

**КУРСОВА РОБОТА**

**Група ІП-22-1**

**ІП-220424**

**Токарєв Сергій**

**2023**

Міністерство освіти і науки України  
Івано-Франківський національний технічний університет нафти і газу  
Інститут інформаційних технологій

## КУРСОВИЙ ПРОЄКТ

з дисципліни "Об'єктно-орієнтоване програмування"

на тему: Програмна автоматизація інформаційної діяльності в  
предметній області «Міської філармонії» засобами C++/ООП шляхом  
побудови програмного рішення

Студента 1 курсу 1 групи  
напряму підготовки спеціальності  
Інженерія програмного забезпечення  
Токарєв С.О.

Керівник:

Доцент Шекета В.І

Національна шкала: \_\_\_\_\_

Кількість балів: \_\_\_\_ Оцінка: ECTS: \_\_\_\_

Члени комісії

\_\_\_\_\_  
(підпис) (прізвище та ініціали)

\_\_\_\_\_  
(підпис) (прізвище та ініціали)

\_\_\_\_\_  
(підпис) (прізвище та ініціали)

м. Івано-Франківськ

2023 рік

Міністерство освіти і науки України

Івано-Франківський національний технічний університет нафти і газу

Інститут інформаційних технологій

ЗАТВЕРДЖУЮ

зав. кафедри ІПЗ, проф., л.т.н.

\_\_\_\_\_ **В. І. Шекета**

«\_\_\_» \_\_\_\_\_ **2023 р.**

### ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

На курсовий проект з дисципліни «ООП» студенту Токарєву Сергію Олександровичу групи ІІ-22-1.

ТЕМА: Розробка прикладної програми з проєктування інформаційних систем згідно завдання.

Постановка задачі.

1. Розробити прикладну програму.
2. Розробити інтерфейс користувача.
3. Розробити та використати активні елементи.
4. Розробити та використати шаблони не менше 5 шт.
5. Розрахувати швидкодію роботи з інтерфейсом користувача.
6. Адаптувати розроблене прикладне програмне забезпечення до різних роздільних здатностей цифрових дисплеїв.

Дата видачі: «\_\_\_» 2023 р.

Керівник \_\_\_\_\_ **Шекета В.І.**

Завдання отримав \_\_\_\_\_ **Токарєв С.О.**

# Зміст

## 1. ТЕОРЕТИЧНА ЧАСТИНА

ВСТУП.....	5
1.1 Опис динамічних та статичних властивостей предметної області.....	6
1.2 Виділення та опис об'єктів в предметній області.....	8

## 2. ТЕХНІЧНА ЧАСТИНА

2.1. Аналіз сучасних тенденцій до оформлення проекту з проектування інформаційних систем.....	11
2.2. Положення меню.....	14
2.3. Особливості розробки активних програмних елементів.....	15
2.4. Особливості розробки активних програмних шаблонів.....	15
2.5. Розрахунок якості інтерфейсу користувача.....	16
2.6. Розрахунок швидкості заповнення діалогових форм GOMS.....	19
2.7. Моделювання та тестування інформаційної системи (ІС).....	20
ВИСНОВКИ.....	22
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	23
ДОДАТОК А.....	25
ДОДАТОК Б.....	26
ДОДАТОК В.....	34

					КР.ІІІ-220424 ІІЗ				
					Розробка прикладної програми «АІС міської філармонії»	Літ.	Лист	Листів	
Змн	Лист	№ докум.	Підпис	Дата			3	51	
Розроб.	Токарев С.О.					ІФНТУНГ,ІП-22-1			
Перевір.	Шекета В.І.								
Реценз.									
Н. Контр.									
Затверд.									

# ТЕОРЕТИЧНА ЧАСТИНА

## Вступ

Мета цієї курсової роботи – застосувати свої вміння проектування та конфігурування інформаційних систем у виконанні індивідуального курсового проекту. Постановка задачі – моделювання, аналіз та створення автоматизованої інформаційної системи (АІС) організації засобами об’єктно-орієнтованої мови програмування. За власним бажанням була обрана мова C++, адже вона не тільки дозволить використати концепції ООП, а й краще освоїти операції з фіксованою та динамічною пам’яттю.

Ефективне управління підприємством в сучасному світі неможливе без втручання технологій. Надійний вибір програмного продукту й компанії-розробника – це найважливіший етап автоматизації бухгалтерського обліку.

Тому проблема вибору інформаційної системи (ІС) із певним завданням сьогодні являється стандартною процедурою. Ці ІС швидко розвиваються та стають все більш складними та досконалими. Проте технології їхньої розробки також вдосконалюються, з метою спростити процес створення таких систем. Останніми роками виникли мови програмування з підтримкою або повністю базовані на ООП; сервіси, що впроваджують системи керування версіями; програмне забезпечення для візуалізації та віртуалізації; забезпечення для тестування ІС.

Виконання даного курсового проекту направлено на закріплення знань, отриманих при вивченні теоретичної частини курсу, що дасть практичні вміння зі створення класів та баз даних. Створення бази даних завжди було і буде актуально, адже майже всі інформаційні системи використовуються не тільки для оброблення даних, але й для їх збереження. І оскільки кількість даних з часом збільшується, бази даних змушені вдосконалюватись.

