 3.16 – Атрибути таблиці «Positions»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Назва посади

Таблиця «PlanApplications» (таблиця 3.17) містить інформацію про заявки в план створення розкладів.

Таблиця 3.17 – Атрибути таблиці «PlanApplications»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	FK_teacher	int	Вторинний ключ, посилання на запис таблиці «Teachers» для співставлення з відповідним викладачем
3.	FK_discipline	int	Вторинний ключ, посилання на запис таблиці «Disciplines» для співставлення з відповідною дисципліною
4.	FK_stream	int	Вторинний ключ, посилання на запис таблиці «Streams» для співставлення з відповідним потоком
5.	lessonsCount	int	Кількість занять
6.	FK_typeOfLesson	int	Вторинний ключ, посилання на запис таблиці «TypesOfLessons» для співставлення з відповідним типом заняття

Таблиця «ApplicationsOnTheSchedule» (таблиця 3.18) призначена для збереження інформації про заявки в розкладі.

Таблиця 3.18 – Атрибути таблиці «ApplicationsOnTheSchedule»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	FK_application	int	Вторинний ключ, посилання на запис таблиці «PlanApplications» для співставлення з відповідною заявкою
3.	FK_classroom	int	Вторинний ключ, посилання на запис таблиці «Classrooms» для співставлення з відповідною аудиторією
4.	FK_dayOfWeek	int	Вторинний ключ, посилання на запис таблиці «DaysOfWeek» для співставлення з відповідним днем тижня
5.	originLesson	varchar(50)	Назва початкового заняття
6.	FK_schedule	int	Вторинний ключ, посилання на запис таблиці «Schedule» для співставлення з відповідним розкладом

Таблиця «TypesOfSchedule» (таблиця 3.19) призначена для збереження назв типів розкладу.

Таблиця 3.19 – атрибути таблиці «TypesOfSchedule»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Назва типу розкладу

Таблиця «Schedule» (таблиця 3.20) створена для збереження інформації про існуючі розклади.

Таблиця 3.20 – атрибути таблиці «Schedule»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	name	varchar(50)	Назва розкладу
3.	description	varchar(500)	Опис розкладу
4.	author	varchar(50)	Автор розкладу
5.	FK_scheduleType	int	Вторинний ключ, посилання на запис таблиці «TypesOfSchedule» для співставлення з відповідним типом розкладу
6.	datetimeOfCreate	datetime	Дата й час створення розкладу
7.	year	int	Рік розкладу
8.	term	int	Семестр розкладу
9.	coef1	float	Коефіцієнт 1 для розкладу
10.	coef2	float	Коефіцієнт 2 для розкладу
11.	coef3	float	Коефіцієнт 3 для розкладу
12.	coef4	float	Коефіцієнт 4 для розкладу
13.	coef5	float	Коефіцієнт 5 для розкладу
14.	totalScore	float	Загальна оцінка розкладу

Таблиця «PrioritiesOfTeacherLessons» (таблиця 3.21) створена для збереження інформації про пріоритети пар викладачів.

Таблиця 3.21 – Атрибути таблиці «PrioritiesOfTeacherLessons»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	FK_teacher	int	Вторинний ключ, посилання на запис таблиці «Teachers» для співставлення з відповідним викладачем
3.	lessonNumber	int	Номер заняття в розкладі
4.	priority	int	Пріоритет
5.	FK_discipline	int	Вторинний ключ, посилання на запис таблиці «Disciplines» для співставлення з відповідною дисципліною

Таблиця «PrioritiesOfClassrooms» (таблиця 3.22) створена для збереження даних щодо пріоритетів викладачів по аудиторіях.

Таблиця 3.22 – Атрибути таблиці «PrioritiesOfClassrooms»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	FK_teacher	int	Вторинний ключ, посилення на запис таблиці «Teachers» для співставлення з відповідним викладачем
3.	FK_classroom	int	Вторинний ключ, посилення на запис таблиці «Classrooms» для співставлення з відповідною аудиторією
4.	priority	int	Пріоритет

Таблиця «PrioritiesOfDays» (таблиця 3.23) створена для збереження пріоритетів викладачів щодо дня тижня.

Таблиця 3.23 – атрибути таблиці «PrioritiesOfDays»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	FK_teacher	int	Вторинний ключ, посилення на запис таблиці «Teachers» для співставлення з відповідним викладачем
3.	FK_dayOfWeek	int	Вторинний ключ, посилення на запис таблиці «Teachers» для співставлення з відповідним днем тижня
4.	priority	int	Пріоритет

Таблиця «ScheduleScoresForTeachers» (таблиця 3.24) створена для збереження інформації щодо оцінок розкладу для викладачів.

Таблиця 3.24 – Атрибути таблиці «ScheduleScoresForGroups»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	FK_group	int	Вторинний ключ, посилення на запис таблиці «Groups» для співставлення з відповідною групою
3.	score	float	Загальна оцінка розкладу
4.	FK_schedule	int	Вторинний ключ, посилення на запис таблиці «Schedule» для співставлення з відповідним розкладом

Таблиця «ScheduleScoresForTeachers» (таблиця 3.25) створена для збереження інформації щодо оцінок розкладу для викладачів.

Таблиця 3.25 – Атрибути таблиці «ScheduleScoresForTeachers»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	FK_teacher	int	Вторинний ключ, посилення на запис таблиці «Teachers» для співставлення з відповідним викладачем
3.	score	float	Загальна оцінка розкладу
4.	FK_schedule	int	Вторинний ключ, посилення на запис таблиці «Schedule» для співставлення з відповідним розкладом

Таким чином було створено не лише базу даних, а й її структуру. Заповнили таблиці початковими даними для подальшої роботи тестової інформаційної системи генерації розкладів занять за генетичним алгоритмом.

Висновки до розділу 3

У розділі було описано розроблену інформаційну систему генерації розкладів занять за генетичним алгоритмом, що дозволяє з використанням створених інформаційної моделі розкладу занять, методу побудови розкладів занять за генетичним алгоритмом і математичної моделі обрахунку відповідності розкладу занять вимогам виконувати ручне та автоматизоване створення нових розкладів, формувати нові похідні розклади з використанням генетичного алгоритму, оцінювати сформовані розклади занять з урахуванням вимог викладачів й академічних груп до параметрів проведення пар і визначати кращі зразки розкладів занять за результатом порівняння їх оцінок відповідності вимогам.

Інформаційна система генерації розкладів занять за генетичним алгоритмом включає в себе базу даних, в якій зберігаються наступні сутності й їх властивості: викладачі, аудиторії, типи аудиторій і занять, входження підгруп у потоки, групи, підгрупи, потоки, кафедри як включення викладачів і потоків до аудиторного фонду, дні тижня, вимоги на планування пар (як комбінацій викладача, потоку, дисципліни й типу заняття), заплановані у розкладі занять пари, пріоритети викладачів до пар, пріоритети викладачів до аудиторій, пріоритети викладачів до днів тижнів, оцінки розкладів для груп, оцінки розкладів для викладачів, вагові коефіцієнти для оцінки розкладів.

В розділі було виконано проєктування відповідної структури реляційної бази даних інформаційної системи генерації розкладів занять за генетичним алгоритмом.

Крім БД, інформаційна система генерації розкладів занять за генетичним алгоритмом має підсистеми: підсистема роботи з відомостями бази даних по

розкладах, підсистема користувацького створення зразків розкладів, підсистема роботи з налаштуваннями генерації розкладів та підсистема автоматизованої генерації розкладів. Була наведена прикладна структура складових інформаційної системи генерації розкладів занять продається у вигляді діаграми класів та їх властивостей, яка має забезпечити можливість системи виконувати ручне та автоматизоване створення нових розкладів, формувати нові похідні розклади із використанням генетичного алгоритму, оцінювати сформовані розклади занять із урахуванням вимог викладачів й академічних груп до параметрів проведення пар й визначати кращі зразки розкладів занять за результатом порівняння їх оцінок відповідності вимогам.

Також в розділі було наведено аргументацію вибору рекомендованих програмних засобів для успішної розробки інформаційної системи генерації розкладів занять за генетичним алгоритмом.

Розділ 4

Дослідження ефективності методу побудови розкладів занять за генетичним алгоритмом

4.1 Розробка прикладних програмних компонентів інформаційної системи

Основним завданням роботи системи автоматизованої генерації розкладів занять за генетичним алгоритмом є створенням розкладів занять із врахуванням пріоритетів викладачів та потоків студентів. Для цього в програмі використовуються методи та класи, що описані та проілюстровані вище. Етапи роботи користувача із системою будуть поділятися на наступні етапи: «Викладачі та їх вимоги», «Ресурс аудиторного фонду», «Двотижневий план навчання», «Академічні групи і потоки», «Ручне створення розкладів та генератор розкладів», «Параметри роботи генетичного алгоритму» та «Автоматизована еволюція розкладів», що є основним у роботі.

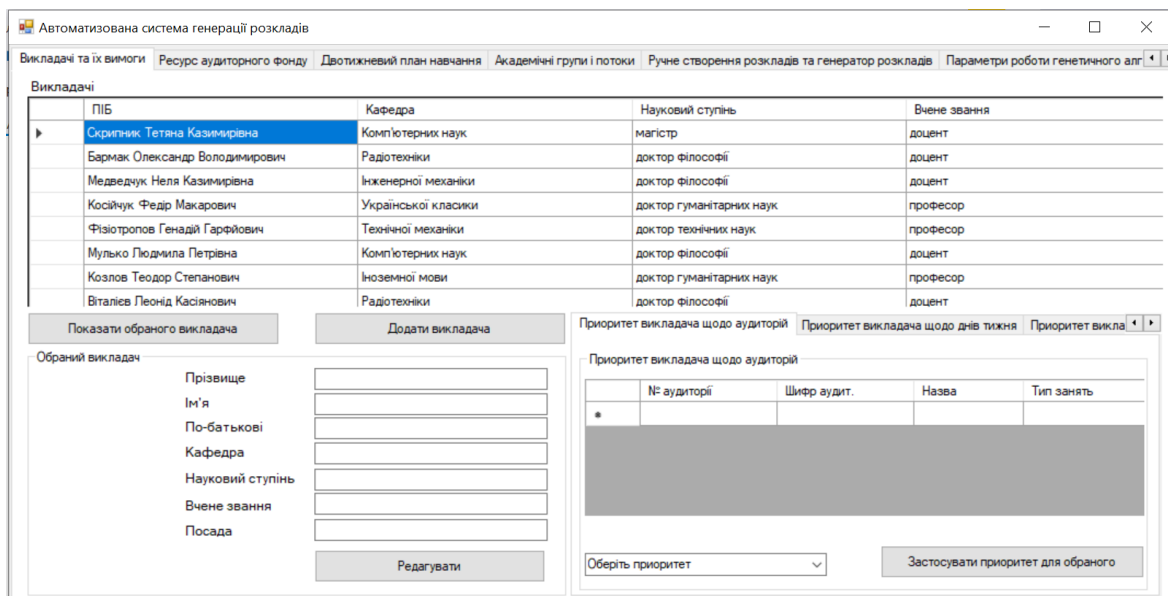


Рисунок 4.1 – Вкладка «Викладачі та їх вимоги»

Перед початком роботи, нам потрібно завантажити у базу даних необхідні матеріали, котрі необхідно дослідити. Після цього можемо взаємодіяти із

вкладками на формі. Вкладка «Викладачі та їх вимоги» призначена для вибору необхідної інформації з бази даних, натиснувши на поле необхідного викладача та кнопку «Показати обраного викладача» (рисунок 4.1).

Фрагмент програмного коду, що відповідає за реалізацію завантаження інформації про викладачів наведено нижче:

```
private void LoadData()
{
    string connectionString = "Server=.\SQLEXPRESS;Initial
Catalog=TimeTable_1;Integrated Security=true;";
    SqlConnection myConnection = new SqlConnection(connectionString);
    myConnection.Open();

    string query = "SELECT * FROM Teachers ORDER BY ID";
    SqlCommand command = new SqlCommand(query, myConnection);

    SqlDataReader reader = command.ExecuteReader();

    List<string[]> dataa = new List<string[]>();

    while (reader.Read())
    {
        dataa.Add(new string[6]);
        dataa[dataa.Count - 1][0] = reader[0].ToString();
        dataa[dataa.Count - 1][1] = reader[1].ToString();
        dataa[dataa.Count - 1][2] = reader[2].ToString();
        dataa[dataa.Count - 1][3] = reader[3].ToString();
        dataa[dataa.Count - 1][4] = reader[4].ToString();
        dataa[dataa.Count - 1][5] = reader[5].ToString();
    }
    reader.Close();
    myConnection.Close();

    foreach (string[] s in dataa)
        dataGridView1.Rows.Add(s);
}
```

Дані, котрі необхідно проаналізувати, мають завантажуватись із записом відповідних полів, а саме: «ПІБ», «Кафедра», «Науковий ступінь», «Вчене звання» та «Посада». Уся інформація, яка завантажується із бази даних, відповідно заповнює всі поля таблиці. Також була створена підкладка з пріоритетами викладача щодо аудиторій (рисунок 4.2), днів тижня (рисунок 4.3) та пар (рисунок 4.4), у яких викладач може визначити пріоритетність вищевказаних аспектів. Спершу необхідно завантажити дані із бази даних для представлення у вигляді таблиці.

Приоритет викладача щодо аудиторій Приоритет викладача щодо днів тижня Приоритет викла ◀ ▶

Приоритет викладача щодо аудиторій

	№ аудиторії	Шифр аудит.	Назва	Тип занять
▶	2	1	3-403	Лекц.ауд
	3	2	3-406	Лаб.ауд
	4	3	3-404	Лаб.ауд.
*				

Оберіть пріоритет ▼ Застосувати пріоритет для обраного

Рисунок 4.2 – Вкладка для вибору пріоритетів викладача щодо аудиторій

Приоритет викладача щодо аудиторій Приоритет викладача щодо днів тижня Приоритет викла ◀ ▶

Приоритет викладача щодо днів тижня

	№	День тижня	Пріоритет
*			

Оберіть пріоритет ▼ Застосувати пріоритет для обраного

Рисунок 4.3 – Вкладка для вибору пріоритетів викладача днів тижня

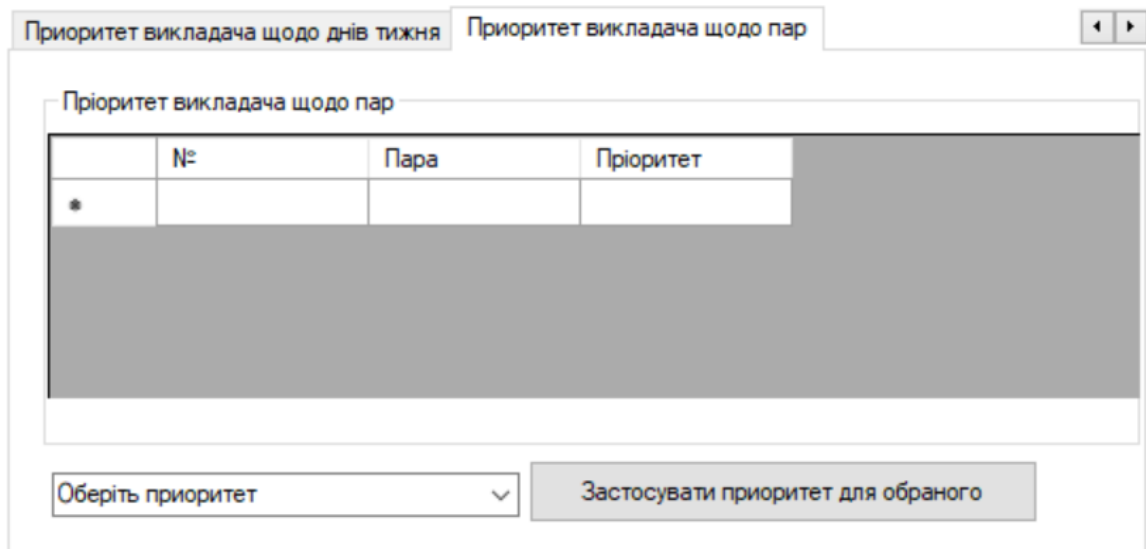


Рисунок 4.4 – Вкладка для вибору пріоритетів викладача щодо пар

Фрагмент програмного коду методу, що завантажує дані до таблиць наведено нижче:

```
private void LoadPrior()
{
    string connectionString = "Server=.\SQLEXPRESS;Initial
Catalog=TimeTable_1;Integrated Security=true;";
    SqlConnection myConnection = new SqlConnection(connectionString);
    myConnection.Open();

    string query = "SELECT * FROM prioritAud ORDER BY ID";
    SqlCommand command = new SqlCommand(query, myConnection);

    SqlDataReader reader = command.ExecuteReader();

    List<string[]> dataa = new List<string[]>();

    while (reader.Read())
    {
        dataa.Add(new string[4]);
        dataa[dataa.Count - 1][0] = reader[0].ToString();
        dataa[dataa.Count - 1][1] = reader[1].ToString();
        dataa[dataa.Count - 1][2] = reader[2].ToString();
        dataa[dataa.Count - 1][3] = reader[3].ToString();
    }
    reader.Close();
    myConnection.Close();

    foreach (string[] s in dataa)
        dataGridView4.Rows.Add(s);
}
```

Для виконання наступних операцій користувачеві необхідно натиснути кнопку «Застосувати пріоритет для обраного», після чого відповідні методи класу обчислюють необхідні параметри для подальшої роботи із застосунком.

Перейдемо до реалізації роботи наступної вкладки «Ресурс аудиторного фонду». Дана вкладка відображає перелік усіх доступних аудиторій, а також їх шифр, назву, тип, доступність та актуальність (рисунок 4.5).