					ІП-220424	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		

## 1.1 Опис динамічних та статичних властивостей предметної області

Предметна область — це частина реального світу, що розглядається в межах певного контексту. Під контекстом можна розуміти область дослідження чи область, яка є об'єктом певної діяльності. Предметна область представляється безліччю фрагментів. Усі фрагменти предметної області безліччю об'єктів і процесів, що використовують об'єкти, і безліччю користувачів, які мають різні погляди на предметну область.

Ключовим поняттям в об'єктно-орієнтованому програмуванні на C++ є поняття класу. Загалом, певному класу притаманні ті чи інші властивості. Вони представляють собою ту чи іншу змінну, однак при різних умовах, на відмінну від змінної, властивість може повертати різні значення. Класи - це розширене поняття структур даних : як і структури даних, вони можуть містити члени даних, але вони також можуть містити функції як члени. Об'єкт є конкретизацією класу. З точки зору змінних, клас буде типом, а об'єкт - змінною. Фактично – це визначений програмістом нестандартний тип даних, тому поняття полів і методів класу повністю співпадають з аналогічними поняттями абстрактного типу даних.

Інфраструктура міської філармонії представлена культурними спорудами різного типу: театри, концертні майданчики, естради, палаци культури тощо. Кожна з категорій культурних споруд має атрибути, специфічні тільки для неї: театр характеризується місткістю, кінотеатр - розміром екрана.

Артисти під керівництвом імпресарію (агенту артиста) виступають у різних жанрах, при цьому один і той же артист може виступати в кількох жанрах, і може працювати з декількома імпресарію. Організатори концертних заходів проводять виступи, концерти, конкурси в культурних спорудах міста, організовуючи участь у ньому артистів. За результатами участі артистів у конкурсах проводиться нагородження.

					ІП-220424	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

Спеціалізоване програмне забезпечення локального масштабу повинне автоматизувати та контролювати бізнес-процеси, зберігати інформацію про акторів, персонал та інші об'єкти міської філармонії.

Дане програмне забезпечення сприятиме подальшому розвитку та масштабуванню організацій міста, а також збільшенню прибутку.

Це – комплексне рішення для індустрії мистецтва , яке охоплює оперативний облік в управлінні молоді та творчості Івано-Франківська.

Наступні переваги автоматизації:

- контроль за ефективністю проведення тих чи інших заходів;
- оцінка якості роботи споруд;
- ведення обліку діючих інфраструктур;
- оптимізація кількості акторів;
- аналіз ефективності роботи спонсорського комітету;
- контроль за викладацької роботи викладачів;
- скорочення витрат;
- можливість проводити складні розрахунки автоматично;
- локалізація всієї інформації про діяльність організацій в одному місці та можливість використовувати її для прийняття поточних і стратегічних рішень.

Автоматизація особлива необхідна для автопідприємства за наявності таких факторів:

- ✓ вся інформація знаходиться в окремих файлах;
- ✓ існує залежність виконання поставленого плану на рік;
- ✓ постійне дублювання одних і тих самих дій;

Отже, мистецькі організації потребують якісного програмного забезпечення, для того щоб автоматизувати та покращити якість праці.

					ІІІ-220424	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

Ці продукти вже в базовій версії повинні покривати всі бізнес-процеси , а також гнучко підлаштовуватися під певні умови. Також важливі такі характеристики як:

- зручний інтерфейс;
- можливість масштабування;
- подальша підтримка та розвиток продукту;
- вартість володіння продуктом;
- висока продуктивність;
- можливість інтеграції з іншими ІС.

## 1.2 Виділення та опис об'єктів в предметній області

Реалізація баз даних ґрунтується на виділенні об'єктів предметної області та структуризації документів, де ці об'єкти, ґрунтуючись на логічному аналізі атрибутів, можна виділити.

Предметна область (бази даних) — це сфера пов'язаних між собою функцій, за допомогою яких досягається виконання поставлених цілей і застосування конкретної бази даних. Кожна предметна область має фрагменти, а самі фрагменти містять в собі процеси і об'єкти, що використовують об'єкти, а також безліччю користувачів, що характеризуються різними поглядами на предметну область.

Проектування баз даних включає кілька етапів:

- Проектування інфологічної концептуальної моделі БД;
- Проектування даталогічної моделі БД.;
- Проектування фізичної моделі БД;
- Реалізація БД.

Кожен етап має на увазі послідовне проектування бази даних і включає ряд особливостей. Інфологічна модель відображає реальний світ у деякі зрозумілі людині концепції, повністю незалежні від параметрів середовища зберігання даних. Це узагальнене неформальний опис створюваної бази даних.

					ІП-220424	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		



Даталогічна модель відображає логічні зв'язки між елементами даних, незалежно від їх змісту і середовища зберігання. Даталогічне моделювання передбачає опис, створюване за інфологічної моделі даних.

Фізична модель даних - модель, яка визначає розміщення даних на зовнішніх носіях, методи доступу і техніку індексування. Вона так само називається внутрішньою моделлю системи, яка визначає і оперує розміщенням даних і їх взаємозв'язками на запам'ятовуючих пристроях.

Згідно вибраного варіанту, для аналізу моєї предметної області вибрано культурні споруди міста Івано-Франківська. Інфраструктура мистецтва представлена культурними спорудами різного типу: театри, концертні майданчики, естради, палаци культури тощо. Кожна з категорій культурних споруд має атрибути, специфічні тільки для неї: театр характеризується місткістю, кінотеатр - розміром екрана.