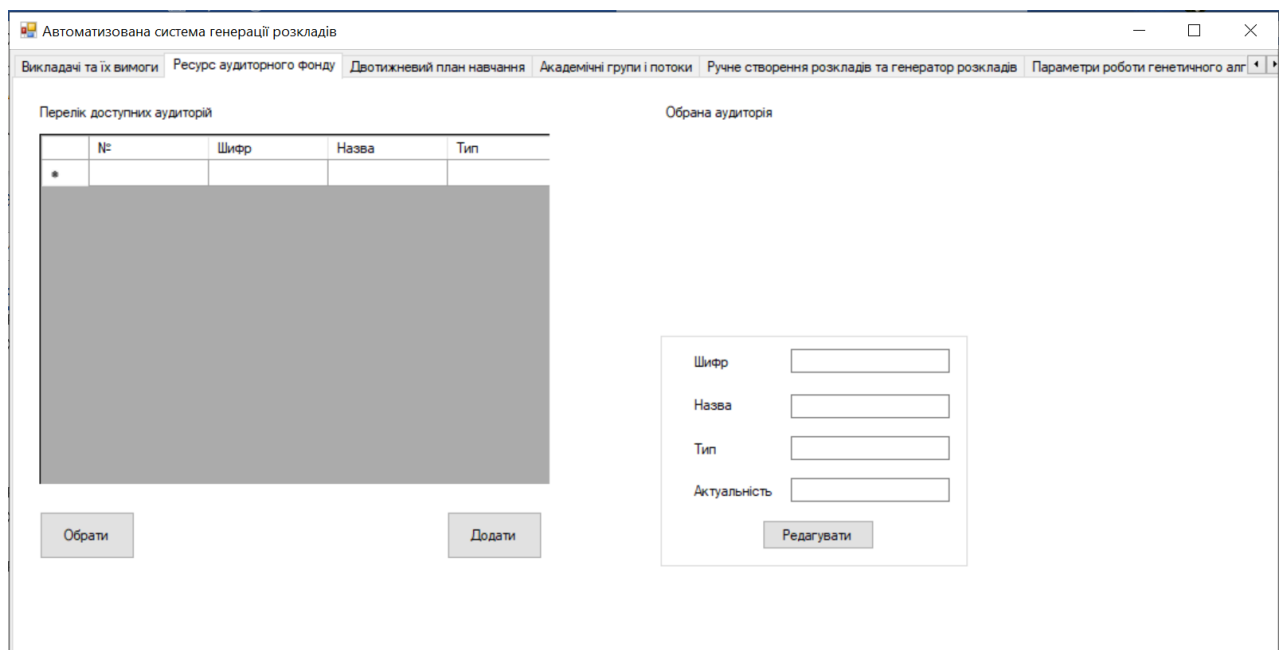


Рисунок 4.5 – Вкладка «Ресурс аудиторного фонду»

Призначення вкладки полягає у спрощенні отримання та обробки інформації щодо аудиторій викладачем, який може додати, редагувати або переглянути будь-яку аудиторію та детальну інформацію про неї, натиснувши відповідні кнопки. Також існує метод, що дозволяє редагувати інформацію про обрану аудиторію, фрагмент програмної реалізації наведено нижче:

```
private void EditClasses()
{
    c = new Classes(textBox1.Text, textBox2.Text, textBox3.Text, textBox4.Text)
    this._saveCallbackClasses(c, true);
}
```

Наступною вкладкою є «Академічні групи і потоки» (рисунок 4.6).

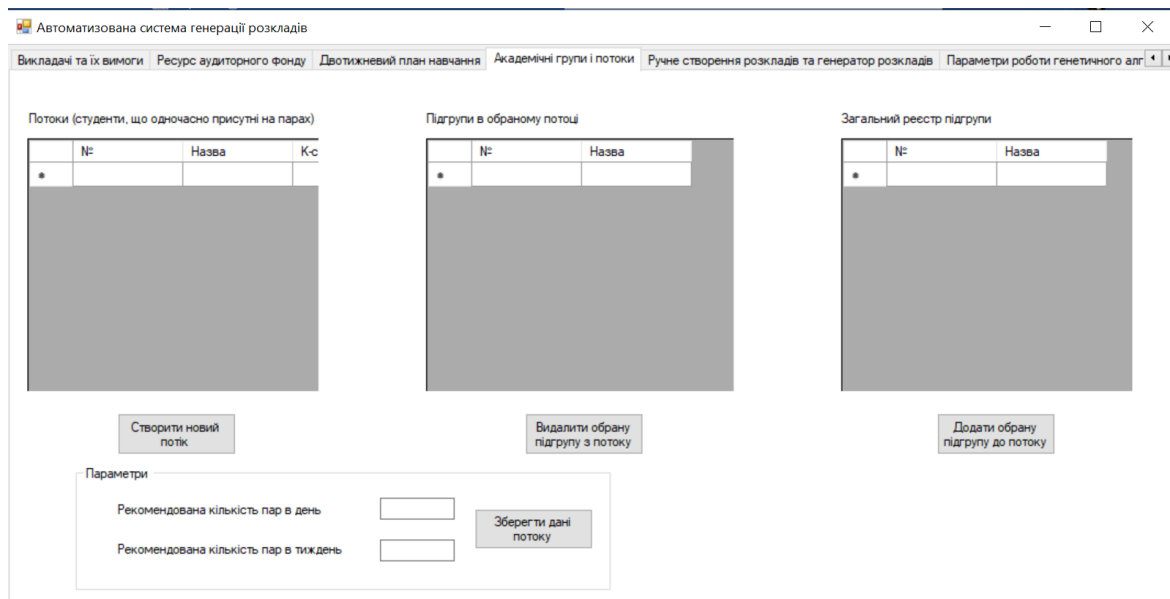


Рисунок 4.6 – Вкладка «Академічні групи і потоки»

Програмний код, що завантажує дані про групи та потоки у відповідні таблиці наведено нижче:

```
private void LoadStreams()
{
    string connectionString = "Server=.\SQLEXPRESS;Initial
Catalog=TimeTable_1;Integrated Security=true;";
    SqlConnection myConnection = new SqlConnection(connectionString);
    myConnection.Open();

    string query = "SELECT * FROM Streams ORDER BY ID";
    SqlCommand command = new SqlCommand(query, myConnection);

    SqlDataReader reader = command.ExecuteReader();

    List<string[]> dataa = new List<string[]>();

    while (reader.Read())
    {
        dataa.Add(new string[3]);
        dataa[dataa.Count - 1][0] = reader[0].ToString();
        dataa[dataa.Count - 1][1] = reader[1].ToString();
        dataa[dataa.Count - 1][2] = reader[2].ToString();
    }
    reader.Close();
    myConnection.Close();

    foreach (string[] s in dataa)
        dataGridView6.Rows.Add(s);
}
```

Функціонал цієї вкладки представляє можливість створення та редагування розкладу, в який вноситься інформація про потоки, заплановані на

відвідування пари. Сам потік може складатись з низки підгруп, для чого було створено окрему таблицю з відповідною назвою.

Також додано регульований параметр рекомендованої кількості пар в день/тиждень з відповідними кнопками (рисунок 4.7).

Рисунок 4.7 – Модуль параметру регулювання рекомендованої кількості пар.

Фрагмент програмного коду методу, що зберігає параметри на вкладці «Академічні групи та потоки» наведено нижче:

```
private void SaveParametersStreams()
{
    s = new Streams(textBox1.Text, textBox2.Text);
    this._SaveParametersStreams(s, true);
}
```

Рисунок 4.8 – Вкладка «Ручне створення розкладів та генератор розкладів»

Наступною вкладкою застосунку є «Ручне створення розкладів та генератор розкладів». Дана вкладка призначена перегляду, мануального створення, редагування, та інших дій з розкладом (рисунок 4.8).

Для початку, за допомогою таблиці «Версії розкладу до плану» користувач вибирає один з наведених розкладів, інформація про який наведена у вигляді двох стовпців: Номер плану та Назва плану. Натискаючи на кнопку «Вивести пари з обраного розкладу», виводиться повний опис вибраного ним елементу, а саме: «День тижня», «№Пари», «Аудиторія», «Дисципліна», «Поток», «Викладач», «Тип заняття» та «Неперервна кількість пар». Також є можливість редагувати, видалити, або згенерувати розклад по плану автоматично, натиснувши відповідні кнопки нижче (рисунок 4.9).

Розклад пар

	№	День тижня	№пари	Аудиторія	Дисципліна	Поток	Викладач	Тип заняття	Неперервна к-сть пар
»*									
[Головна таблиця розкладу]									

Розклад:

Рисунок 4.9 – Кнопки взаємодії з головною таблицею

Окрім того, до вкладки додано окремий модуль, який дозволяє відокремлено додати будь-яку пару до розкладу, та детально її описати за

допомогою низки полів: «Навчальна позиція», «День тижня», «Номер пари», «Аудиторія», «Поток», «Викладач» та «Тип заняття» (рисунок 4.10).

Додати пару до розкладу

Навчальна позиція

День тижня

№ пари

Аудиторія

Дисципліна

Поток

Викладач

Додати пару до розкладу

Рисунок 4.10 – Модуль «Додати пару до розкладу»

Передостання вкладка застосунку носить назву «Параметри роботи генетичного алгоритму» (рисунок 4.11), у якій задаються вхідні параметри для роботи генетичного алгоритму. Вкладка має 3 групи елементів, а саме:

- Параметри роботи генетичного алгоритму
- Обмеження на роботу алгоритму
- Критерії оцінки розкладів

Група «Параметри роботи генетичного алгоритму» має 2 функції:

- Дозвіл на наскрізну мутацію
- Дозвіл на точкову мутацію

Слугує для можливості дозволити користувачеві чи заборонити наскрізну та точкову мутацію при генерації розкладу.

Група «Обмеження на роботу алгоритму» має 4 функції:

- Максимальна кількість пар на день у викладача
- Максимальна кількість пар на тиждень у викладача
- Максимальна кількість пар на день у підгрупи
- Максимальна кількість пар на тиждень у підгрупи

Використовується для визначення пріоритетності кожного вхідного параметру при генерації розкладу.

Група «Критерії оцінки розкладів» має також 4 функції, а саме:

- Важливість врахування пріоритетів викладачів (аудиторії, дні тижня, тощо)
- Важливість врахування пріоритетів потоків (аудиторії, дні тижня, тощо)
- Важливість відсутності вікон у викладача
- Важливість відсутності вікон у підгрупи

Надає можливість визначення пріоритетності вхідних даних.

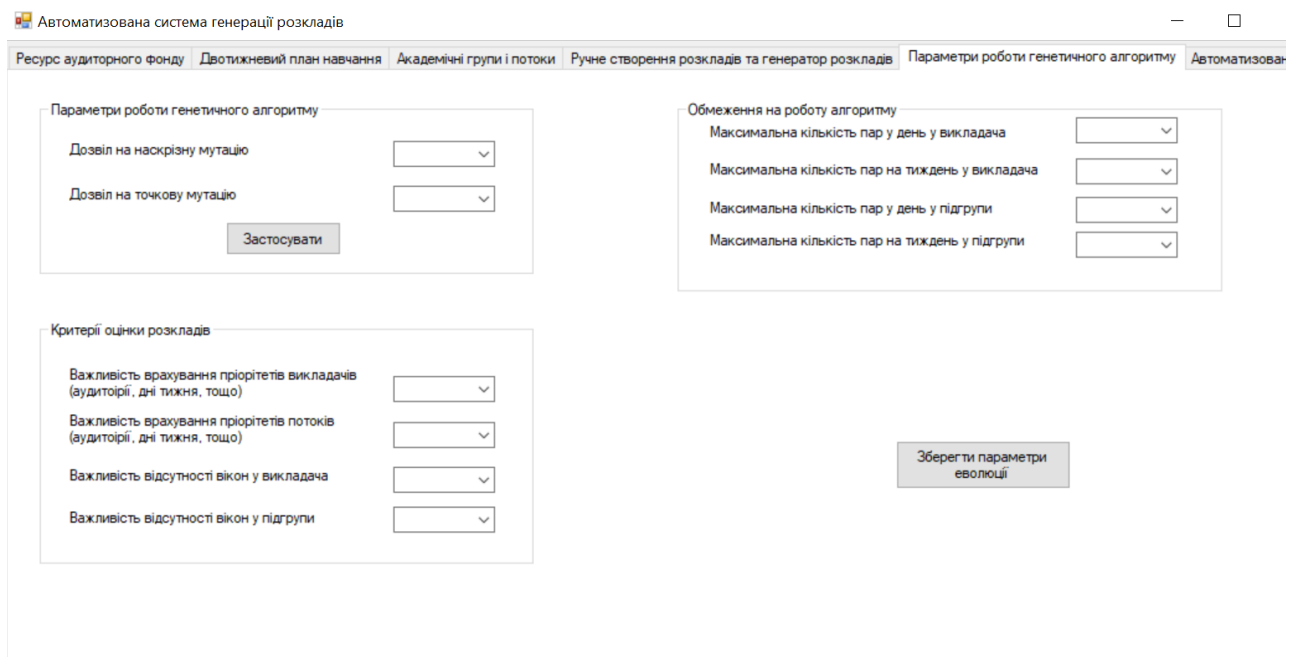


Рисунок 4.11 – Вкладка «Параметри роботи генетичного алгоритму»

Остання вкладка застосунку «Автоматизована еволюція розкладів» відображає поточну назву і сформовану оцінку розкладу. Також дозволяє переглянути його і згенерувати новий шляхом використання генетичного алгоритму (рисунок 4.12).

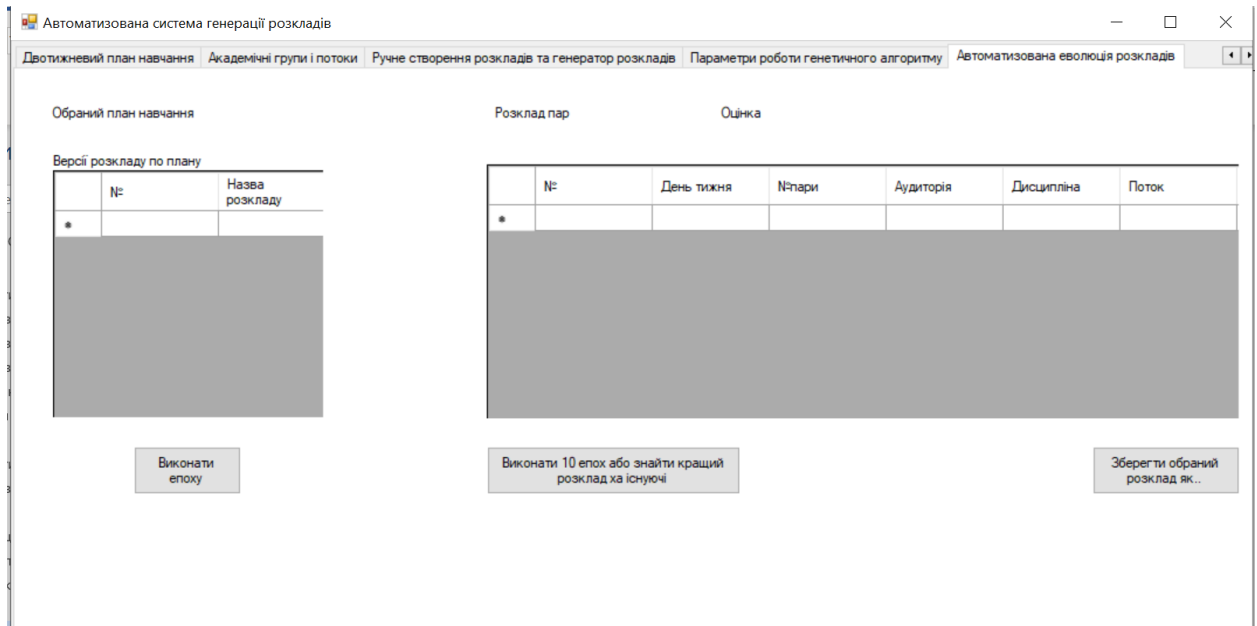


Рисунок 4.12 – Вкладка «Автоматизована еволюція розкладів»

На вкладці наявні 2 таблиці:

– Версії розкладу по плану

Показує назву розкладу та оцінку, та дозволяє виконати епоху по вибраному розкладу.

– Розклад пар

У таблиці наведена детальна інформація про розклад, а саме: «День тижня», «№Пари», «Аудиторія», «Дисципліна», «Поток», «Викладач», «Тип заняття» та «Неперервна кількість пар». Функціонал таблиці дозволяє виконати одразу 10 епох по вибраному розкладу, або вибрати найкращий з існуючих. Для завершення операцій користувач має натиснути кнопку «Зберегти обраний розклад як...» у разі задовільного результату.

Таким чином, було реалізовано відповідні модулі та складові програми, що повністю відповідають поставленій задачі, а саме створенню технології формування розкладів занять за генетичним алгоритмом. Інтерфейс програми створено максимально простим та зрозумілим, а методи та класи програми повністю забезпечують надійне функціонування програмного продукту.

4.2 Тестування інформаційної системи

Задля того, щоб переконатись у правильності роботи інформаційної системи формування розкладів занять за генетичним алгоритмом, проведемо тестування системи. Для цього було створено ряд тест кейсів (тестові випадки). В першому тестовому випадку перевіriamo роботу методів, що завантажують відомості про викладачів до окремих таблиць та перевіriamo роботу методів, що передають значення обраних викладачами пріоритетів щодо аудиторій, днів тижня та пар (рисунок 4.11).

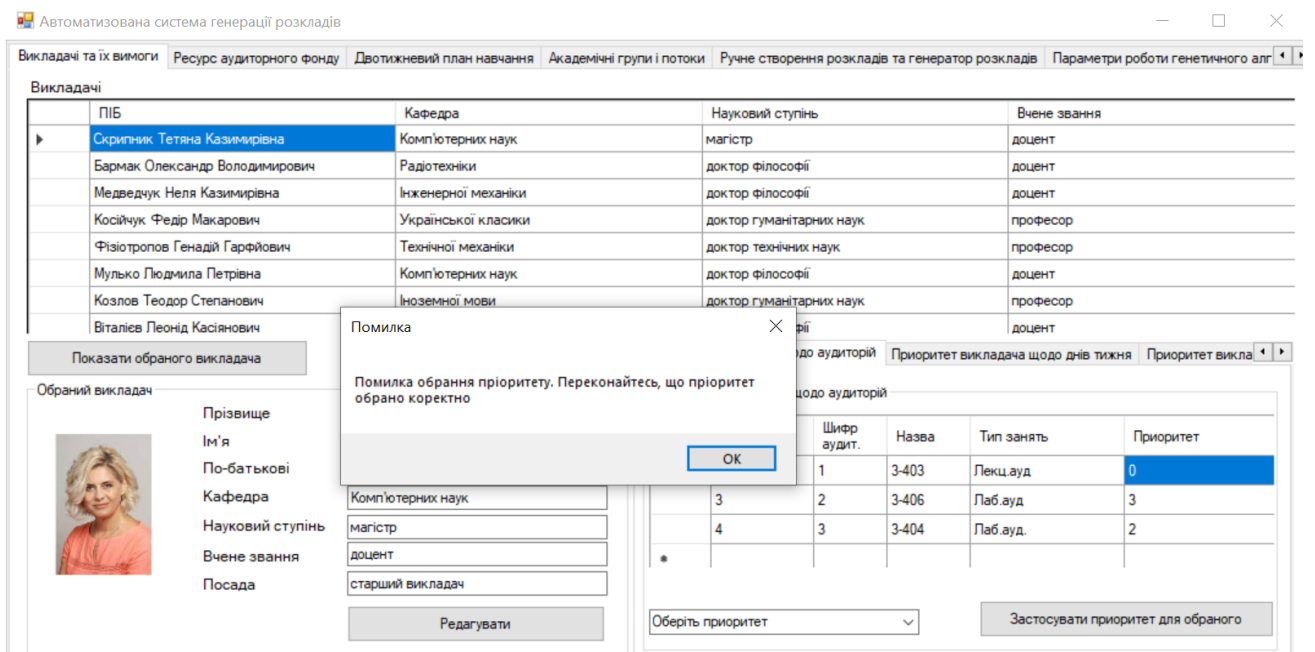


Рисунок 4.11 – Тестування некоректного введення пріоритету

За вказаною помилкою сформовано наступний тест-кейс (таблиця 4.1).

Наступний тест-кейс призначений для тестування роботи попередження в разі некоректного вибору параметрів роботи генетичного алгоритму (рисунок 4.12).

Таблиця 4.1 – Тест-кейс АТ0001

Тест-кейс ID: АТ0001	Пріоритет: 3	Створено: 29.11.2021, В.В. Богач
Назва: Тестування передачі даних методами Вхідні дані: дані з бази даних TimeTable.dbo		
Кроки	Очікуваний результат	
<p><i>Передумова:</i> користувач на вкладці «Викладачі та їх вимоги» має обрати рядок в таблиці, що відповідає за певного викладача та обрати пріоритет аудиторії на підвкладці «Пріоритет викладача щодо аудиторій» ввести значення «0», що є некоректним.</p> <ol style="list-style-type: none"> 1. Запустити додаток 2. Перейти на вкладку «Викладачі та їх вимоги» 3. Натиснути на відповідний запис викладачем (1 рядок) 4. На вкладці «Пріоритети викладачів щодо аудиторій» ввести значення «0» в полі «Пріоритет» 5. Порівняти фактичний результат з очікуваним 6. Закрити діалог із помилкою 7. Натиснути на відповідний запис викладачем (1 рядок) 8. На вкладці «Пріоритети викладачів щодо аудиторій» ввести значення від 1 до 5 в полі «Пріоритет» 9. Порівняти фактичний результат з очікуваним 	<p>Діалог із помилкою: «Помилка обрання пріоритету. Переконайтесь, що пріоритет обрано коректно».</p> <p>Значення пріоритету оброблено системою.</p>	
Результат виконання тест-кейсу: пройдено успішно		

За вказаною помилкою сформовано наступний тест-кейс (таблиця 4.2). Реалізоване попередження про необхідність заповнити усі поля форми забезпечує надійну та коректну роботу програми, мінімізацію помилок при обчисленнях, та як результат – кращу якість сформованих розкладів.

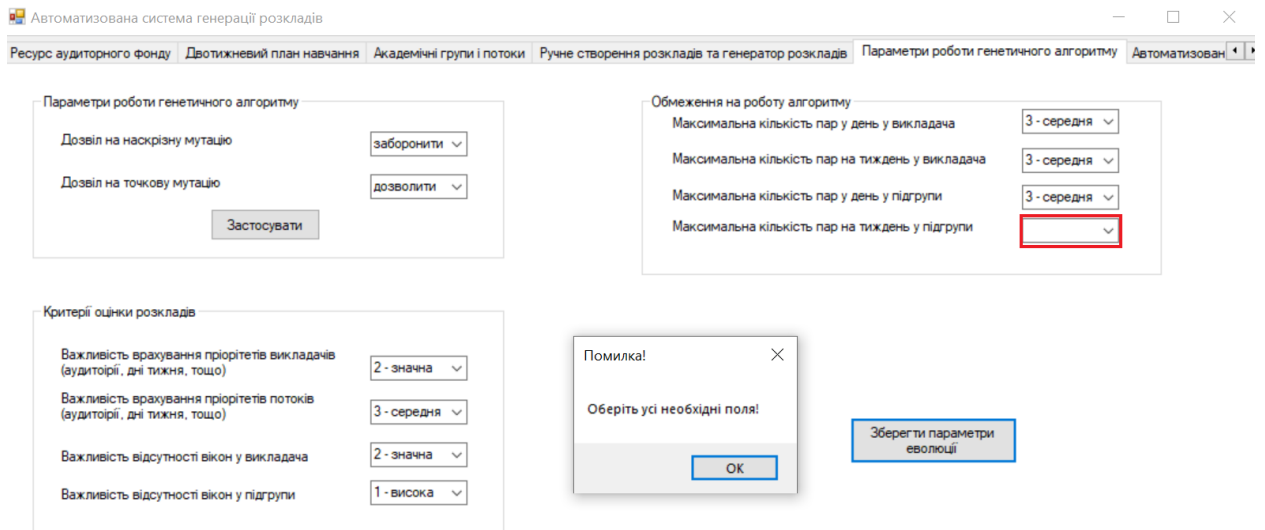


Рисунок 4.12 – Результат перевірки коректності заповнення форми

Таблиця 4.2– Тест-кейс АТ0002

Тест-кейс ID: АТ0001	Пріоритет: 2	Створено: 29.11.2021, В.В. Богач
Назва: Тестування функції завантаження існуючого начального плану Вхідні дані: comboBox-и з вкладки «Параметри роботи генетичного алгоритму»		
Кроки	Очікуваний результат	
<p><i>Передумова:</i> користувач на вкладці «Параметри роботи генетичного алгоритму» має залишити одне або декілька полів порожніми, що є некоректним варіантом задання параметрів.</p> <ol style="list-style-type: none"> Запустити додаток Перейти на вкладку «Параметри роботи генетичного алгоритму» Залишити поле із групи «Обмеження на роботу алгоритму» незаповненим Порівняти фактичний результат з очікуваним Закрити діалог із помилкою Перейти на вкладку «Параметри роботи генетичного алгоритму» Залишити поле із групи «Обмеження на роботу алгоритму» заповненим Порівняти фактичний результат з очікуваним 	<p>Діалог із помилкою: «Помилка. Оберіть усі необхідні поля».</p> <p>Значення параметру оброблено системою.</p>	
Результат виконання тест-кейсу: пройдено успішно		

Наступне тестування призначене для перевірки правильності завантаження даних із диску користувача. Завантажуємо існуючі навчальні плани із носія за допомогою методу, що прикріплений до кнопки «Завантажити існуючий навчальний план» на вкладці «Двотижневий план навчання» (рисунок 4.13, 4.14).

За вказаною помилкою сформовано наступний тест-кейс (таблиця 4.3). Розроблене попередження про помилку завантаження даних убезпечує користувача від завантаження некоректних даних, що в свою чергу знижує ризик виникнення збою при генерації розкладу та наступних обчисленнях.

Таблиця 4.3 – Тест-кейс АТ0003

Тест-кейс ID: АТ0001	Пріоритет: 1	Створено: 29.11.2021, В.В. Богач
Назва: Тестування роботи попередження помилки заповнення полів Вхідні дані: порожня папка на робочому столі		
Кроки	Очікуваний результат	
<p><i>Передумова:</i> користувач на вкладці «Двотижневий план навчання» має натиснути кнопку «Завантажити існуючий план навчання» та обрати папку, що є порожньою, тобто навчального плану не існує.</p> <ol style="list-style-type: none"> 9. Запустити додаток 10. Перейти на вкладку «Двотижневий план навчання» 11. Натиснути кнопку «Завантажити існуючий план навчання» 12. Обрати порожню папку на робочому столі для тестування 13. Порівняти фактичний результат з очікуваним 14. Закрити діалог із помилкою 15. Перейти на вкладку «Двотижневий план навчання» 16. Натиснути кнопку «Завантажити існуючий план навчання» 17. Обрати папку із існуючим навчальним планом 18. Порівняти фактичний результат з очікуваним 	<p>Діалог із помилкою: «Помилка завантаження даних».</p> <p>Файл успішно завантажено</p>	
Результат виконання тест-кейсу: пройдено успішно		

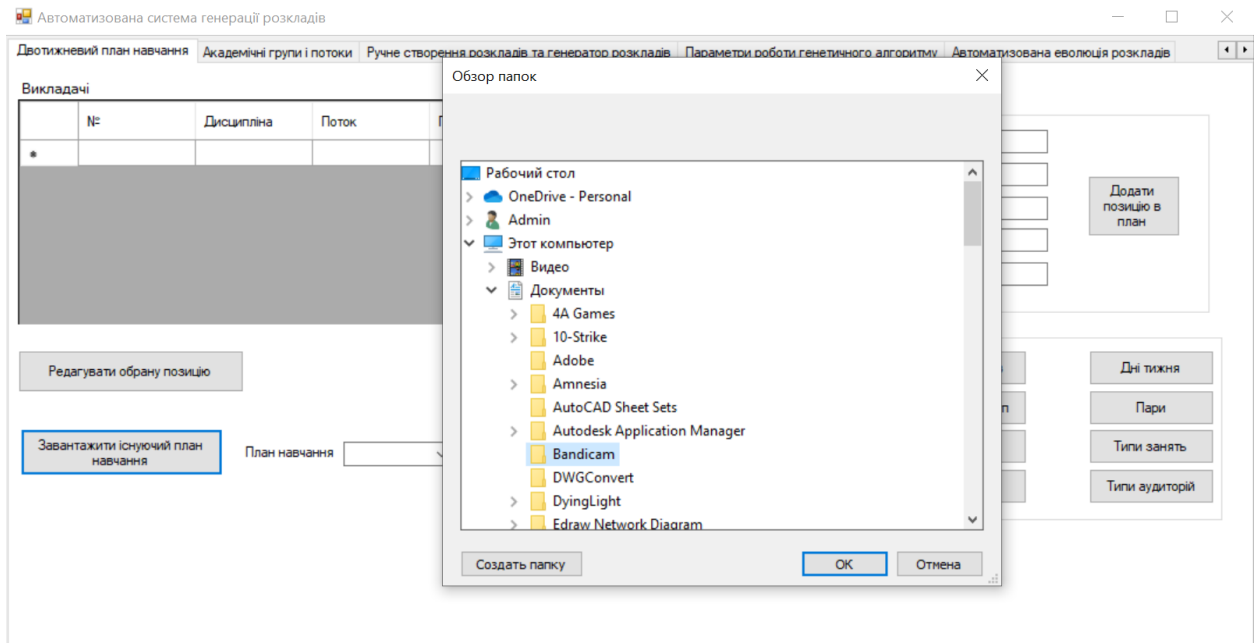


Рисунок 4.13 – Обираємо порожню папку при завантаженні навчального плану для проведення тестування

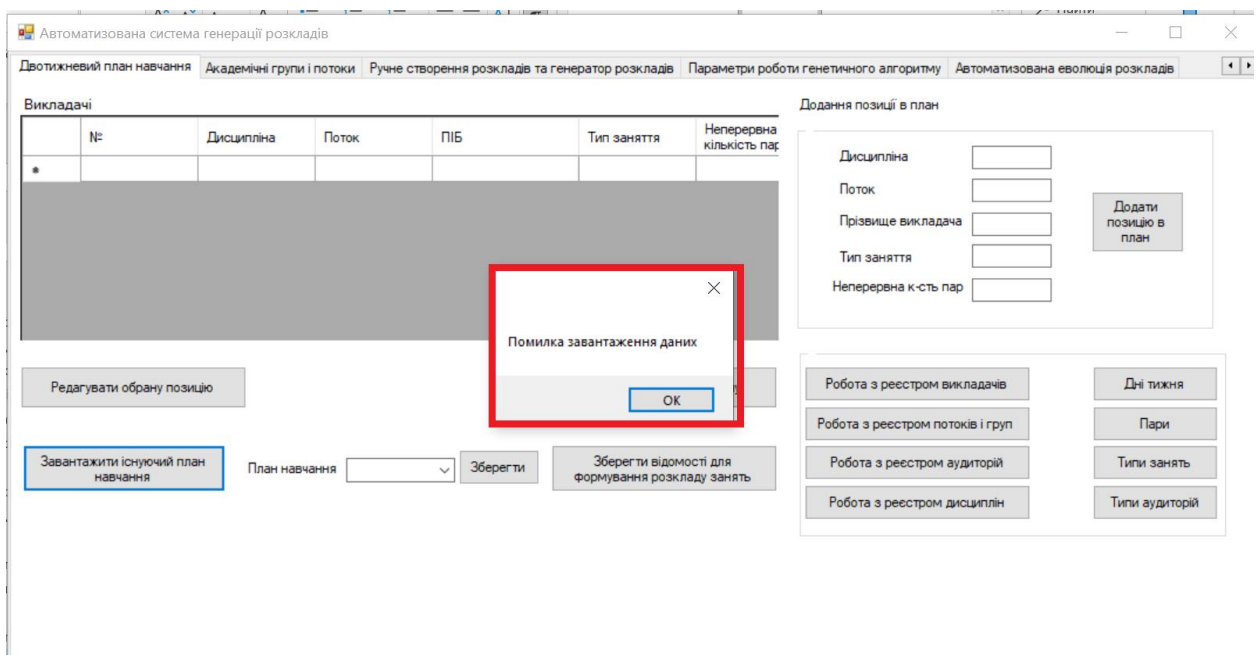


Рисунок 4.14 – Повідомлення про помилку завантаження навчального плану

Отже, тестування коректності виконання функцій інформаційної системи автоматизованої генерації розкладів занять за генетичним алгоритмом було успішно пройдено, заявлений в п.2.4 функціонал реалізовано.

4.3 Функціональне дослідження інформаційної системи генерації розкладів занять за генетичним алгоритмом

Інформаційна система генерації розкладів занять за генетичним алгоритмом включає в себе базу даних, а також підсистеми: підсистема роботи з відомостями бази даних по розкладах, підсистема користувацького створення зразків розкладів, підсистема роботи з налаштуваннями генерації розкладів та підсистема автоматизованої генерації розкладів. Кожна з підсистем має виконувати функції відповідно до поданого в п.3.1 їх розподілу.

Після того, як користувач запускає програму, перше вікно, з яким він працює – «Вибір викладача». Саме тут необхідно обрати перші дії для подальшого створення розкладу. Обираємо поле із відповідним викладачем на натискаємо кнопку «Показати обраного викладача». Після цього його дані автоматично завантажуються у секцію «Обраний викладач» (рисунок 4.15).

Автоматизована система генерації розкладів

Викладачі та їх вимоги | Ресурс аудиторного фонду | Двотижневий план навчання | Академічні групи і потоки | Ручне створення розкладів та генератор розкладів | Параметри роботи генетичного алг

Викладачі	ПІБ	Кафедра	Науковий ступінь	Вчене звання
▶	Скрипник Тетяна Казимирівна	Комп'ютерних наук	магістр	доцент
	Бармак Олександр Володимирович	Радіотехніки	доктор філософії	доцент
	Медведчук Неля Казимирівна	Інженерної механіки	доктор філософії	доцент
	Косійчук Федір Макарович	Української класики	доктор гуманітарних наук	професор
	Фізютропов Геннадій Гарфйович	Технічної механіки	доктор технічних наук	професор
	Мулько Людмила Петрівна	Комп'ютерних наук	доктор філософії	доцент
	Козлов Теодор Степанович	Іноземної мови	доктор гуманітарних наук	професор
	Віталієв Леонід Касіанович	Радіотехніки	доктор філософії	доцент

Показати обраного викладача | Додати викладача

Обраний викладач

Прізвище: Скрипник
 Ім'я: Тетяна
 По-батькові: Казимирівна
 Кафедра: Комп'ютерних наук
 Науковий ступінь: магістр
 Вчене звання: доцент
 Посада: старший викладач

Редагувати

Пріоритет викладача щодо аудиторій | Пріоритет викладача щодо днів тижня | Пріоритет викла

Пріоритет викладача щодо аудиторій				
	№ аудиторії	Шифр аудит.	Назва	Тип занять
▶	2	1	3-403	Лекц. ауд.
	3	2	3-406	Лаб. ауд.
	4	3	3-404	Лаб. ауд.
*				

Оберіть пріоритет | Застосувати пріоритет для обраного

Рисунок 4.15 – Результат роботи вкладки «Вибір викладача»

На підвкладці «Пріоритети викладача щодо..» вводимо необхідну для подальших операцій інформацію, а саме пріоритети викладача щодо аудиторій, днів тижня та пар (рисунок 4.16).

Приоритет викладача щодо аудиторій Приоритет викладача щодо днів тижня Приоритет викла

Приоритет викладача щодо аудиторій

	№ аудиторії	Шифр аудит.	Назва	Тип занять	Приоритет
▶	2	1	3-403	Лекц. ауд	1
	3	2	3-406	Лаб. ауд	3
	4	3	3-404	Лаб. ауд.	2
	2	1	3-403	Лекц. ауд	4

3 - Середня Застосувати пріоритет для обраного

Приоритет викладача щодо аудиторій Приоритет викладача щодо днів тижня Приоритет викла

Приоритет викладача щодо днів тижня

	№	День тижня	Пріоритет
▶	1	понеділок	3
	2	субота	1
	3	вівторок	2
	*		

1 - Висока Застосувати пріоритет для обраного

Приоритет викладача щодо днів тижня Приоритет викладача щодо пар

Приоритет викладача щодо пар

	№	Пара	Пріоритет
	1	2	1
▶	4	6	2
	2	3	4
	3	4	5

2 - Значна Застосувати пріоритет для обраного

Рисунок 4.16 – Обираємо пріоритети викладача щодо аудиторій, днів тижня та пар

Опрацювавши вкладку «Викладачі та їх вимоги» переходимо до наступної – «Ресурс аудиторного фонду» (рисунок 4.17). саме тут користувач може переглянути, редагувати та додати до переліку нову аудиторію, в якій проводитимуться заняття. Розглянемо, як переглянути кабінет: для цього необхідно обрати поле відповідної аудиторії та натиснути «Обрати».

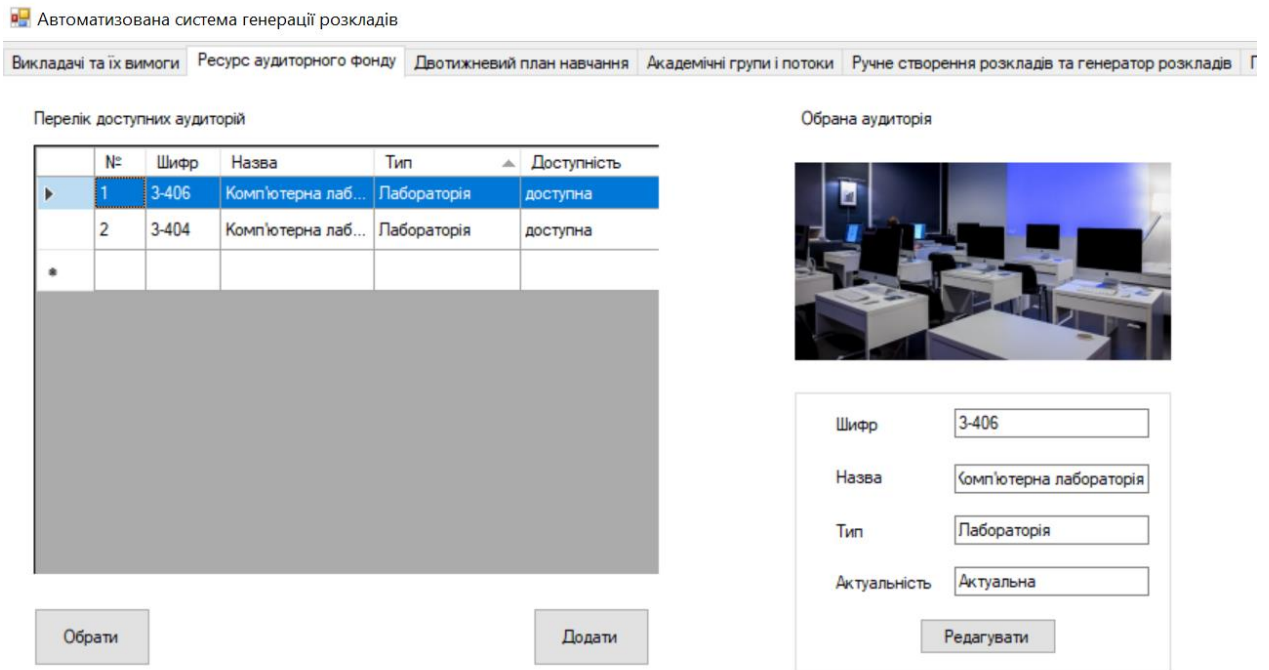


Рисунок 4.17 – Перегляд аудиторії та відомостей про неї

Перейдемо до наступного розділу програми «Двотижневий план навчання». Тут користувач може завантажити існуючий навчальний план. Для цього необхідно натиснути кнопку «Завантажити існуючий навчальний план» та обрати необхідний об'єкт із файлової системи (рисунок 4.18).