Для реалізації поставленої задачі потрібно ввести систему класів. Клас можна реалізувати 2 способами:

- 1) створити окремі класи для кожного виду запиту і використовувати наслідування класів;
- 2) створити один клас, який буде зберігати інформацію про різні види об'єктів.

У даному курсовому проєкті я обрав другий спосіб, оскільки це дає можливість ефективно побудувати складні ієрархічні класи з можливістю їх зручної модифікації. Роботу класів в ієрархії можна змінювати шляхом додавання нових успадкованих класів у потрібному місці ієрархії. Новий клас можна отримати від декількох базових класів. Такий процес називається множинним наслідуванням.

Наслідування – це спосіб реалізації нових класів, основою якого є основний клас. Клас, з якого унаслідується новий клас успадковує функціонал та параметри «батьківського» класу. При успішній реалізації похідного класу, він може бути доповненим власними властивостями та методами.

C++ дозволяє інкапсулювати дані. Інкапсуляція (або «приховування інформації») – це процес прихованого зберігання деталей реалізації об'єкта.

					ІП-220424	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

Користувачі звертаються до об'єкта через відкритий інтерфейс. Абстракція даних – це механізм викриття тільки інтерфейсів і приховування деталей реалізації від користувача. інкапсуляція реалізована через специфікатор доступу. Як правило, всі змінні-члени класу є закритими, а більшість методів є відкритими (з відкритим інтерфейсом для користувача).

Інкапсуляція має багато переваг:

1. Інкапсульовані класи простіші у використанні і зменшують складність програм;
2. інкапсульовані класи допомагають захистити дані і запобігають їх неправильному використанню;
3. інкапсульовані класи легше змінити;
4. інкапсульовані класи легше налагодити.

Інкапсуляція здійснюється специфікаторами доступу. Специфікатор доступу визначає, хто має доступ до членів цього специфікатора. Кожен з членів має свій рівень доступу відповідно до специфікатора доступу (або, якщо він не вказаний, відповідно до специфікатора доступу за замовчуванням). Класи мають той самий формат, що і звичайні структури даних, за винятком того, що вони також можуть включати функції та мати ці нові речі, які називаються специфікаторами доступу. Специфікатор доступу є один з наступних трьох ключових слів: `private`, `public` або `protected`.

Ці специфікатори змінюють права доступу для членів, які слідують за ними:

- У розділах із заголовками `public` розміщуються загальнодоступні поля та методи, які використовуються для інтерфейсу об'єктів даного класу з програмою.
- У розділах із заголовками `private` розміщуються закриті дані, доступ до яких може бути здійснений лише за допомогою методів самого класу або за допомогою друзів класу
- У розділах із заголовками `protected` учасники доступні від інших членів того ж класу(або від їх "друзів"), а також від членів їх похідних класів.

					ІП-220424	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

# ТЕХНІЧНА ЧАСТИНА

## **2.1 Аналіз сучасних тенденцій до оформлення проекту з проектування інформаційних систем**

Сучасний стан інформаційних систем та технологій можна охарактеризувати наступними тенденціями:

- Наявність великої кількості промислово функціонуючих баз даних великого обсягу, що містять інформацію практично по всіх видах діяльності суспільства.
- Створення технологій, що забезпечують інтерактивний доступ масового користувача до цих інформаційних ресурсів. Технічною основою даної тенденції з'явилися державні та приватні системи зв'язку та передачі даних загального призначення і спеціалізовані, об'єднані в національні, регіональні і глобальні інформаційно – обчислювальні мережі.
- Розширення функціональних можливостей інформаційних систем, що забезпечують паралельну одночасну обробку баз даних з різноманітною структурою даних, мультиоб'єктних документів, гіперсередовища, в тому числі реалізують технології створення та ведення гіпертекстових баз даних. Створення локальних, багатофункціональних проблемно-орієнтованих інформаційних систем різного призначення на основі потужних персональних комп'ютерів і локальних обчислювальних мереж.
- Включення в інформаційні системи елементів інтелектуалізації інтерфейсу користувача, експертних систем, систем машинного перекладу, автоіндексування та інших технологічних засобів.

В наслідок чого, виділяють п'ять основних тенденцій у розвитку інформаційних систем та технологій:

					<b>ІП-220424</b>	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

Ускладнення інформаційних продуктів (послуг). Інформаційний продукт у вигляді програмних засобів, баз даних і служб експертного забезпечення набуває стратегічного значення.

Здатність до взаємодії. З ростом значущості інформаційного продукту можливість провести ідеальний обмін цим продуктом між комп'ютером і людиною або між інформаційними системами набуває значення ведучої технологічної проблеми. Також ця проблема стосується сумісності технічних і програмних засобів. Всі проблеми обробки і передачі інформаційного продукту перебували в повній відповідності щодо сумісності і швидкодії.

Ліквідація проміжних ланок. Розвиток здатності до взаємодії веде до вдосконалення процесу обміну інформаційним продуктом, а отже, при взаємовідносини постачальників і споживачів у цій області ліквідуються проміжні ланки.

Не потрібні посередники, якщо є можливість розміщувати замовлення безпосередньо за допомогою інформаційних технологій.

Глобалізація. Фірми можуть за допомогою інформаційних технологій вести справи де завгодно, отримуючи вичерпну інформацію. Глобалізація ринку інформаційного продукту націлена на отримання переваг за рахунок розподілу постійних і напівпостійних витрат на більш широкий географічний регіон.