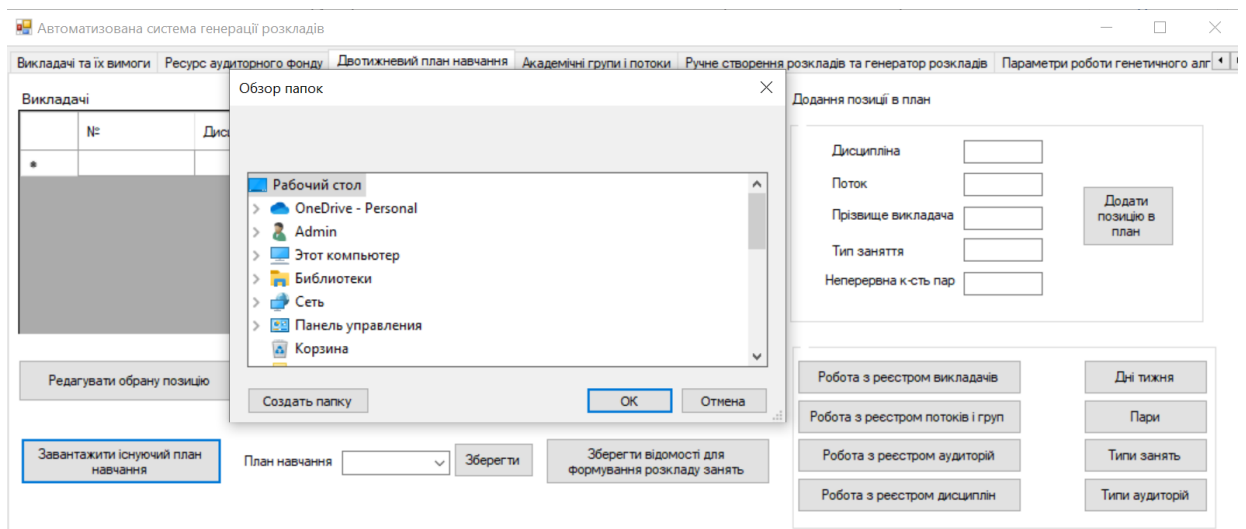


Рисунок 4.18 – Завантаження існуючого навчального плану

Далі переходимо до розділу «Академічні групи та потоки», де необхідно сформувати потоки груп для проведення наступних занять (рисунок 4.19).

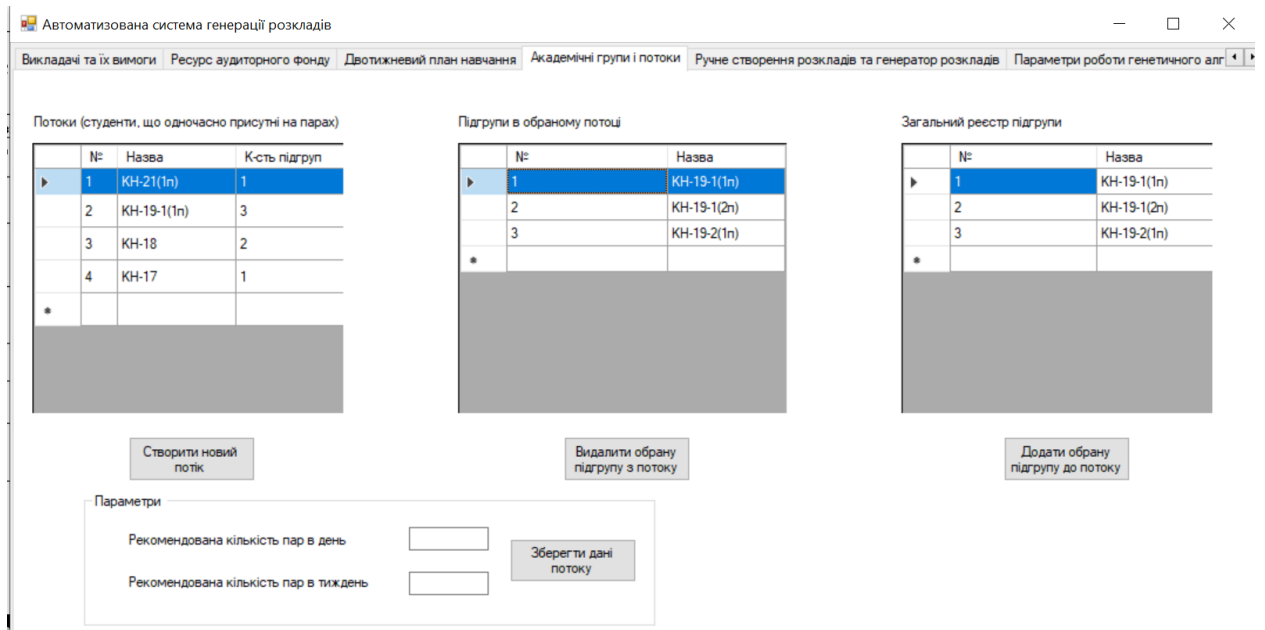


Рисунок 4.19 – Формування потоків груп

Наступним кроком є перехід до вкладки «Двотижневий план навчання» (рисунок 4.20).

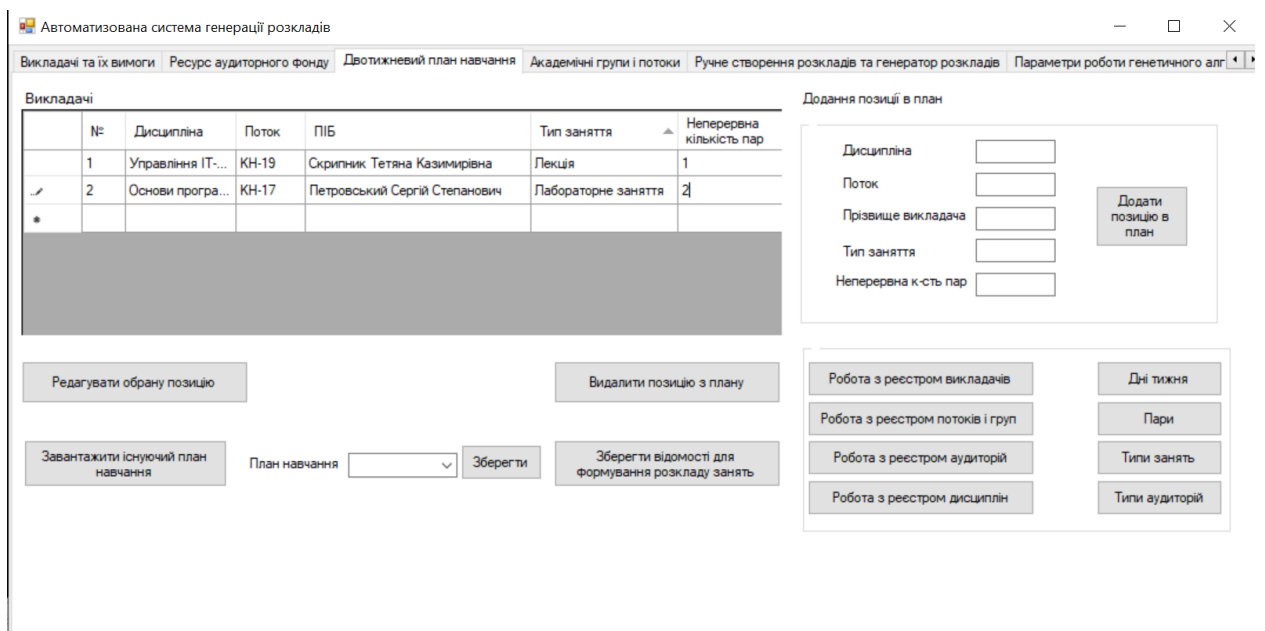


Рисунок 4.20 – Вкладка «Двотижневий план навчання»

На даній вкладці оберемо план навчання потоків та додаймо ще одну позицію, натиснувши на кнопку «Додати» та заповнивши всі поля перед цим у секції «Додання позиції в план» (рисунок 4.21).

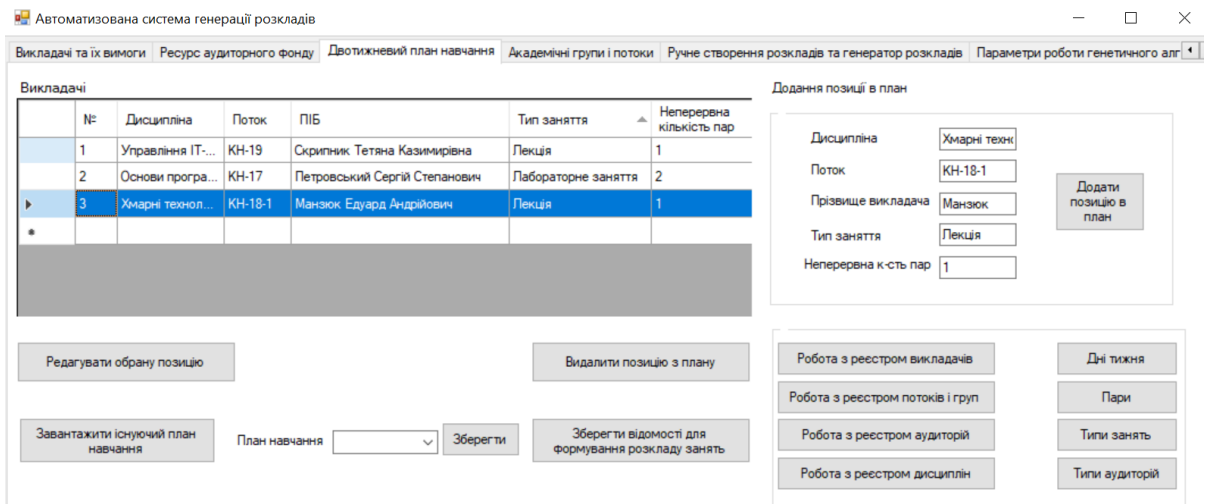


Рисунок 4.21 – Додання позиції в план

Перейдемо безпосередньо до роботи головного алгоритму та оберемо вкладку «Параметри роботи генетичного алгоритму». На даній вкладці оберемо всі параметри, які хоче врахувати користувач при генерації розкладу (рисунок 4.22).

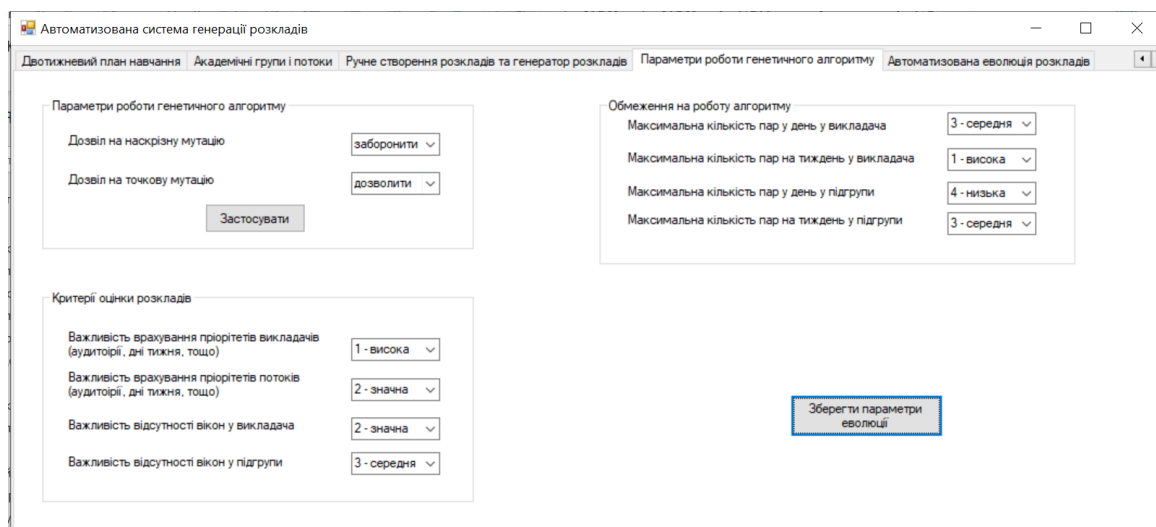


Рисунок 4.22 – Вибір параметрів на вкладці «Параметри роботи генетичного алгоритму»

Нехай користувач заповнює поля наступним чином. Тепер переходимо до вкладки «Автоматизована еволюція алгоритмів». На даній вкладці відобразатимуться розклади, що були сформовані алгоритмом (рисунок 4.23).

Автоматизована система генерації розкладів

Двотижневий план навчання Академічні групи і потоки Ручне створення розкладів та генератор розкладів Параметри роботи генетичного алгоритму Автоматизована еволюція розкладів

Обраний план навчання Розклад пар Оцінка

Версії розкладу по плану

№	Назва розкладу	Оцінка
1	новийN=35442	44.3261983
2	поточнийN=44	33.003817
3	новийN=35443	55.248401
*		

№	День тижня	№пари	Аудиторія	Дисципліна	Поток	Викладач	Тип заняття	Неперервна к-ть пар
1	понеділок	1	3-402	Управління I...	КН-19	Скрипник Т...	Лекція	1
2	понеділок	2	3-402	Хмарн техно...	КН-18-1	Манзюк Еду...	Лабораторне ...	2
3	вівторок	2	3-404	Основи прог...	КН-17	Петровський	Лабораторне	3
*								

Виконати епоху

Виконати 10 епох або знайти кращий розклад за існуючі

Зберегти обраний розклад як...

Рисунок 4.23 – Перегляд створеного розкладу

Таким чином, програмний продукт було створено із максимально простим та інтуїтивним інтерфейсом. Окремі модулі розташовано на різних вкладках, що дає змогу користувачеві швидко зорієнтуватись та використовувати програму швидко та зручно. Створена система генерації розкладів занять за генетичним алгоритмом виконана в повному обсязі та задовольняє функціональним вимогам поданим в п.3.1 для ефективного прикладного використання за призначенням.

4.4 Прикладне використання інформаційної системи

За результатом проведеного в п.4.3 функціонального дослідження інформаційної системи генерації розкладів занять за генетичним алгоритмом,

було визначено вимоги до її розгортання, що потрібно для ефективного використання та дослідження системи та її компонентів. Для забезпечення коректної роботи застосунку для генерації розкладів занять за генетичним алгоритмом необхідно виконати певні технічні вимоги та переконатись, що апаратна частина комп'ютера користувача задовольняє певні вимоги для запуску та коректної роботи програми.

Вимоги до апаратної частини:

- Microsoft .NET Framework, від версії 4.0 – поточна версія;
- дисковий простір – не менше 1000 Мб.
- основним сервером повинна бути СКБД SQL Server версії 14 0 17289 0

та вище;

- Windows Server, від версії 2012 – поточна версія;
- Microsoft SQL Server, від версії 2012 – поточна версія;

Вимоги до завантаження існуючих навчальних планів для аналізу:

- цифрові матеріали необхідно завантажити із розширенням .xml;
- необхідно зазначити назву цифрового матеріалу, що буде

проаналізовано.

Нижче наведено вимоги до обладнання та програмного забезпечення користувача – мінімальні та рекомендовані.

Мінімальні:

- ОС: Windows 10;
- Процесор: x64, 800GH;
- Пам'ять: 512 МВ;
- Дисплей: 1024x768.

Рекомендовані:

- ОС: Windows 10;
- Процесор: x64, Intel Core i5-7300 HQ;
- Пам'ять: 4096 МВ;
- Дисплей: 1920x1080.

Таблиця 4.4 – Можливості прикладного використання деяких функцій інформаційної системи для побудови розкладів занять за генетичним алгоритмом

Функції	Розроблені шляхи забезпечення функцій
Ручне створення розкладів занять	Інформаційна модель розкладу занять та структура БД, що забезпечують необхідну та достатню формалізацію й структуру сутностей предметної області
Автоматизоване створення розкладів занять	При виконанні методу побудови розкладів занять за генетичним алгоритмом у такий спосіб забезпечується створення початкових зразків розкладів занять
Автоматизована оцінка наявних розкладів занять	Шляхом прикладного використання розробленої математичної моделі обрахунку відповідності розкладу занять вимогам
Автоматизоване порівняння наявних розкладів занять	Шляхом порівняння оцінок відповідності вимогам цих розкладів занять, використовується при виконанні методу побудови розкладів занять за генетичним алгоритмом для пошуку кращих зразків
Автоматизований пошук шляхів покращення наявних розкладів занять	Шляхом використання операторів точкової та наскрізної мутації при роботі за генетичного алгоритму в методі побудови розкладів занять
Врахування вимог викладачів до аудиторій, днів тижня та номерів пар у розкладі	Врахування за рахунок виставлення пріоритетів викладачів до аудиторій, днів тижня та номерів пар
Врахування вимог підгрупи (групи) до кількостей пар в тиждень та поденно	Врахування за рахунок виставлення рекомендованих підгрупі кількостей пар в тиждень та поденно
Баланс груп вимог до розкладів викладачів і підгруп	Врахування за рахунок вагових коефіцієнтів врахування при оцінці розкладів важливості пріоритетів викладачів, пріоритетів потоків, відсутності вікон у викладача й у підгрупи
Врахування вимог викладачів і підгруп до відсутності вікон у розкладі	Врахування за рахунок зменшення умовної оцінки розкладу при наявності вікон у викладачів і підгруп
Врахування нормативних значень по кількості пар в день та тиждень при створенні розкладу занять	Врахування за рахунок встановлення обмежень на роботу методу (максимальна кількість пар в день у викладача, максимальна кількість пар в тиждень у викладача, максимальна кількість пар в день у групи, максимальна кількість пар в тиждень у групи)

За результатом прикладного використання створеного методу побудови розкладів занять за генетичним алгоритмом та супутніх засобів, було визначено

наступні практичні можливості для автоматизації задач предметної області автоматизації формування розкладів занять (Таблиця 4.4).