Конвергенція. Зникають відмінності між виробами та послугами, інформаційним продуктом та засобами, використанням в побуті і для ділових цілей, інформацією і розвагою, а також серед різних режимів роботи, таких як передача звукових, цифрових і відеосигналів.

					ІІ-220424	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

Стосовно до бізнесу ці тенденції призводять до:

- здійснення розподілених персональних обчислень, коли на кожному робочому місці достатньо ресурсів для обробки інформації в місцях її виникнення;
- створення розвинених систем комунікацій, коли робочі місця з'єднані для пересилання повідомлень;
- гнучким глобальним комунікаціям, коли підприємство включається у світовий інформаційний потік;
- створення та розвитку систем електронної торгівлі;
- усунення проміжних ланок у системі інтеграції організація – зовнішнє середовище.

					<b>ІІІ-220424</b>	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

## 2.2 Положення меню:

**Меню** – це список певних команд, які користувач може виконати в даний момент часу під час роботи, в нашому випадку, з автоматизованою інформаційною системою (ІС).

Меню створеної консольної програми складається з 12 пунктів та має такий вигляд:

```

      MENU
-----
      АІС МІСЬКА ФІЛАРМОНІЯ
/1/ Культурні споруди
/2/ Список артистів по жанрах
/3/ Список артистів та імпресаріо
/4/ Артисти в деяких жанрах
/5/ Імпресаріо артиста
/6/ Концертні заходи
/7/ Список призерів конкурсів
/8/ Заходи в культурних спорудах
/9/ Список імпресаріо за жанром
/10/ Артисти-конкурсанти
/11/ Список організаторів заходів
/12/ Культурні споруди
/13/ Exit
-----
-----Виконав-----
-----Студент групи ІП-22-1-----
----- ІП - 220424-----
- Токарев Сергій Олександрович -
-----

Ваш вибір:
```

## 2.3 Особливості розробки активних програмних елементів

Під терміном «активні програмні елементи» в даній роботі будемо вважати певні елементи інформаційної системи, взаємодіяти напрямую з якими можна через розроблений інтерфейс.

Ці елементи створені для того, щоб користувач даної інформаційної системи зміг повноцінно працювати з нею, не створюючи «непорозумінь» між програмою та ним безпосередньо.

Вони унеможливають неправильну роботу програми шляхом перевірки даних відповідно до заданих умов.

Прикладом застосування активних програмних елементів в розробленому мною середовищі може слугувати динамічний вивід даних, форматування дати або ж перевірка вибору користувача системою на те чи така позиція взагалі існує (мається на увазі вибір пунктів та підпунктів), якщо в системі такого немає, вона одразу нам про це повідомляє.

## 2.4 Особливості розробки активних програмних шаблонів

Шаблони, або ж патерни, являють собою оптимізовані, універсальні вирішення для проблем програмування, які зустрічаються найбільше.

Він не є класом або ж бібліотекою, які можна просто включити в нашу систему – це поняття ширшого характеру, а саме архітектура, яка може бути застосована у відповідному випадку. Шаблон хорошої якості повинен бути сумісним з більшістю мов програмування.

Але потрібно розуміти, що у шаблонів є дві сторони медалі, адже, якщо їх застосувати не за призначенням та не в тому місці, це може створити чимало проблем в майбутньому.

Також, важливими активні шаблони, які дають можливість коректно відображати введені дані, що збільшує рівень розуміння програми користувачем там зменшує кількість можливих помилок.

					ІІІ-220424	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

Беручи до прикладу проект даної курсової роботи, тут я використовував один з основних елементів створення активного програмного шаблону – табуляцію. За допомогою неї дана інформаційна система виглядає естетичною та зрозумілою для користувача.

Наприклад, під час вводу інформації про будь-який об'єкт інформаційної системи, за допомогою «\n» або ж «endl», кожен наступний необхідний запит відображається з нового рядка, так само і при виводі інформації: нова характеристика – новий рядок. Також, слід звернути увагу на ввід та вивід ПІБ певних об'єктів нашої роботи. Адже в цьому випадку, програмний код зроблений так, щоб для зручності користувача ці три змінні (прізвище, ім'я, по батькові) вводились підряд через пробіл.

Також, щоб не захаращувати екран, після кожного виконаного запиту, або ж підзапиту, виконується очищення консолі за допомогою system("cls").

Отже, активні програмні шаблони допомагають покращити процес взаємодії між користувачем та інформаційною системою.

## **2.5 Розрахунок якості інтерфейсу користувача:**

Однією з обов'язкових деталей виконання проекту є розрахунок якості інтерфейсу користувача. Перш ніж проводити розрахунки, необхідно визначити що ж це таке.

Отже, інтерфейсом користувача називають сукупність певних засобів, за допомогою яких користувач спілкується з різними пристроями (комп'ютером, побутовою технікою) або іншим складним інструментарієм, тобто це такий різновид інтерфейсів, в якому з одного боку знаходиться людина, а з іншого – машина, а саме певний пристрій чи програмне забезпечення.

Ведучи розмову про розрахунок якості інтерфейсу користувача, ми повинні визначити якими саме є ті критерії, щоб провести розрахунки.

					<b>ІП-220424</b>	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		



Таким чином, існує чотири основні критерії якості будь-якого інтерфейсу, а саме: швидкість роботи користувачів, кількість людських помилок, швидкість навчання і суб'єктивне задоволення користувача, під яким розуміють відповідність інтерфейсу завданням користувача, що є невід'ємною його властивістю.

Почнемо зі швидкості виконання роботи. Це надзвичайно важливий критерій якості інтерфейсу, який в чистому вигляді цінують відносно рідко, але завжди він є вкрай бажаною складовою цілого. Тривалість виконання роботи користувачем складається з таких частин як: сприйняття початкової інформації, тривалостей інтелектуальної роботи, фізичних дій користувача та реакції системи. Детальніший розрахунок наведений в пункті під номером 9 (Розрахунок швидкості заповнення діалогових форм GOMS).

Кількість людських помилок. Так як людина не мислить як комп'ютер, необхідно враховувати той факт, що людина здатна на помилку. Тому, зі сторони розробника, ми повинні забезпечити все зі сторони інтерфейсу, щоб знизити кількість помилок вже зі сторони людини.

Так як ми працюємо з консольною програмою, нам необхідно, щоб інтерфейс був доступний для розуміння кожному користувачеві даної інформаційної системи. Від цього залежить як і швидкість навчання, так і суб'єктивне задоволення користувача.

Для приємнішого сприйняття людиною розробленого інтерфейсу, я використала функцію `system("color f4 && cls")` для зміни в консолі кольору фону на білий та шрифту на червоний. Також за допомогою символів «-» та «\_» створила умовні рамки, які вдосконалюють зовнішній вигляд даної інформаційної системи.

Головною складовою розуміння інтерфейсу є мова. Так як у Visual Studio 2022 за замовчуванням не вказана українська мова, без підключення конкретної бібліотеки ми бачимо всього лише набір певних символів.

					ІІІ-220424	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

Тому для того, щоб користувач розумів як працювати з системою ми перекодували за допомогою функцій SetConsoleCP(1251) та SetConsoleOutputCP(1251).

Також, не менш важливо складовою є структурований вигляд меню та програми в цілому. Для такої структуризації в даній курсовій роботі використано функцію endl, яка виводить текст, який буде наступним на новий рядок.

					<b>ІІІ-220424</b>	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

## 2.6 Розрахунок швидкості заповнення діалогових форм GOMS:

GOMS – «The model of Goals, Objects, Methods, and Selection rules» (в перекладі з англійської мови: «Модель Цілей, Об'єктів, Методів та Правил вибору»).

GOMS – група методів, що дає змогу модулювати виконання певної задачі користувачем та на основі цієї моделі визначити час виконання цієї задачі. Дуже часто ці методи використовують саме тоді, коли необхідно обрати один з двох варіантів розробки, у випадках коли навіть невеликі відмінності в швидкості можуть давати великий економічний та психологічний ефект.

Звісно, що потрібно враховувати людський фактор і те, що для кожної людини час виконання певної дії може відрізнятися, але, за допомогою лабораторних досліджень, вченими був отриманий набір часових інтервалів, потрібних для виконання певних жестів.

Для прикладу можемо розрахувати швидкість вводу даних з файлу користувачем.

Отже, загальний час дорівнюватиме (час запуску програми враховувати не будемо):

(Час:) піднесення руки на клавіші + міркування + вибір необхідної клавіші + міркування над наступним запитом + вибір необхідної клавіші.

Орієнтовно, спираючись на дані з досліджень, час цих дій дорівнюватиме:

- Піднесення руки на клавіші – 0,4 секунди;
- Міркування – 1,35 секунди;
- Вибір необхідної клавіші – 0,2 секунди;

Таким чином, приблизний результат матиме такий вигляд:

					ІІІ-220424	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

$$0,4 + 1,35 + 0,2 + 1,35 + 0,2 = 3,5 \text{ секунди.}$$

Таким чином можна порахувати швидкість будь-якої дії, яка виконується в інформацій системі.

## 2.7 Моделювання та тестування роботи інформаційної системи:

У більшості випадках термін «модулювання» трактують як процес створення точного опису системи або ж як метод пізнання, який в свою чергу базується на створенні і дослідженні моделей.

Ми використовуємо моделювання для полегшення вивчення об'єкта і, відповідно, його створення, перетворення та розвитку.

Для проектування інформаційних систем найчастіше використовують саме інформаційне моделювання, яке представляє собою використання певних інформаційних моделей, які відображають об'єкти та процеси у формі рисунків, схем, креслень, тощо.

Для кращого опису об'єктів нашої системи та їхнього вивчення, використовують таке поняття як ієрархія класів – класифікація об'єктних типів, розглядаючи об'єкти як реалізацію класів і пов'язуючи їх відношеннями по типу «наслідує», «розширює», «є його абстракцією» та ін.

Тепер розглянемо детальніше на прикладі даної курсової роботи. Для програмної реалізації поставленої задачі, в описі прототипів класів, для зручності передання інформації, використано таке поняття як структура – це власний користувацький тип даних, що дозволяє згрупувати змінні різних типів в одне ціле.

Тестування роботи інформаційної системи – це найнадійніший спосіб забезпечення якості розробки програмного забезпечення, який входить також в набір ефективних засобів забезпечення якості програмного продукту сучасності.

					ІП-220424	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

Тестування полягає у виконанні програми з метою перевірки інформаційної системи на відповідність програмного продукту на виході відносно нормативних, бізнес, технічних та функціональних вимог та вимог користувачів; виявленні технічних помилок з подальшим їх усуненням; оцінці зручності, продуктивності, сумісності та встановлення системи.