Як слідує із Таблиці 4.4, можливості розробленого методу побудови розкладів занять за генетичним алгоритмом та супутніх засобів забезпечують виконання ряду важливих функцій при створенні розкладів занять, що підкреслює практичну значущість одержаних результатів. Зокрема, забезпечується ручне створення розкладів занять, автоматизоване створення розкладів занять, автоматизована оцінка наявних розкладів занять, автоматизоване порівняння наявних розкладів занять, автоматизований пошук шляхів покращення наявних розкладів занять, врахування вимог викладачів до аудиторій, днів тижня та номерів пар у розкладі, врахування вимог підгрупи (групи) до кількостей пар в тиждень та поденно, баланс груп вимог до розкладів викладачів і підгруп, врахування вимог викладачів і підгруп до відсутності вікон у розкладі та врахування нормативних значень по кількості пар в день та тиждень при створенні розкладу занять.

Висновки до розділу 4

В розділі розглянуто особливості прикладної розробки інформаційної системи для побудови розкладів занять за генетичним алгоритмом, яка необхідна для подальшого дослідження ефективності методу побудови розкладів занять за генетичним алгоритмом. Для перевірки коректності роботи інформаційної системи для побудови розкладів занять за генетичним алгоритмом було проведено прикладне тестування інформаційної системи, визначено функції й особливості використання інформаційної системи для її подальшого функціонального дослідження.

Практичне використання розробленої системи підтвердило спроможність інформаційної системи забезпечити виконання функцій побудови розкладів занять за генетичним алгоритмом. Можливості розробленого методу побудови розкладів занять за генетичним алгоритмом та супутніх засобів забезпечують

виконання ряду важливих функцій при створенні розкладів занять, що підкреслює практичну значущість одержаних результатів. Зокрема, визначними рисами розробленого методу побудови розкладів занять за генетичним алгоритмом є врахування при оцінюванні зразків розкладів таких критеріїв, як пріоритети викладачів (аудиторії, дні тижня, пари), пріоритети потоків (рекомендовані кількості пар у день і в тиждень), а також врахування обмежень на роботу алгоритму таких як максимальна кількість пар в день й у тиждень у викладача і максимальна кількість пар в день і в тиждень в підгрупі, відсутність вікон в викладача і підгрупі.

Загальні висновки

Кваліфікаційна робота магістра розв'язує науково-технічну задачу автоматизації побудови розкладів занять за генетичним алгоритмом, який дозволяє із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму. В якості вхідних зразків розкладів слід передбачити як можливість введення вручну існуючих аналогів, так і можливість автоматизованого створення зразків розкладів. У результаті виконання роботи були поставлені та вирішені наступні завдання:

1. Проведено аналіз відомих підходів до автоматизованої побудови розкладів занять й шляхів до їх удосконалення.

2. Вдосконалено інформаційну модель розкладу занять як вичерпного подання предметної області, достатнього для побудови розкладів занять за генетичним алгоритмом.

3. Удосконалено метод побудови розкладів занять за генетичним алгоритмом, який використовує математичну модель обрахунку відповідності розкладу занять вимогам.

4. Розроблено математичну модель обрахунку відповідності розкладу занять вимогам.

5. Розроблено інформаційну систему генерації розкладів занять за генетичним алгоритмом згідно створеного методу.

6. Проведено прикладне та функціональне дослідження методу побудови розкладів занять за генетичним алгоритмом.

В результаті виконаної роботи були отримані такі інновації та положення наукової новизни:

1. Вдосконалено інформаційну модель розкладу занять, яка відрізняється тим, що містить подання всіх необхідних сутностей для побудови розкладів занять за генетичним алгоритмом, включаючи параметри роботи генетичного

алгоритму, обмеження на компоненти розкладу занять та критерії оцінки розкладів.

2. Вдосконалено метод побудови розкладів занять за генетичним алгоритмом, який використовує математичну модель обрахунку відповідності розкладу занять вимогам і відрізняється тим, що дозволяє за наявними в інформаційній моделі розкладу занять параметрами із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму.

3. Розроблено нову інформаційну систему генерації розкладів занять за генетичним алгоритмом, що дозволяє з використанням створених інформаційної моделі розкладу занять, методу побудови розкладів занять за генетичним алгоритмом та математичної моделі обрахунку відповідності розкладу занять вимогам виконувати ручне та автоматизоване створення нових розкладів, формувати нові похідні розклади із використанням генетичного алгоритму, оцінювати сформовані розклади занять із урахуванням вимог викладачів та академічних груп до параметрів проведення пар й визначати кращі зразки розкладів занять за результатом порівняння їх оцінок відповідності вимогам.

Також в роботі розроблено математичну модель обрахунку відповідності розкладу занять вимогам, що дозволяє за наявними в інформаційній моделі розкладу занять параметрами автоматизовано обраховувати умовну оцінку відповідності визначеного зразка розкладу занять вимогам.

Практичне використання розробленої системи підтвердило спроможність інформаційної системи забезпечити виконання функцій побудови розкладів занять за генетичним алгоритмом. Можливості розробленого методу побудови розкладів занять за генетичним алгоритмом і супутніх засобів забезпечують виконання ряду важливих функцій при створенні розкладів занять, що підкреслює практичну значущість одержаних результатів. Зокрема,

забезпечується ручне створення розкладів занять, автоматизоване створення розкладів занять, автоматизована оцінка наявних розкладів занять, автоматизоване порівняння наявних розкладів занять, автоматизований пошук шляхів покращення наявних розкладів занять, врахування вимог викладачів до аудиторій, днів тижня та номерів пар у розкладі, врахування вимог підгрупи (групи) до кількостей пар в тиждень та поденно, баланс груп вимог до розкладів викладачів і підгруп, врахування вимог викладачів і підгруп до відсутності вікон у розкладі та врахування нормативних значень по кількості пар у день і тиждень при створенні розкладу занять.

Визначними рисами розробленого методу побудови розкладів занять за генетичним алгоритмом є врахування при оцінюванні зразків розкладів таких критеріїв, як пріоритети викладачів (аудиторії, дні тижня, пари), пріоритети потоків (рекомендовані кількості пар у день і в тиждень), відсутність вікон в викладача і підгрупи, а також врахування обмежень на роботу алгоритму таких як максимальна кількість пар в день і у тиждень у викладача і максимальна кількість пар в день і в тиждень в підгрупи.

Основні наукові й практичні результати кваліфікаційної роботи магістра у доповіді на тему «Метод побудови розкладів занять за генетичним алгоритмом» доповідались на XIII Всеукраїнській науково-практичній конференції «Актуальні проблеми комп'ютерних наук АПКН-2021» (15-16 жовтня 2021 року); за темою роботи автором виконано наукову публікацію [39].

Перелік посилань

1. Бойко О.М. Еволюційна технологія розв'язування задачі складання розкладів навчальних занять. Штучний інтелект. – 2006. – №.3. – С. 341-348.
2. Освіта як вища цінність людини і суспільства. Новосад О. В. URL: <http://ena.lp.edu.ua:8080/bitstream/ntb/20469/1/138-Novosad-225-227.pdf>
3. Психологиня. Освіта як спосіб входження людини у світ науки і культури. URL: <https://ru.osvita.ua/vnz/reports/psychology/29246/>
4. Wikipedia. Початкова освіта. URL: https://uk.wikipedia.org/wiki/Початкова_освіта
5. Верховна Рада. Про загальну середню освіту. URL: <https://zakon.rada.gov.ua/laws/show/651-14#Find>
6. Wikipedia. Професійно-технічна освіта. URL: https://uk.wikipedia.org/wiki/Професійно-технічна_освіта
7. Wikipedia. Вища освіта. URL: https://uk.wikipedia.org/wiki/Вища_освіта
8. Освітня політика. Навчальний процес. URL: <http://education-ua.org/navchalniy-proces.php>
9. Верховна Рада України. Закон України про вищу освіту URL: <https://zakon.rada.gov.ua/laws/show/1556-18/print1382613528661298#Text>
10. Деканова М.В. Математична модель и алгоритм побудови розкладу навчальних занять університету / Деканова М.В. // Вісник Полоцького державного університету. Серія С. – 2013. – №12. – С. 24-33.
11. Wikipedia. Теорія розкладів. URL: https://uk.wikipedia.org/wiki/Теорія_розкладів
12. Wikipedia. Генетичний алгоритм. URL: https://uk.wikipedia.org/wiki/Генетичний_алгоритм
13. Портал знань. Генетичні алгоритми. Ключові поняття і методи реалізації URL: http://www.znannya.org/?view=ga_general

14. Паралельний генетичний алгоритм побудови розкладу занять / М.М. Глибовець, Н.М. Гулаєва, М.М. Пасічник // Проблеми програмування. - 2015. - № 2. - С. 76-85.
15. University Timetabling. Comprehensive Academic Scheduling Solutions. Comprehensive University Timetabling System. URL: <https://www.unitime.org/>
16. 1С. Автоматизоване складання розкладу. Університет. URL: https://solutions.1c.ru/catalog/asp_univer/features
17. Розклад занять/уроків: «Ректор» – Програма лідер! Розклад занять: «Ректор-ВНЗ». URL: <http://rector.spb.ru/raspisanie-vuz-4u.php>
18. Програми для дому та бізнесу AllSoft.ua. Програма складання розкладу занять «НІКА». URL: https://allsoft.ua/software/vendors/nika-soft/programma-sostavleniya-raspisaniya-zanyatiy-nika/#product_description
19. Лабораторія ММІС. «Додаток АВТОРозклад». URL: <https://www.mmis.ru/programs/avtor>
20. Бурнасов П.В. Математична постановка задачі складання розкладу занять // Вісник ІрГТУ. 2014. №4. С. 12-18.
21. Бульонков М.А., Ємельянов П.Г., Пак Є.В. Стандартизація даних для складання розкладу в навчальних закладах. Відкрита освіта. 2010. № 3. С. 45-57.
22. Томашевський В.М., Новіков Ю.Л., Камінська П.А. Складання розкладів занять у дистанційних системах навчання // Вісник НТУУ «КПІ» Інформатика, управління та обчислювальна техніка, 2010. №. 52. С. 118-130.
23. Jha S.K. Exam Timetabling Problem using Genetic algorithm // International Journal of Research in Engineering and Technology. 2014. Vol.3, №5, Pp. 649-655
24. Юрчак І.Ю., Москович Т.Р. Дослідження генетичних алгоритмів та їх застосування їхнього в автоматизованій системі розподілу навантаження для викладачів і студентів. URL: <http://eom.lp.edu.ua/sntk/doc/ksm2018/moskovytch.pdf>

25. Verma O.P., Garg. R., Bisht V.S., Optimal Time-Table Generation by Hybridized Bacterial Foraging and Genetic Algorithm //Proceedings of International Conference on Communication Systems and Network Technologies (CSNT'12), (2012), Pp. 919-923.

26. Ю. С. Кабальнов, Л. И. Шехтман, Г. Ф. Нізамова, Н. А. Земченкова. Композиційний генетичний алгоритм складання розкладу навчальних занять. URL: <https://cyber.ru/article/n/kompozitsionnyu-geneticheskiy-algoritm-sostavleniya-raspisaniya-uchebnyh-zanyatiy/viewer>

27. Годлевський М. Д. Дослідження ефективності паралельних генетичних алгоритмів для вирішення задачі створення розкладу занять вузу на базі Grid-системи / М. Д. Годлевський, О. О. Абабілов // Радіоелектронні і комп'ютерні системи. - 2011. - № 3. - С. 68–72.

28. What is PostgreSQL? URL: <https://www.educba.com/what-is-postgresql/>

29. PostgreSQL for data science : pro and cons. URL: <https://dataintoreresults.com/post/postgresql-for-data-science-pro-and-cons/>

30. What is SQL Server. URL: <https://www.sqlservertutorial.net/getting-started/what-is-sql-server/>

31. Pros and cons of SQL Server Stored Procedures. URL: <https://itzone.com.vn/en/article/pros-and-cons-of-sql-server-stored-procedures/>

32. Top Pros and Cons of Java development Platform. URL: <http://expertjavadev.over-blog.com/top-pros-and-cons-of-java-development-platform>

33. The Good And The Bad Of .NET Framework Development. URL: <https://www.cisin.com/coffee-break/enterprise/the-good-and-the-bad-of-net-framework-development.html>

34. What is F#. URL: <https://www.codeproject.com/Articles/741874/What-is-Fsharp>

35. Visual Basic (VB) URL: <https://whatis.techtarget.com/definition/Visual-Basic-VB>

36. What are disadvantages of Visual Basic? URL:
<https://answerstoall.com/science/what-are-disadvantages-of-visual-basic/>

37. C Sharp – Features, Advantages and Disadvantages. URL:
<https://urbannaturale.com/c-sharp-features-advantages-and-disadvantages/>

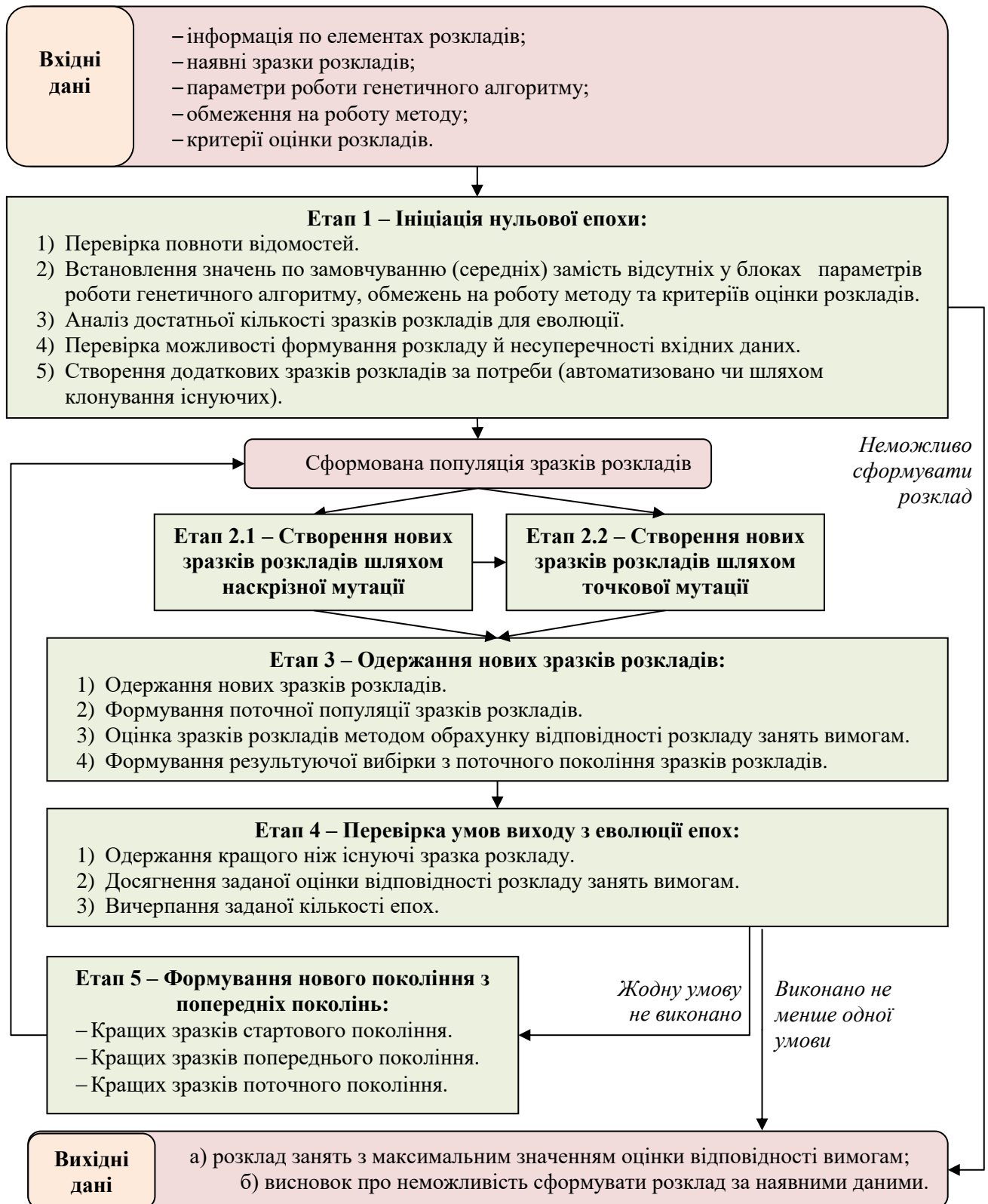
38. Гарсія-Моліна Г., Ульман Дж., Уідом Дж. Системи баз даних. Database Systems: The Complete Book. Вільямс, 2003. 1088 с.

39. Богач В. В., Шамрелюк В. В., Шпичко А. В., Мазурець О. В. Метод побудови розкладів занять за генетичним алгоритмом. Збірник наукових праць за матеріалами XIII Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2021». Хмельницький, 2021. с. 291-297.

ДОДАТКИ

Додаток А

Схема методу побудови розкладів занять за генетичним алгоритмом



Додаток Б

Схема інформаційної системи генерації розкладів занять за генетичним алгоритмом



Додаток В

Положення Закону України «Про вищу освіту», що регламентують формування розкладів занять у вишах

Стаття 10. Стандарти вищої освіти

1. Стандарт вищої освіти - це сукупність вимог до освітніх програм вищої освіти, які є спільними для всіх освітніх програм у межах певного рівня вищої освіти та спеціальності.

2. Стандарти вищої освіти розробляються для кожного рівня вищої освіти в межах кожної спеціальності відповідно до Національної рамки кваліфікацій і використовуються для визначення та оцінювання якості вищої освіти та результатів освітньої діяльності закладів вищої освіти (наукових установ), результатів навчання за відповідними спеціальностями.

3. Стандарт вищої освіти визначає такі вимоги до освітньої програми:

1) обсяг кредитів ЄКТС, необхідний для здобуття відповідного ступеня вищої освіти;

2) вимоги до рівня освіти осіб, які можуть розпочати навчання за цією програмою, та результатів їх навчання;

3) перелік обов'язкових компетентностей випускника;

4) нормативний зміст підготовки здобувачів вищої освіти, сформульований у термінах результатів навчання;

5) форми атестації здобувачів вищої освіти;

6) вимоги до створення освітніх програм підготовки за галуззю знань, двома галузями знань або групою спеціальностей (у стандартах рівня молодшого бакалавра), міждисциплінарних освітньо-наукових програм (у стандартах магістра та доктора філософії);

7) вимоги професійних стандартів (за їх наявності).

4. Стандарти вищої освіти спеціальностей, необхідних для доступу до професій, для яких запроваджене додаткове регулювання, можуть містити додаткові вимоги до правил прийому, структури освітньої програми, змісту освіти, організації освітнього процесу та атестації випускників. Нормативний зміст підготовки здобувачів вищої освіти для відповідних освітніх програм формується у термінах програмних результатів навчання.