У випадку моєї інформаційної системи, методи та функції я створював і, відповідно, реалізовував поступово, щоб уникнути громіздких помилок та проблем. Відповідно, для тестування обраного мною методу, я створив лише один клас та декілька структур. Тільки після того, як я впевнився в коректній роботі обраного мною методу створення консольної інформаційної системи, я продовжив вдосконалювати програмний код відповідно до технічного завдання.

Також, я провів тестування після завершення роботи над кодом, щоб впевнитись, що все працює вірно. За допомогою цього тестування я виявив помилки у роботі програми, виправив їх та розширив функціонал і інтерфейс програми.

					<b>ІІ-220424</b>	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

# ВИСНОВКИ

У цій курсовій роботі було зроблено задачу розробки проекту консольної бази даних. У ній передбачалася робота з функціями та файлами, виділенням пам'яті, алгоритмом пошуку унікальних значень і т. п. Програму було написано у середовищі Visual Studio – C++.

Виконання курсової роботи була із метою освоєння усіх теоретичних знань, отриманих з дисципліни «Об'єктно-орієнтоване програмування» і

Перевірка роботи програмі і її тестування показали, що завдання реалізовано коректно. Розроблена система використання баз даних і їх управління, завдяки чому, можна повноцінно реалізовувати бази даних і можливі маніпуляції з ними, завдяки основних розроблених функцій, які надають можливість роботи з базами даних і навіть можливість використання власних методів.

Отже, можна зробити висновок, що C++ - це мова програмування високого рівня, яка попри виконання інших завдань, підходить і для використання бази даних

Під час написання курсової роботи були освоєні основні бібліотеки мови C++ і їх засоби реалізації, а також були поглиблені знання про роботу з консоллю і базами. Вивчивши та проаналізувавши мову C++, зроблено висновок, що ця мова є досить гнучким і доступним інструментом для роботи з базами даних і також використання у проєктах, що реалізовані на різних мовах програмування.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Інтернет-джерело «Cplusplus.com».
2. Інтернет-джерело «Wikipedia.org».
3. Гарасимів Т. Г. Структурне програмування: лабораторний практикум / Т. Г. Гарасимів, В. Б. Кропивницька, В. М. Гарасимів - Івано-Франківськ: ІФНТУНГ, 2016. – 117 с.
4. Бази даних: проектування та реалізація/ Г. С. Погромська, Н.А. Махровська. – Місто: Видавництво, 2019. – 183 с.
5. Інтернет-джерело «Cyberforim.ru».

					<b>ІІІ-220424</b>	Арк.
						23
Змн.	Арк.	№ докум.	Підпис	Дата		

# ДОДАТКИ

					ІІІ-220424	Арк.
						24
Змн.	Арк.	№ докум.	Підпис	Дата		



## Додаток А:

### Технічне завдання

Міністерство освіти і науки України

Івано-Франківський національний технічний університет нафти і газу

Інститут інформаційних технологій

ЗАТВЕРДЖУЮ

зав. кафедри ІПЗ, проф., д. т. н.

\_\_\_\_\_ В. І. Шекета

« \_\_\_\_\_ » \_\_\_\_\_ 2023 р.

### ТЕХНІЧНЕ ЗАВДАННЯ

На розробку прикладного програмного забезпечення з моделювання та аналізу програмного забезпечення»

1. Область застосування — проектування інформаційних систем.

2. Основа розробки — робочий навчальний план дисципліни.

3. Мета та експлуатаційне призначення:

а. мета – отримання практичних навичок проектування та конфігурування інформаційних систем ;

б. призначення розробки — навчальний курсовий проект із дисципліни «Проектування інформаційних систем»;

4. Джерела розробки — індивідуальне завдання на курсовий проект із дисципліни, технічні рекомендації щодо проектування інформаційних систем та інші технічні матеріали для налаштування окремих компонентів програмної системи.

5. Технічні вимоги

Кінцевий термін виконання курсового проекту « \_\_\_\_\_ » 2023 р

Початок розробки «20 квітня» 2023 р

Порядок контролю та прийняття.

6.1. Виконання етапів технічної та розрахункової документації курсового проекту, а також моделювання роботи інформаційної системи контролюється викладачем згідно з графіком виконання проекту;

6.2. Прийняття проекту здійснюється комісією, затвердженою зав. кафедри згідно графіку захисту.

6.3. Коригування технічного завдання допускається з дозволу керівника проекту.

Розробив студент групи ІП-22-1 Токарев Сергій

					ІП-220424	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

## Додаток Б:

Фото програми:

MENU	
-----	
АІС МІСЬКА ФІЛАРМОНІЯ	
/1/ Культурні споруди	
/2/ Список артистів по жанрах	
/3/ Список артистів та імпресаріо	
/4/ Артисти в деяких жанрах	
/5/ Імпресаріо артиста	
/6/ Концертні заходи	
/7/ Список призерів конкурсів	
/8/ Заходи в культурних спорудах	
/9/ Список імпресаріо за жанром	
/10/ Артисти-конкурсанти	
/11/ Список організаторів заходів	
/12/ Культурні споруди	
/13/ Exit	
-----	
-----Виконав-----	
-----Студент групи ІП-22-1-----	
----- ІП - 220424-----	
- Токарєв Сергій Олександрович -	
-----	
Ваш вибір:	

-----Культурні споруди-----

Назва: 'Зірка'

Тип: Кінотеатр

Назва: 'IMAX'

Тип: Кінотеатр

Назва: 'Ренесанс'

Тип: Театр

Назва: 'Бароко'

Тип: Театр

Назва: 'PowerUP'

Тип: Концертний майданчик

Назва: 'RapPlace'

Тип: Концертний майданчик

Для продовження натисніть будь-яку клавішу . . .

-----Артисти та їх жанри-----

Ім'я: Олексій\_Потапенко

Жанр: Реп, Хіп-хоп, Джаз

Ім'я: Леонід\_Кучма

Жанр: Драма

Ім'я: Константин\_Мудрий

Жанр: Драма, Опера

Ім'я: Олександр\_Великий

Жанр: Опера

Ім'я: Анатолій\_Смішко

Жанр: Стенд-ап

Ім'я: Ігнатій\_Тишко

Жанр: Драма

Для продовження натисніть будь-яку клавішу . . .

-----

----Артисти та їх імпресаріо----

-----

-----

Ім'я: Олексій\_Потапенко

Імпресаріо: Дмитро Шеремета

-----

-----

Ім'я: Леонід\_Кучма

Імпресаріо: Дмитро Шеремета, Василь Коломиєць

-----

-----

Ім'я: Константин\_Мудрий

Імпресаріо: Василь Коломиєць, Олена Кушнір

-----

-----

Ім'я: Олександр\_Великий

Імпресаріо: Анна Романів

-----

-----

Ім'я: Анатолій\_Смішко

Імпресаріо: Володимир Зелемський

-----

-----

Ім'я: Ігнатій\_Тишко

Імпресаріо: Олена Кушнір

-----

-----

Для продовження натисніть будь-яку клавішу . . .

-----

-----

--Артисти з декількома жанрами--

-----

-----

Ім'я: Олексій\_Потапенко

Жанр: Реп, Хіп-хоп, Джаз

-----

-----

Ім'я: Константин\_Мудрий

Жанр: Драма, Опера

-----

-----

Для продовження натисніть будь-яку клавішу . . .

-----

-----Імпресаріо артиста-----	
Введіть артиста: Олексій_Потапенко	
-----	
Ім'я: Олексій_Потапенко	
Імпресаріо: Дмитро Шеремета	
Для продовження натисніть будь-яку клавішу . . .	

-----Імпресаріо артиста-----	
Введіть артиста: Константин_Мудрий	
-----	
Ім'я: Константин_Мудрий	
Імпресаріо: Василь Коломиєць, Олена Кушнір	
Для продовження натисніть будь-яку клавішу . . .	

-----Концертні заходи-----				
Назва концерту: RockFest Дата: 28.07 Організатор: Дмитро Шеремета				
Назва концерту: DZIDZIO Дата: 25.06 Організатор: Василь Коломиєць				
Назва концерту: TapTak Дата: 1.07 Організатор: Василь Коломиєць				
Назва концерту: Антитіла Дата: 12.07 Організатор: Олена Кушнір				
Назва концерту: HardKiss Дата: 23.07 Організатор: Володимир Зелемський				
Назва концерту: Тінь Сонця Дата: 2.07 Організатор: Анна Романів				
-----Призери конкурсів-----				
Назва: Реп-батл Переможець: Олексій Потапенко				
Назва: Найкращий драматург Переможець: Леонід Кучма				
Назва: Розсміши Коміка Переможець: Володимир Зелемський				
Назва: Найкращий вокал Переможець: Олександр Великий				
Для продовження натисніть будь-яку клавішу . . .				

-----Імпресаріо та їх жанри-----

Імпресаріо: Дмитро Шеремета

Жанр: Реп, Хіп-хоп, Джаз

Імпресаріо: Дмитро Шеремета, Василь Коломиєць

Жанр: Драма

Імпресаріо: Василь Коломиєць, Олена Кушнір

Жанр: Драма, Опера

Імпресаріо: Анна Романів

Жанр: Опера

Імпресаріо: Володимир Зелемський

Жанр: Стенд-ап

Імпресаріо: Олена Кушнір

Жанр: Драма

Для продовження натисніть будь-яку клавішу . . . ■

---Участь артистів в конкурсах---

Ім'я: Олексій\_Потапенко

Брав(ла) участь в конкурсах: так

Ім'я: Леонід\_Кучма

Брав(ла) участь в конкурсах: ні

Ім'я: Константин\_Мудрий

Брав(ла) участь в конкурсах: ні

Ім'я: Олександр\_Великий

Брав(ла) участь в конкурсах: так

Ім'я: Анатолій\_Смішко

Брав(ла) участь в конкурсах: так

Ім'я: Ігнатій\_Тишко

Брав(ла) участь в конкурсах: так

Для продовження натисніть будь-яку клавішу . . .

-----Організатори культурних заходів-----

Ім'я: Іван Хрущ

Кількість проведених концертів: 32

Ім'я: Дмитро Паровозов

Кількість проведених концертів: 88

Ім'я: Олександр Авраменко

Кількість проведених концертів: 12

Ім'я: Лев Волощук

Кількість проведених концертів: 114

Ім'я: Григорій Пачка

Кількість проведених концертів: 223

Для продолжения нажмите любую клавишу . . .