5. Заклад вищої освіти на підставі відповідної освітньої програми розробляє навчальний план, що визначає перелік та обсяг освітніх компонентів у кредитах ЄКТС, їх логічну послідовність, форми організації освітнього процесу, види та обсяг навчальних занять, графік навчального процесу, форми поточного і підсумкового контролю, що забезпечують досягнення здобувачем відповідного ступеня вищої освіти програмних результатів навчання.

На основі навчального плану у визначеному закладом вищої освіти порядку для кожного здобувача вищої освіти розробляються та затверджуються індивідуальні навчальні плани на кожний навчальний рік. Індивідуальний навчальний план формується за результатами особистого вибору здобувачем вищої освіти дисциплін в обсязі, не меншому за встановлений цим Законом, з урахуванням вимог освітньої програми щодо вивчення її обов'язкових компонентів. Індивідуальний навчальний план є обов'язковим для виконання здобувачем вищої освіти.

6. Стандарти вищої освіти за кожною спеціальністю розробляє центральний орган виконавчої влади у сфері освіти і науки з урахуванням пропозицій галузевих державних органів, що забезпечують формування і реалізують державну політику у відповідних сферах, галузевих об'єднань організацій роботодавців і затверджує їх за погодженням з Національним агентством із забезпечення якості вищої освіти.

Стаття 50. Форми організації освітнього процесу та види навчальних занять

1. Освітній процес у закладах вищої освіти здійснюється за такими формами:

- 1) навчальні заняття;
- 2) самостійна робота;
- 3) практична підготовка;
- 4) контрольні заходи.

2. Основними видами навчальних занять у закладах вищої освіти є:

- 1) лекція;
- 2) лабораторне, практичне, семінарське, індивідуальне заняття;
- 3) консультація.

3. Заклад вищої освіти має право встановлювати інші форми освітнього процесу та види навчальних занять.

Додаток Г
Світлини наукових публікацій, виконаних при роботі над
кваліфікаційною роботою магістра

(ксерокопії титульної сторінки, сторінки змісту та всіх сторінок із публікацією)

Перелік наукових публікацій:

1. Богач В. В., Шамрелюк В. В., Шпичко А. В., Мазурець О. В. Метод побудови розкладів занять за генетичним алгоритмом. Збірник наукових праць за матеріалами XIII Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2021». Хмельницький, 2021. с. 291-297.

Міністерство освіти і науки України
Хмельницький національний університет



ЗБІРНИК НАУКОВИХ ПРАЦЬ
за матеріалами XIII Всеукраїнської науково-практичної конференції
«Актуальні проблеми комп'ютерних наук АПКН-2021»

15-16 жовтня 2021

Хмельницький 2021

Федчук М. Ю.	251
Веб-сайт замовлення продуктів харчування	251
Федоринин О. М., Яцків В. В.	254
Спосіб кодування даних сенсорів на основі системи залишкових класів	254
Френс В. О., Бармак О. В.	257
Особливості використання протоколу NB-IoT для проектування та оптимізації взаємодії компонентів інтернету речей	257
Чума Е. В.	260
Інтелектуальний алгоритм розв'язування логістичних проблем міського трафіку	260
Шамрелюк В. В., Собко О. В., Молчанова М. О., Мазурець О. В.	264
Інформаційна модель генетичного алгоритму навчання нейронної мережі	264
Швайко В. К., Авсієнч В. Р.	268
Інформаційна система візуалізації пунктів переробки вторинної сировини для забезпечення концепції сталого розвитку	268
Шевченко В. Л., Лазоренко Я. С.	272
Формалізація закономірностей зміни інтонації	272
Шевчук О. О.	274
Методи прийняття рішень в умовах нечіткої інформації в задачах розподілення робіт між працівниками	274
Шиникін О. В., Марченко А. В.	278
Інформаційна система аналізу збітків від техногенних та природних катастроф ..	278
Андрушко В. В., Суринник Т. К.	281
Моделі та методи для веб-аналітики відвідуваності сайтів	281
Бананіко Т. Г., Петроєвський С. С.	284
Методи та засоби оцінювання релевантності мультимедійних навчальних курсів у школі	284
Біловол А. І.	287
Удосконалення методу та засобів очищення даних на основі matching dependency technique	287
Богач В. В., Шамрелюк В. В., Шиничко А. В., Мазурець О. В.	291
Метод побудови розкладів занять за генетичним алгоритмом	291
Бойчинин О. О., Залуцька О. О., Попов Ю. М., Курійчук В. О.	298
Інформаційна технологія автоматизованого формування семантичного ядра цифрових текстів	298

Галліна Р. І., Багрий Р. О., Суринник Т. К.	306
Застосування адаптивного підходу для реалізації системи опитувань та тестувань	306
Гринь С. С., Пивоєар О. С., Таранчук А. А.	309
Забезпечення прихованості дії та криптографічного захисту аналогових сигналів в хаотичній системі зв'язку	309
Данчук С. В., Багрий Р. О.	312
Технологія автоматизованого отримання даних з веб-ресурсів для бізнес-аналітики	312
Длушунювич Н. А.	316
Інформаційна технологія фінансового моделювання для розвитку малого підприємництва	316
Дрозд А. І., Формун Ю. В.	319
Метод розподілу обчислювальних ресурсів для обробки розподілених потоків даних	319
Дудар О. В., Михалевський В. П., Суринник Т. К.	321
Інформаційна система для забезпечення підтримки екологічної рівноваги	321
Єфімчук А. С., Суринник Т. К., Мазурець О. В., Молчанова М. О.	324
Автоматизований розподіл процесів при управлінні IT-проєктами в складних критично-безпекових умовах	324
Житкевич В. В., Медведчук В. Ю.	332
Метод віновлення пошкоджених растрових зображень	332
Заровний В. І., Суринник Т. К.	335
Методи шифрованої передачі даних між змаркованими підпросторами	335
Курдявцев В. В., Формун Ю. В.	338
Аналіз та застосування методів оптимізації пвпшкелі та відмовостійкості програмних продуктів	338
Курдибаха А. В., Мазурець О. В., Собко О. В., Молчанова М. О.	340
Інформаційна технологія оцінювання діяльності сімейного лікаря за даними прийомів	340
Лавренцій А. А., Петроєвський С. С.	349
Метод оцінювання наповненості дистанційних курсів предметів у школі	349
Левченко Т. В., Блажук В. Д., Молчанова М. О., Собко О. В.	352
Метод оптимізації транспортних перевезень засобами біологічної метаверстивки	352

УДК 004

Богач В. В., Шамрелок В. В., Шпичко А. В., Мазурець О. В.

Хмельницький національний університет

МЕТОД ПОБУДОВИ РОЗКЛАДІВ ЗАНЯТЬ ЗА ГЕНЕТИЧНИМ АЛГОРИТМОМ

Розглянуто метод побудови розкладів занять за генетичним алгоритмом призначений для автоматизованого формування зразків розкладів занять й пошуку серед них розкладу з максимальним значенням оцінки відповідності розкладу заняття поставленим вимогам за використання генетичного алгоритму. Формування зразків розкладів занять виконується за інформування по елементах розкладів, наявними зразками розкладів, параметрами роботи генетичного алгоритму, параметрами обмеження на роботу методу та критеріями оцінки розкладів. Визначення рисами розробленого методу побудови розкладів занять за генетичним алгоритмом є врахування при оцінюванні зразків розкладів таких критеріїв, як пріоритетні викладачі (аудиторії, дні тижня, пари), пріоритетні пари (рекомендовані кількості пар в день і в тиждень), відсутність віком у викладача і підручні, а також врахування обмежень на роботу алгоритму таких як максимальна кількість пар в день і в тиждень у викладача й максимальна кількість пар в день і в тиждень у підручні.

The method of classes construction according to the genetic algorithm is considered for automated formation of samples of classes and search among them of the schedule with maximum value of estimation of conformity of classes schedule to the set requirements for use of genetic algorithm. Formation of samples of schedules of employment is carried out on information on elements of schedules, available samples of schedules, parameters of work of genetic algorithm, parameters of restriction on work of a method and criteria of an estimation of schedules. Distinctive features of developed method of scheduling classes by genetic algorithm are taking into account when evaluating samples of schedules such criteria as teacher priorities (classrooms, days of the week, pairs), flow priorities (recommended number of pairs per day and per week), no teacher and subgroup windows, as well as taking into account restrictions on the operation of the algorithm such as the maximum number of pairs per day and per week for the teacher and the maximum number of pairs per day and per week for subgroup.

Для того, щоб навчальний процес відбувався безперебійно та за планом не достатньо лише викладача, учня та підручника. Завичай навчальний процес відбувається паралельно між кількома групами учнів/студентів та викладачем. Навчальний процес можна умовно розбити на заняття, що мають свої властивості – це викладач, студент (якщо це заняття вищих навчальних закладів), предмет заняття, час та день тижня, аудиторія в якій проводиться заняття [1].

Для забезпечення нормального функціонування всіх складових навчального процесу складаються розклади. Особливо важливо це для вищих навчальних

закладів, так як групи студентів не мають закріплених за ними аудиторій, як, наприклад, в школі. Також свою складність в процес розробки розкладу вносять і тип занять – лекція, лабораторна чи практична робота. Відповідно до кількості студентів та типу занять потрібно підібрати відповідну аудиторію та час [2, 3].

Розклад – це чіткий план взаємодії людей у робочому процесі, розписаний у часі і просторі. За класичним означенням, розклад – це документ, що містить, який регламентує робочий ритм, визначає часові обмеження всіх робочих процесів і формує оптимальне розподілення такого важливого ресурсу як час [4]. В загальному вираженням задачі побудови розкладу сьогоднішній час займається розділ дискретної математики, який називається «Теорія розкладів».

Одним з інструментів для складання розкладів сьогодні є генетичний алгоритм – еволюційний алгоритм пошуку, що використовується для вирішення задач оптимізації і моделювання шляхом послідовного підбору, комбінування і варіації шуканих параметрів з використанням механізмів, що нагадують біологічну еволюцію.

Класична схема розробки генетичного алгоритму в контексті складання розкладу занять включає декілька етапів. На першому – розробляється структура та представлення хромосоми, в якій буде зберігатися розв'язок. Обрана структура має враховувати всі особливості, які висуваються до шуканого розкладу, а також те, що від її вибору напряму залежить реалізація алгоритмів кросингверу та мутації. Потім створюється початкова популяція, розмір якої залежить від розмірності задачі. Для організації оптимізаційного процесу необхідно створити напрямки розвитку популяції. Цим напрямком може виступати мінімізація цільової функції (функції пристосованості чи здоров'я). Ця функція має адекватно відізняти два варіанти розкладу. На основі значення оцінки хромосоми відбувається відбір хромосом у батьківський пул, подальше застосування операторів кросингверу та мутації. Помітивши початкову популяцію у створене штучне середовище і реалізувавши процес селекції, кросингверу і мутації, приходимо до ітераційного алгоритму пошуку оптимального розв'язку [5].

Мета роботи полягає у розробці методу побудови розкладів занять за генетичним алгоритмом, який дозволить із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вищів за допомогою еволюційного генетичного алгоритму. В якості викідних зразків розкладів слід передбачити як можливість введення вручну існуючих аналогів, так і можливість автоматизованого створення зразків розкладів.

Метод побудови розкладів занять за генетичним алгоритмом призначений для формування розкладу занять з максимальним значенням оцінки відповідності розкладу занять поставленим вимогам за інформування по елементах розкладів, наявними зразками розкладів, параметрами роботи генетичного алгоритму, параметрами обмеження на роботу методу та критеріями оцінки розкладів. На Рисунку 1 зображено схему етапів роботи методу побудови розкладів занять за генетичним алгоритмом.

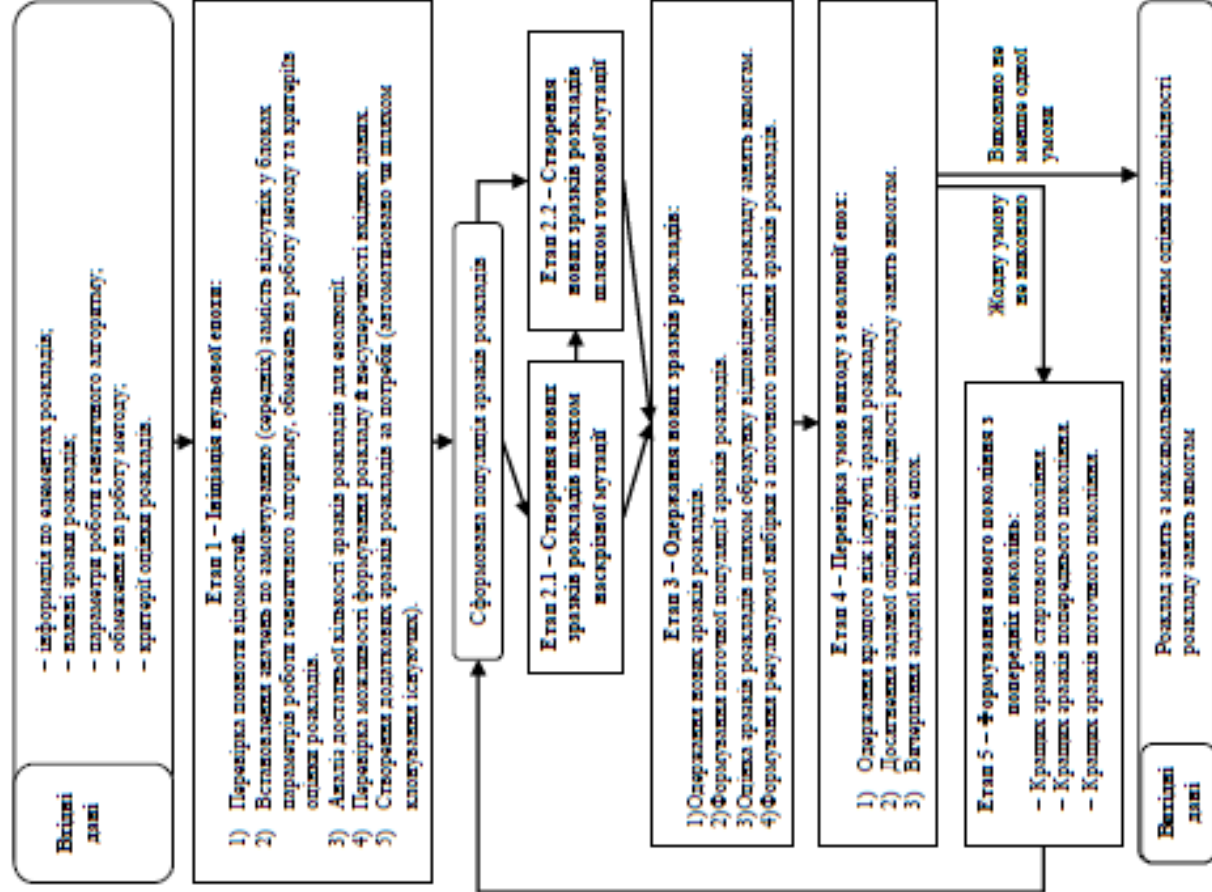


Рисунок 1 – Схема методу побудови розкладів занятя за генетичним алгоритмом

Вхідними даними методу побудови розкладів занятя за генетичним алгоритмом є масив даних (Рисунок 2), що групується за ознаками у інформацию по елементах розкладів, назвні зразки розкладів, параметри роботи генетичного алгоритму, параметри обмеження на роботу методу та критерії оцінки розкладів. У розгорнутому вигляді масив вільних даних має наступний зміст:

- інформація по елементах розкладів (викладачі, дисципліни, потоки, групи, аудиторії, дні тижня, типи занять, типи аудиторій);
- назвні зразки розкладів;
- параметри роботи генетичного алгоритму (дозволи на наскрізну та точкову мутації);
- обмеження на роботу методу (максимальна кількість пар в день у викладача, максимальна кількість пар в тиждень у викладача, максимальна кількість пар в день у групи, максимальна кількість пар в тиждень у групи);
- критерії оцінки розкладів (важливність врахування пріоритетів викладачів, важливність врахування пріоритетів потоків, важливність відсутності вікон у викладача, важливність відсутності вікон у підгрупи).

На Етапі 1 виконується ініціалізація нульової епохи роботи генетичного алгоритму. Для цього спершу проводиться перевірка повноти відомостей у вільних даних. Далі здійснюється встановлення значень по замовчуванню (середні) змість відсутніх у блоках параметрів роботи генетичного алгоритму. Також здійснюється встановлення обмежень на роботу методу та критеріїв оцінки розкладів.

Наступним кроком виконується аналіз достатньої кількості зразків розкладів для еволюції й перевірка можливості формування розкладу і несуперечливості вільних даних. У випадку потреби автоматизовано чи шляхом клонування існуючих здійснюється створення додаткових зразків розкладів за потреби. Результатом цих дій є сформована популяція зразків розкладів.

При роботі генетичного алгоритму на Етапі 2.1 відбувається створення нових зразків розкладів шляхом наскрізної мутації, одночасно на Етапі 2.2 відбувається створення нових зразків розкладів занятя шляхом точкової мутації. При наскрізній мутації в зразках розкладів відбувається повна заміна двох днів тижня чи аудиторій однакового призначення. При точковій мутації відбувається заміна одного параметру (аудиторія, номер пари, день тижня) на випадковий із числа доступних і несуперечливих наявним позиціям розкладу.

Як наслідок, на Етапі 3 здійснюється одержання нових зразків розкладів. Для цього спершу відбувається формування поточної популяції зразків розкладів. Далі виконується оцінка зразків розкладів шляхом обчислення відповідності розкладу заданим вимогам. Це дозволяє виконати формування результуючої вибірки з поточного покоління зразків розкладів.

Етап 4 відповідає за перевірку умов виходу з еволюцій епохи. Такими умовами виходу з еволюцій епохи можуть бути одержання кращого ніж існуючі

зразка розкладу, досягнення заданої відповідності розкладу занять вимогам або вичерпання заданої кількості епох.

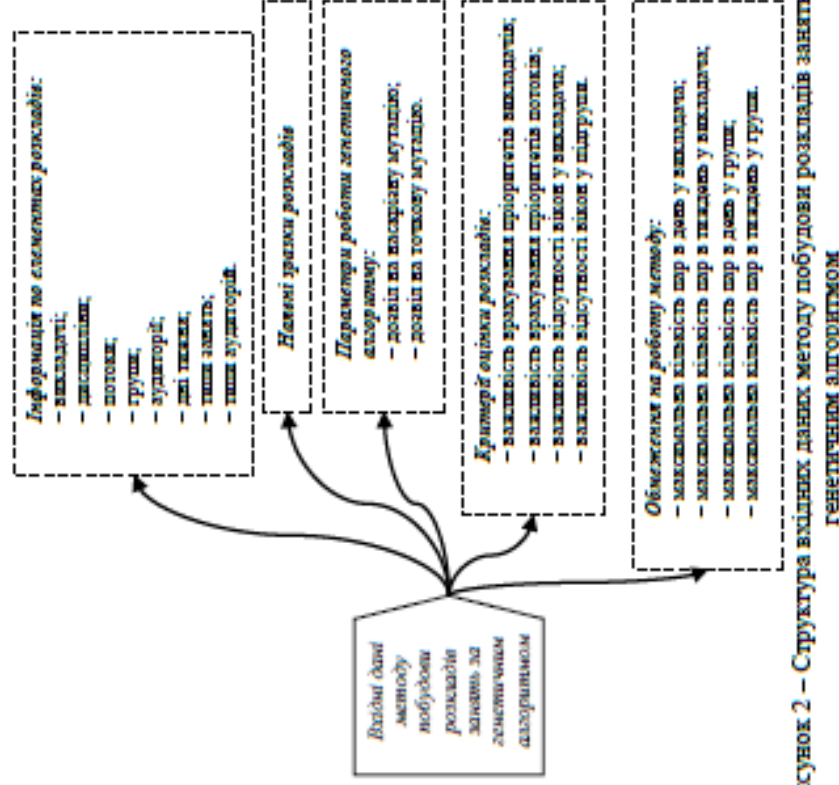


Рисунок 2 – Структура вхідних даних методу побудови розкладів занять за генетичним алгоритмом

Якщо при перевірці умов виходу з еволюції епох жодну умову не виконано, то відбувається перехід на Етап 5 для формування нового покоління з зразків розкладів із попередніх поколінь. При цьому для формування нового покоління використовуються визначені кількості кращих зразків стартового покоління, кращих зразків попереднього покоління та кращих зразків поточного покоління.

Якщо ж при перевірці умов виходу з еволюції епох виконано не менше одної умови, то відбувається формування вихідних даних методу у вигляді розкладу занять із максимальним значенням оцінки відповідності розкладу занять вимогам. На Рисунок 3 зображено створену авторами програмну систему, яка використовує розроблений метод і дозволяє формувати розклад занять з максимальним значенням

оцінки відповідності розкладу занять поставленим вимогам за інформацією про елементи розкладів, наявними зразками розкладів, параметрами роботи генетичного алгоритму, параметрами обмеження на роботу методу та критеріями оцінки розкладів.

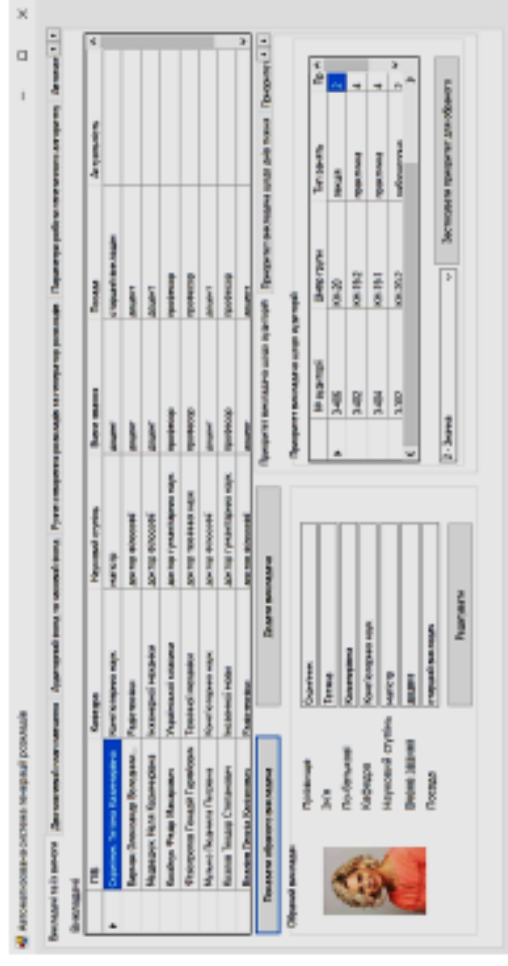


Рисунок 3 – Розроблена інформаційна система автоматизованого формування розкладів занять за генетичним алгоритмом

Таким чином, розроблено метод побудови розкладів занять за генетичним алгоритмом призначений для автоматизованого формування зразків розкладів занять й пошуку серед них розкладу з максимальним значенням оцінки відповідності розкладу занять поставленим вимогам за використанням генетичного алгоритму. Формування зразків розкладів занять виконується за інформацією про елементи розкладів, наявними зразками розкладів, параметрами роботи генетичного алгоритму, параметрами обмеження на роботу методу та критеріями оцінки розкладів.

Визначними рисами розробленого методу побудови розкладів занять за генетичним алгоритмом є врахування при оцінюванні зразків розкладів таких критеріїв, як пріоритети викладачів (аудиторії, дні тижня, пари), пріоритети потоків (рекомендовані кількості пар в день і в тиждень), відсутність вікон у викладача і підгрупи, а також врахування обмежень на роботу алгоритму таких як максимальна кількість пар в день і в тиждень у викладача й максимальна кількість пар в день і в тиждень у підгрупи.

Перелік посилань:

1. Бойко О.М. Еволюційна технологія розв'язування задачі складання розкладів навчальних занять // Бойко О.М. // Штучний інтелект. – 2006. – № 3. – С. 341-348.
2. Бурнасов П.В. Математична постановка задачі складання розкладу занять // Вісник ІрІТУ. 2014. №4. С. 12-18.
3. Томашевський В.М., Новіков Ю.Л., Каміська П.А. Складання розкладів занять у дистанційних системах навчання // Вісник НТУУ «КПІ» Інформатика, управління та обчислювальна техніка, 2010. № 52. С. 118-130.
4. Демчук М.В., Мазурень О.В. Автоматизація ведення розкладу занять у вузі. Збірник наукових праць за матеріалами восьмої міжнародної науково-технічної конференції «Актуальні проблеми комп'ютерних технологій 2014». Хмельницький, 2014. С.87-93.
5. Паралельний генетичний алгоритм побудови розкладу занять / М.М. Глибовель, Н.М. Гулаєва, М.М. Пастічник // Проблеми програмування. - 2015. - № 2. - С. 76-85.

УДК 004

Войчипшин О. О., Залупська О. О., Погоч Ю. М., Курпійчук В. О.

Хмельницький національний університет

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ АВТОМАТИЗОВАНОГО ФОРМУВАННЯ СЕМАНТИЧНОГО ЯДРА ЦИФРОВИХ ТЕКСТІВ

Розглянуто інформаційну технологію автоматизованого формування семантичного ядра цифрових текстів, яка дозволяє перетворювати вхідні дані у вигляді цифрового тексту, можливі слів і словосполучень тексту з показниками їх семантичної важливості в вихідні дані у вигляді зразків семантичного ядра тексту. Зразки семантичного ядра тексту одержуються у вигляді: із слів при обрахунку порогу щільності у символі, із словосполучень при обрахунку порогу щільності у символі, із слів при обрахунку порогу щільності у словах та із словосполучень при обрахунку порогу щільності у словах.

Наведені в статті зразки програмного забезпечення, які дозволяють створювати можливі терміни цифрових текстів, формувати семантичне ядро методом прикладного застосування розробленої інформаційної технології, а також практичне використання результатів для адаптивної пропозиційної технології у інтернет-магазинах за семантичними ознаками, демонструють певний набір компонентів для програмного вирішення актуальної задачі інформаційних технологій.

Information technology for automated formation of semantic core of digital texts is considered, which allows to convert input data in form of digital text, sets of words and phrases of text with indicators of their semantic importance into source data in the form of samples of text semantic core. Samples of semantic core of text are obtained in variations: from words when calculating the density threshold in symbols, from phrases when calculating density threshold in symbols, from words when calculating the density threshold in words and from phrases when calculating density threshold in words.

The software samples presented in article, which allow to create sets of digital text texts, form a semantic core by applying developed information technology, as well as practical use of results for adaptive supply of goods in online store on semantic features, demonstrate a full set of components for practical solution.

Електронний текст став феноменом, якому у сучасному науковому просторі приділяється велика кількість уваги. Саме він розглядається як основне джерело інформації. Існує кілька підходів до його аналізу. Можна, наприклад, визначати тему і ідею текстів, аналізувати, оцінювати смислове навантаження або виділяти сферу, з якою вони пов'язані (математика, комп'ютерні науки, література, соціологія) [1].

У зв'язку з тим, що мова являє собою досить складне утворення, в комп'ютерній лінгвістиці склалися і розвиваються різні напрями, приблизно порівнянні з окремими рівнями мови, з процесами породження і сприйняття

Додаток Д

Презентаційний матеріал

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

МЕТОД ПОБУДОВИ РОЗКЛАДІВ ЗАНЯТЬ ЗА ГЕНЕТИЧНИМ АЛГОРИТМОМ

Виконав:

Богач Владислав Валерійович

Керівник:

Мазурець Олександр Вікторович

Мета роботи

Мета кваліфікаційної роботи магістра полягає у розробці методу побудови розкладів занять за генетичним алгоритмом, який дозволяє із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму. В якості вхідних зразків розкладів слід передбачити як можливість введення вручну існуючих аналогів, так і можливість автоматизованого створення зразків розкладів.

При оцінці сформованих розкладів занять у межах роботи генетичного алгоритму потрібно врахувати пріоритети викладачів щодо бажаної аудиторії проведення пари, днів тижня та власне номерів пар. Також слід врахувати пріоритети академічних груп щодо рекомендованої кількості пар в день і рекомендованої кількості пар в тиждень. Окрім урахування наведених пріоритетів викладачів та пріоритетів академічних груп, слід забезпечити врахування показників:

- важливості врахування пріоритетів викладачів;
- важливості врахування пріоритетів потоків;
- важливості відсутності вікон у викладачів;
- важливості відсутності вікон у групах.

Обмеженнями на роботу методу побудови розкладів занять за генетичним алгоритмом мають бути:

- максимальна кількість пар в день у викладача;
- максимальна кількість пар в тиждень у викладача;
- максимальна кількість пар в день у групи;
- максимальна кількість пар в тиждень у групи.

Задачі дослідження

Для досягнення поставленої мети розробки методу побудови розкладів занять за генетичним алгоритмом необхідно розв'язати наступні **задачі дослідження**:

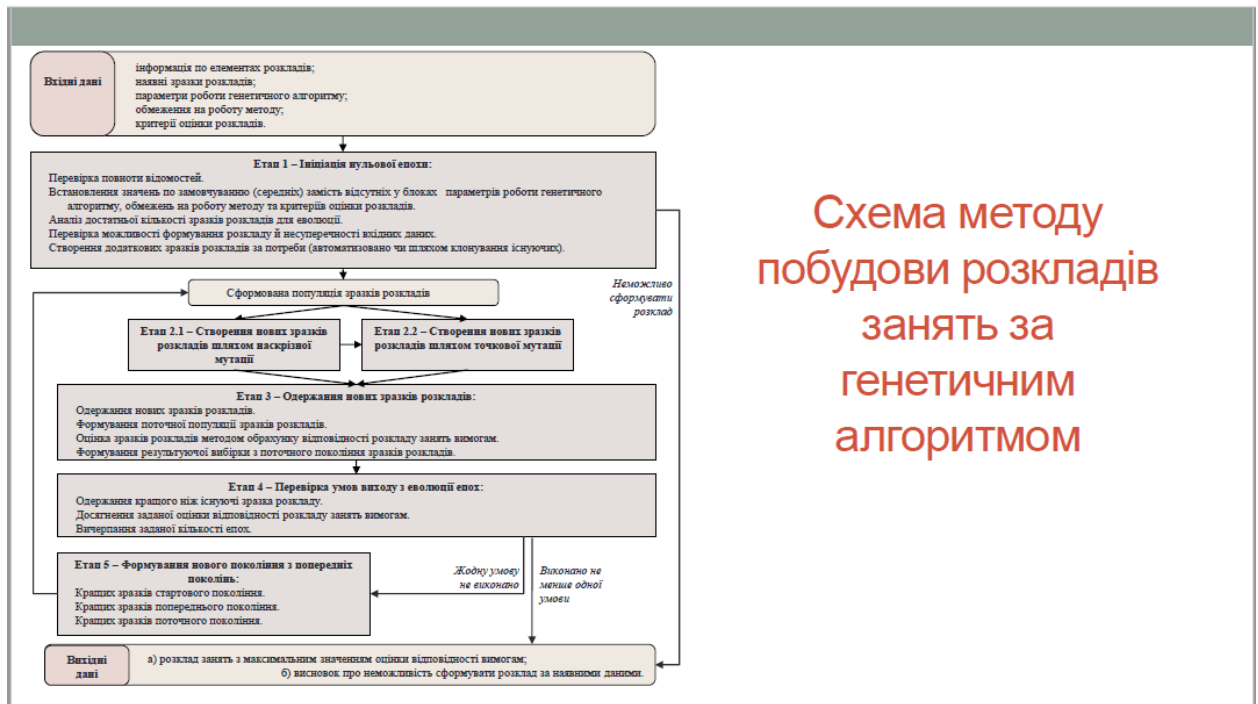
1. Провести аналіз відомих підходів до автоматизованої побудови розкладів занять й шляхів до їх удосконалення.
2. Вдосконалити інформаційну модель розкладу занять як вичерпного подання предметної області, достатнього для побудови розкладів занять за генетичним алгоритмом.
3. Удосконалити метод побудови розкладів занять за генетичним алгоритмом, який використовує математичну модель обрахунку відповідності розкладу занять вимогам.
4. Розробити математичну модель обрахунку відповідності розкладу занять вимогам.
5. Розробити інформаційну систему генерації розкладів занять за генетичним алгоритмом згідно створеного методу.
6. Провести прикладне та функціональне дослідження методу побудови розкладів занять за генетичним алгоритмом.

Інформаційна модель розкладу занять вищого навчального закладу

Інформаційна модель розкладу занять потрібна для формального подання сутностей предметної області в вигляді множин, що забезпечує подальше використання цих відомостей для розробки й реалізації методів та засобів автоматизації побудови розкладів занять за генетичним алгоритмом. Інформаційна модель розкладу занять вищого навчального закладу R має наступний вигляд:

$$\{V \cup A \cup T \cup M \cup G \cup Y \cup P \cup U \cup D \cup E \cup F \cup L \cup B \cup C \cup H \cup Q \cup K\} \subset R,$$

де V – множина викладачів, A – множина аудиторій, T – множина типів аудиторій і занять, M – множина входжень підгруп у потоки, G – множина груп, Y – множина підгруп, P – множина потоків, U – множина кафедр як включень викладачів і потоків до аудиторного фонду, D – множина днів тижня, E – множина вимог на планування пар (як комбінацій викладача, потоку, дисципліни та типу заняття), F – множина запланованих у розкладі пар (як комбінацій вимог на планування пари, аудиторії, дня тижня та номеру пари), L – множина пріоритетів викладачів до пар, B – множина пріоритетів викладачів до аудиторій, C – множина пріоритетів викладачів до днів тижнів, H – множина оцінок розкладів для груп, Q – множина оцінок розкладів для викладачів, K – множина вагових коефіцієнтів для оцінки розкладів.



Математична модель обрахунку відповідності розкладу занять вимогам

Умовна оцінка відповідності визначеного зразка розкладу занять вимогам визначеного викладача OV обраховується наступним чином:

$$OV = VA_{poz} - VA_{neg} + VD_{poz} - VD_{neg} + VP_{poz} - VP_{neg},$$

де VA_{poz} – кількість випадків пар у аудиторіях в розкладі згідно позитивних пріоритетів викладача, VA_{neg} – кількість випадків пар у аудиторіях в розкладі згідно негативних пріоритетів викладача, VD_{poz} – кількість випадків пар у дні тижня в розкладі згідно позитивних пріоритетів викладача, VD_{neg} – кількість випадків пар у дні тижня в розкладі згідно негативних пріоритетів викладача, VP_{poz} – кількість випадків пар з номером проведення в розкладі згідно позитивних пріоритетів викладача, VP_{neg} – кількість випадків пар з номером проведення в розкладі згідно негативних пріоритетів викладача.

Математична модель обрахунку відповідності розкладу занять вимогам

Умовна оцінка відповідності визначеного зразка розкладу занять вимогам визначеної підгрупи OP обраховується наступним чином:

$$OP = PD_{poz} - PD_{neg} + PT_{poz} - PT_{neg},$$

де PD_{poz} – кількість днів в розкладі у яких кількість пар не відрізняється від рекомендованої, PD_{neg} – кількість днів в розкладі у яких кількість пар відрізняється від рекомендованої, PT_{poz} – кількість тижнів (розглядаються окремо чисельник та знаменник) в розкладі у яких кількість пар не відрізняється від рекомендованої, PT_{neg} – кількість тижнів (розглядаються окремо чисельник та знаменник) в розкладі у яких кількість пар відрізняється від рекомендованої.

Математична модель обрахунку відповідності розкладу занять вимогам

Загальна умовна оцінка відповідності визначеного зразка розкладу занять вимогам O визначається наступним чином:

$$O = k_{OV} \sum_{i=1}^n OV_i + k_{OP} \sum_{j=1}^m OP_j - k_{VV} \sum_{i=1}^n VV_i - k_{VP} \sum_{j=1}^m VP_j,$$

де n – кількість викладачів у розкладі який аналізується, m – кількість підгруп у розкладі який аналізується, OV_i – умовна оцінка відповідності визначеного зразка розкладу занять вимогам визначеного викладача i , OP_j – умовна оцінка відповідності визначеного зразка розкладу занять вимогам визначеної підгрупи j , VV_i – кількість вікон в розкладі який аналізується у визначеного викладача i , VP_j – кількість вікон в розкладі який аналізується у визначеної підгрупи j , k_{OV} – ваговий коефіцієнт важливості врахування пріоритетів викладачів за розкладом,

k_{OP} – ваговий коефіцієнт важливості врахування пріоритетів потоків за розкладом,
 k_{VV} – ваговий коефіцієнт важливості відсутності вікон у викладача за розкладом,
 k_{VP} – ваговий коефіцієнт важливості відсутності вікон у підгрупи за розкладом.



Інформаційна система генерації розкладів занять за генетичним алгоритмом

Автоматизована система генерації розкладів

Викладачі та їх вимоги | Ресурс аудиторного фонду | Двошкільний план навчання | Академічні групи і потоки | Ручне створення розкладів та генератор розкладів | Параметри роботи генетичного алг...

Викладачі	ПІБ	Кафедра	Науковий ступінь	Вчене звання
▶	Скрипник Тетяна Казимирівна	Комп'ютерних наук	магістр	доцент
	Бармак Олександр Володимирович	Радотехніки	доктор філософії	доцент
	Медведчук Неля Казимирівна	Інженерної механіки	доктор філософії	доцент
	Косійчук Федір Макарович	Української класики	доктор гуманітарних наук	професор
	Фізіотропов Геннадій Гарфілоєвич	Технічної механіки	доктор технічних наук	професор
	Мулько Людмила Петрівна	Комп'ютерних наук	доктор філософії	доцент
	Козлов Теодор Степанович	Іноземної мови	доктор гуманітарних наук	професор
	Віталев Леонід Касіянович	Радотехніки	доктор філософії	доцент

Показати обраного викладача | Додати викладача | Пріоритет викладача щодо аудиторій | Пріоритет викладача щодо днів тижня | Пріоритет викладача

Обраний викладач

Прізвище: Скрипник
 Ім'я: Тетяна
 По-батькові: Казимирівна
 Кафедра: Комп'ютерних наук
 Науковий ступінь: магістр
 Вчене звання: доцент
 Посада: старший викладач

Регувати

Пріоритет викладача щодо аудиторій

№ аудиторії	Шир аудит.	Назва	Тип занять
▶ 2	1	3-403	Лекц. ауд.
3	2	3-406	Лаб. ауд.
4	3	3-404	Лаб. ауд.

Оберть пріоритет | Застосувати пріоритет для обраного

Інформаційна система генерації розкладів занять за генетичним алгоритмом

Автоматизована система генерації розкладів

Викладачі та їх вимоги Ресурс аудиторного фонду Двотижневий план навчання Академічні групи і потоки Ручне створення розкладів та генератор розкладів Параметри роботи генетичного алг

Викладачі

№	Дисципліна	Поток	ПІБ	Тип заняття	Неперервна кількість пар
1	Управління IT...	КН-19	Скрипник Тетяна Казимирівна	Лекція	1
2	Основи програ...	КН-17	Петровський Сергій Степанович	Лабораторне заняття	2
3	Жмари технол...	КН-18-1	Манзюк Едуард Андрійович	Лекція	1

Додання позиції в план

Дисципліна: Жмари технол
 Поток: КН-18-1
 Прізвище викладача: Манзюк
 Тип заняття: Лекція
 Неperервна к-сть пар: 1

Додати позицію в план

Редагувати обрану позицію
 Видалити позицію з плану

Завантажити існуючий план навчання
 План навчання: [випадковий]
 Зберегти
 Зберегти відомості для формування розкладу занять

Робота з реєстром викладачів
 Робота з реєстром потоків і груп
 Робота з реєстром аудиторій
 Робота з реєстром дисциплін

Дні тижня
 Пари
 Типи занять
 Типи аудиторій

Інформаційна система генерації розкладів занять за генетичним алгоритмом

Автоматизована система генерації розкладів

Двотижневий план навчання Академічні групи і потоки Ручне створення розкладів та генератор розкладів Параметри роботи генетичного алгоритму Автоматизована еволюція розкладів

Обраний план навчання: новий№35442
 Розклад пар: поточний№44
 Оцінка: 33,00381739

Версії розкладу по плану

№	Назва розкладу	Оцінка
1	новий№35442	44,3261983
2	поточний№44	33,00381739
3	новий№35443	55,248401

№	День тижня	№ пари	Аудиторія	Дисципліна	Поток	Викладач	Тип заняття	Неперервна к-ть пар
1	понедлок	1	3-402	Управління IT...	КН-19	Скрипник Т...	Лекція	1
2	понедлок	2	3-402	Жмари технол...	КН-18-1	Манзюк Еду...	Лабораторне	2
3	вівторок	2	3-404	Основи прог...	КН-17	Петровський	Лабораторне	3

Виконати епоху
 Виконати 10 епох або знайти кращий розклад за існуючі
 Зберегти обраний розклад як...

Можливості прикладного використання деяких функцій інформаційної системи для побудови розкладів занять за генетичним алгоритмом

Функції	Розроблені шляхи забезпечення функцій
Ручне створення розкладів занять	Інформаційна модель розкладу занять та структура БД, що забезпечують необхідну та достатню формалізацію й структуру сутностей предметної області
Автоматизоване створення розкладів занять	При виконанні методу побудови розкладів занять за генетичним алгоритмом у такий спосіб забезпечується створення початкових зразків розкладів занять
Автоматизована оцінка наявних розкладів занять	Шляхом прикладного використання розробленої математичної моделі обрахунку відповідності розкладу занять вимогам
Автоматизоване порівняння наявних розкладів занять	Шляхом порівняння оцінок відповідності вимогам цих розкладів занять, використовується при виконанні методу побудови розкладів занять за генетичним алгоритмом для пошуку кращих зразків
Автоматизований пошук шляхів покращення наявних розкладів занять	Шляхом використання операторів точкової та наскрізної мутації при роботі за генетичного алгоритму в методі побудови розкладів занять
Врахування вимог викладачів до аудиторій, днів тижня та номерів пар у розкладі	Врахування за рахунок виставлення пріоритетів викладачів до аудиторій, днів тижня та номерів пар
Врахування вимог підгрупи (групи) до кількостей пар в тиждень та поденно	Врахування за рахунок виставлення рекомендованих підгрупі кількостей пар в тиждень та поденно
Баланс груп вимог до розкладів викладачів і підгруп	Врахування за рахунок вагових коефіцієнтів врахування при оцінці розкладів важливості пріоритетів викладачів, пріоритетів потоків, відсутності вікон у викладача й у підгрупи
Врахування вимог викладачів і підгруп до відсутності вікон у розкладі	Врахування за рахунок зменшення умовної оцінки розкладу при наявності вікон у викладачів і підгруп
Врахування нормативних значень по кількості пар в день та тиждень при створенні розкладу занять	Врахування за рахунок встановлення обмежень на роботу методу (максимальна кількість пар в день у викладача, максимальна кількість пар в тиждень у викладача, максимальна кількість пар в день у групи, максимальна кількість пар в тиждень у групи)

Практичне значення одержаних результатів

Практичне використання розробленої системи підтвердило спроможність інформаційної системи забезпечити виконання функцій побудови розкладів занять за генетичним алгоритмом. Можливості розробленого методу побудови розкладів занять за генетичним алгоритмом і супутніх засобів забезпечують виконання ряду важливих функцій при створенні розкладів занять, що підкреслює практичну значущість одержаних результатів.

Зокрема, забезпечується ручне створення розкладів занять, автоматизоване створення розкладів занять, автоматизована оцінка наявних розкладів занять, автоматизоване порівняння наявних розкладів занять, автоматизований пошук шляхів покращення наявних розкладів занять, врахування вимог викладачів до аудиторій, днів тижня та номерів пар у розкладі, врахування вимог підгрупи (групи) до кількостей пар в тиждень та поденно, баланс груп вимог до розкладів викладачів і підгруп, врахування вимог викладачів і підгруп до відсутності вікон у розкладі та врахування нормативних значень по кількості пар у день і тиждень при створенні розкладу занять.

Визначними рисами розробленого методу побудови розкладів занять за генетичним алгоритмом є врахування при оцінюванні зразків розкладів таких критеріїв, як пріоритети викладачів (аудиторії, дні тижня, пари), пріоритети потоків (рекомендовані кількості пар у день і в тиждень), відсутність вікон в викладача і підгрупи, а також врахування обмежень на роботу алгоритму таких як максимальна кількість пар в день і у тиждень у викладача і максимальна кількість пар в день і в тиждень в підгрупі.

Загальні висновки

Кваліфікаційна робота магістра розв'язує науково-технічну задачу автоматизації побудови розкладів занять за генетичним алгоритмом, який дозволяє із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму.

У результаті виконання роботи були поставлені та вирішені наступні завдання:

1. Проведено аналіз відомих підходів до автоматизованої побудови розкладів занять й шляхів до їх удосконалення.
2. Вдосконалено інформаційну модель розкладу занять як вичерпного подання предметної області, достатнього для побудови розкладів занять за генетичним алгоритмом.
3. Удосконалено метод побудови розкладів занять за генетичним алгоритмом, який використовує математичну модель обрахунку відповідності розкладу занять вимогам.
4. Розроблено математичну модель обрахунку відповідності розкладу занять вимогам.
5. Розроблено інформаційну систему генерації розкладів занять за генетичним алгоритмом згідно створеного методу.
6. Проведено прикладне та функціональне дослідження методу побудови розкладів занять за генетичним алгоритмом.

Положення новизни та інновації

- **Вдосконалено інформаційну модель розкладу занять**, яка відрізняється тим, що містить подання всіх необхідних сутностей для побудови розкладів занять за генетичним алгоритмом, включаючи параметри роботи генетичного алгоритму, обмеження на компоненти розкладу занять та критерії оцінки розкладів.
- **Вдосконалено метод побудови розкладів занять за генетичним алгоритмом**, який використовує математичну модель обрахунку відповідності розкладу занять вимогам і відрізняється тим, що дозволяє за наявними в інформаційній моделі розкладу занять параметрами із урахуванням вимог викладачів та академічних груп до параметрів проведення пар автоматизовано формувати розклади занять для вишів за допомогою еволюційного генетичного алгоритму.
- **Розроблено нову інформаційну систему генерації розкладів занять за генетичним алгоритмом**, що дозволяє з використанням створених інформаційної моделі розкладу занять, методу побудови розкладів занять за генетичним алгоритмом та математичної моделі обрахунку відповідності розкладу занять вимогам виконувати ручне та автоматизоване створення нових розкладів, формувати нові похідні розклади із використанням генетичного алгоритму, оцінювати сформовані розклади занять із урахуванням вимог викладачів та академічних груп до параметрів проведення пар й визначати кращі зразки розкладів занять за результатом порівняння їх оцінок відповідності вимогам.

Ім'я користувача:
Кафедра КН

Дата перевірки:
05.12.2021 19:44:16 EET

Дата звіту:
05.12.2021 19:52:35 EET

ID перевірки:
1009530320

Тип перевірки:
Doc vs Internet + Library

ID користувача:
100005671

Назва документа: 2021_KPM_Богач 20211205 1 АНТИПЛАГІАТ

Кількість сторінок: 96 Кількість слів: 16797 Кількість символів: 129340 Розмір файлу: 4.84 MB ID файлу: 1009540885

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

6.61% Схожість

Найбільша схожість: 4.41% з джерелом з Бібліотеки (ID файлу: 1009540880)

0.53% Джерела з Інтернету 24 Сторінка 98

6.26% Джерела з Бібліотеки 94 Сторінка 98

3.93% Цитат

Цитати 28 Сторінка 99

Посилання 1 Сторінка 100

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи 13

Підозріле форматування 25 сторінок

Anti-Plagiarism v-15.257

Максимальное совпадение с одним документом 4.0%

Словари проверки: en_US, ru_RU, ua_UA. **Ошибок в документах: 7%**

ID: 98043 Название: Метод побудови розкладів занять за генетичним алгоритмом Добавлено в БД: 2021-12-05 Авторы: В.В. Богач Руководители: О.В. Мазурець Консультанты: Опоненты:	Документ		Суммарное совпадение по Базе Данных	
	Символы	Лексемы	Символы	Лексемы
	113508	806	9703 (9%)	76 (9%)

Источник плагиата

ID	Описание	Наличие плагиата в документе	
		Символы	Лексемы

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ
КАФЕДРИ КОМП'ЮТЕРНИХ НАУК
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ МАГІСТРА ДО ЗАХИСТУ
ЗА РЕЗУЛЬТАТАМИ АНАЛІЗУ ЗВІТУ ПОДІБНОСТІ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Метод побудови розкладів занять за генетичним алгоритмом

Автор: Богач Владислав Валерійович

Спеціальність: 122 – Комп'ютерні науки

Освітня програма: освітньо-професійна

Науковий керівник: к.т.н., доц.каф.КН Мазурець Олександр Вікторович

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	—
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	—
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	—

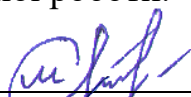
Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

- 1) За програмою Anti-Plagiarism виявлені 9%, які є фрагментарними, не більше 4% на джерело – містять поширені конструкції, загальновідомі терміни та визначення.
- 2) За програмою UNICHECK виявлені 6,61%, які є фрагментарними, не більше 4,41% на джерело – містять поширені конструкції, загальновідомі терміни та визначення.

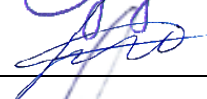
Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 9% і 6,61% відповідно, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи



Олександр Мазурець

Гарант ОП



Руслан Багрій

Завідувач кафедри КН



Олександр Бармак



ВІДГУК ОПОНЕНТА

на кваліфікаційну роботу магістра

гр. КНм-20-1 Богача Владислава Валерійовича за темою: Метод побудови розкладів занять за генетичним алгоритмом

1. Актуальність обраної теми

Оптимальним розклад є тоді, коли він повністю покриває зв'язок навчальних дисциплін під час процесу навчання, задовольняє певним умови та обмеження, а також враховує досвід викладачів. Якщо при складанні розкладу не враховані всі зв'язки між дисциплінами, або не враховано всі обмеження, то це впливає на якість навчального процесу, що звісно погіршує рівень підготовки студентів. Тому робота, присвячена генерації розкладів занять за генетичним алгоритмом, є актуальною та перспективною, що належним чином обґрунтовано у роботі.

2. Відповідність роботи предметній області спеціальності 122 Комп'ютерні науки та загальним вимогам до наукових робіт

Обрана тема автоматизованої побудови розкладів занять за генетичним алгоритмом, в межах якої реалізовані поставлені задачі, повною мірою відповідає предметній області спеціальності 122 «Комп'ютерні науки» та вимогам до кваліфікаційної роботи магістра.

3. Повнота розкриття мети та завдань дослідження

В роботі повністю розкрито мету дослідження та поставленні в межах теми завдання дослідження.

4. Наявність наукової новизни

В кваліфікаційній роботі представлена наукова новизна та інновації, відповідні спеціальності 122 «Комп'ютерні науки» в межах обраної області дослідження. Продемонстровано й обґрунтовано результати, які мають наукове та інноваційне значення. Результати дослідження оприлюдненні на науково-практичній конференції.

5. Зміст кожного розділу роботи

Робота містить чотири розділи. У першому розділі обґрунтовано актуальність автоматизованої побудови розкладів занять за генетичним алгоритмом та поставлені задачі дослідження. Другий розділ присвячено розробці методу побудови розкладів занять за генетичним алгоритмом і супутніх засобів. У третьому розділі виконано розробку

інформаційної системи генерації розкладів занять за генетичним алгоритмом. У четвертому розділі виконано дослідження ефективності розробленого методу побудови розкладів занять за генетичним алгоритмом.

6. Ступінь розкриття теми роботи

Тема роботи в повній мірі обґрунтована й розкрита, проведено аналіз актуальності та відомих досліджень в межах обраної теми, поставлені завдання, які у роботі виконані, та проведено аналіз результатів прикладного застосування запропонованих методу і засобів.

7. Якість оформлення кваліфікаційної роботи

Оформлення роботи відповідає необхідним нормам та вимогам, які ставляться до оформлення кваліфікаційних робіт.

8. Недоліки кваліфікаційної роботи

У роботі відсутні суттєві недоліки. Кілька із використаних джерел є застарілими, втім це не вплинуло на загальну якість роботи та одержаний результат.

9. Загальний висновок (допускається чи не допускається до захисту), якої оцінки заслуговує кваліфікаційна робота

Враховуючи рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка відмінно.

Опонент _____  _____ к.т.н., доцент Пивовар Олег Сергійович



ВІДГУК НАУКОВОГО КЕРІВНИКА

на кваліфікаційну роботу магістра

гр. КНм-20-1 Богача Владислава Валерійовича за темою: Метод побудови розкладів занять за генетичним алгоритмом

1. Актуальність теми

Для забезпечення нормального функціонування всіх складових навчального процесу складаються розклади. Особливо важливо це для вищих навчальних закладів, так як групи студентів не мають закріплених за ними аудиторій як, наприклад, в школі. Розклад навчальних занять є документом, що засвідчує трудовий режим викладачів, й оптимальний розклад позитивно впливає на працездатність як викладачів, так і студентів. Тому робота, присвячена автоматизованій генерації розкладів занять за генетичним алгоритмом, є беззаперечно актуальною.

2. Відповідність роботи предметній області спеціальності 122 Комп'ютерні науки та загальним вимогам до наукових робіт

Поставлена у кваліфікаційній роботі магістра мета, пов'язана з автоматизацією формування розкладів занять за генетичним алгоритмом, цілком відповідає предметній області спеціальності 122 «Комп'ютерні науки» та вимогам до кваліфікаційної роботи.

3. Професійні та особистісні якості магістранта

При роботі над кваліфікаційною роботою магістра Богача Владислава Валерійовича зарекомендував себе кваліфікованим та дисциплінованим студентом і фахівцем, вчасно та пунктуально виконуючи необхідні етапи дослідження. Як в процесі наукових вишукувань, так і за розробки прикладного програмного забезпечення проявив високі і достатні для одержання успішного результату компетентності.

4. Ступінь самостійності під час виконання кваліфікаційної роботи

Магістрант самостійно виконував всі поставлені задачі. Одержані положення наукової новизни та інновації, означені в роботі, є результатом особистої діяльності магістранта. Це дозволило провести створення нових та удосконалення існуючих теоретичних та прикладних засобів, створених та використаних у роботі.

5. Наукова новизна та оригінальність запропонованих підходів

В кваліфікаційній роботі магістра представлена наукова новизна та інновації, відповідні спеціальності 122 «Комп'ютерні науки» в межах обраної області дослідження.

Продемонстровано й обґрунтовано результати, які мають наукове та інноваційне значення. Вдосконалено інформаційну модель розкладу занять, вдосконалено метод побудови розкладів занять за генетичним алгоритмом, розроблено нову інформаційну систему генерації розкладів занять за генетичним алгоритмом. Результати роботи оприлюдненні на науково-практичній конференції.

6. Ступінь оволодіння методами дослідження

В роботі виявлено високий ступінь оволодіння магістрантом необхідними методами дослідження.

7. Повнота та якість розкриття теми роботи

Тема роботи в повній мірі обґрунтована й розкрита, проведено аналіз актуальності та відомих досліджень в межах обраної теми, поставлені завдання, які у роботі виконані, а також проведено аналіз результатів прикладного застосування запропонованих засобів автоматизації формування розкладів занять за генетичним алгоритмом.

8. Логічність, послідовність, аргументованість, літературна грамотність викладу матеріалу

Структура роботи й послідовність викладення логічні та відповідні поставленій меті. Викладення матеріалу грамотне та виявляє високий ступінь відповідності стилю.

9. Можливість практичного застосування кваліфікаційної роботи, окремих її частин

Практичне використання розробленої інформаційної системи підтвердило її спроможність забезпечити виконання функцій побудови розкладів занять за генетичним алгоритмом. Визначними рисами розробленого методу побудови розкладів занять за генетичним алгоритмом є врахування при оцінюванні розкладів таких критеріїв, як пріоритети викладачів і потоків до аудиторії, днів тижня та пар, відсутність вікон в викладачів і підгруп, а також врахування обмежень на роботу алгоритму таких як максимальна кількість пар в день і у тиждень у викладача і в підгрупи.

10. Висновок про можливість допуску кваліфікаційної роботи до захисту, на яку оцінку заслуговує робота

Враховуючи високий рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка відмінно.

Науковий керівник  к.т.н., доц.каф.КН Мазурець Олександр Вікторович