Змн.	Арк.	№ докум.	Підпис	Дата

**ІІІ-220424**

Арк.

32



---Культурні споруди та заходи---

Назва: 'Зірка'  
Тип: концертний зал

Назва концерту: RockFest  
Дата: 28.07  
Організатор: Дмитро Шеремета

Назва: 'IMAX'  
Тип: концертний зал

Назва концерту: DZIDZIO  
Дата: 25.06  
Організатор: Василь Коломиєць

Назва: 'Ренесанс'  
Тип: концертний зал

Назва концерту: TapTak  
Дата: 1.07  
Організатор: Василь Коломиєць

Назва: 'Бароко'  
Тип: концертний зал

Назва концерту: Антитіла  
Дата: 12.07  
Організатор: Олена Кушнір

Назва: 'PowerUP'  
Тип: концертний зал

Назва концерту: HardKiss  
Дата: 23.07  
Організатор: Володимир Зелемський

Назва: 'RapPlace'  
Тип: концертний зал

Назва концерту: Тінь Сонця  
Дата: 2.07  
Організатор: Анна Романів

## Додаток В:

Код програми мовою C++:

Файл main.cpp

```
#include <iostream>
```

```
#include "Header1.h"
```

```
int requestNum = -1;
```

```
void mainMenu()
```

```
{
```

```
    system("cls");
```

```
    cout << "                MENU                " << endl;
```

```
    cout << "|-----|" << endl;
```

```
    cout << "\\t    АІС МІСЬКА ФІЛАРМОНІЯ        |" << endl;
```

```
    cout << "\\t/1/ Культурні споруди                |" << endl;
```

```
    cout << "\\t/2/ Список артистів по жанрах            |" << endl;
```

```
    cout << "\\t/3/ Список артистів та імпресаріо        |" << endl;
```

```
    cout << "\\t/4/ Артисти в деяких жанрах                |" << endl;
```

```
    cout << "\\t/5/ Імпресаріо артиста                    |" << endl;
```

```
    cout << "\\t/6/ Концертні заходи                      |" << endl;
```

```
    cout << "\\t/7/ Список призерів конкурсів                |" << endl;
```

```
    cout << "\\t/8/ Заходи в культурних спорудах            |" << endl;
```

```
    cout << "\\t/9/ Список імпресаріо за жанром              |" << endl;
```

```
    cout << "\\t/10/ Артисти-конкурсанти                   |" << endl;
```

```
    cout << "\\t/11/ Список організаторів заходів          |" << endl;
```

```
    cout << "\\t/12/ Культурні споруди                    |" << endl;
```

```

cout << "\\t/13/ Exit" << endl;

cout <<
"|_____|" <<
endl;

```

```

cout << "\\t |-----Виконав-----|" << endl;
cout << "\\t |-----Студент групи ІП-22-1-----|" << endl;
cout << "\\t |----- ІП - 220424-----|" << endl;
cout << "\\t |- Токарев Сергій Олександрович -|" << endl;
cout << "\\t |_____|" << endl;

```

```

cout << "\\n\\n\\nВаш вибір: " << endl;

```

```

cin >> requestNum;

```

```

}

```

```

int main()
{
    SetConsoleOutputCP(1251);
    SetConsoleCP(1251);

```

```

vector<Building> buildings;

buildings.push_back(Building("Зірка", "Кінотеатр"));
buildings.push_back(Building("IMAX", "Кінотеатр"));
buildings.push_back(Building("Ренесанс", "Театр"));
buildings.push_back(Building("Бароко", "Театр"));
buildings.push_back(Building("PowerUP", "Концертний
майданчик"));

buildings.push_back(Building("RapPlace", "Концертний
майданчик"));


vector<Artist> artists;

artists.push_back(Artist("Олексій_Потапенко", "Дмитро Шеремета",
3, "Реп, Хіп-хоп, Джаз", 1));

artists.push_back(Artist("Леонід_Кучма", "Дмитро Шеремета,
Василь Коломиєць", 1, "Драма", 0));

artists.push_back(Artist("Констянтин_Мудрий", "Василь
Коломиєць, Олена Кушнір", 2, "Драма, Опера", 0));

artists.push_back(Artist("Олександр_Великий", "Анна Романів", 1,
"Опера", 1));

artists.push_back(Artist("Анатолій_Смішко", "Володимир
Зелемський", 1, "Стенд-ап", 1));

artists.push_back(Artist("Ігнатій_Тишко", "Олена Кушнір", 1,
"Драма", 1));


vector<Concert> concerts;

concerts.push_back(Concert("RockFest", 28.07, "Дмитро Шеремета",
"Арена-1"));

```

					ІІІ-220424	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		36

```

concerts.push_back(Concert("DZIDZIO", 25.06, "Василь
Коломиєць", "Арена-2"));

concerts.push_back(Concert("ТарТак", 01.07, "Василь Коломиєць",
"Арена-2"));

concerts.push_back(Concert("Антитіла", 12.07, "Олена Кушнір",
"Арена-1"));

concerts.push_back(Concert("HardKiss", 23.07, "Володимир
Зелемський", "Арена-2"));

concerts.push_back(Concert("Тінь Сонця", 02.07, "Анна Романів",
"Арена-1"));

```

```

vector<Contest> contests;

contests.push_back(Contest("Реп-батл", "Олексій Потапенко"));

contests.push_back(Contest("Найкращий драматург", "Леонід
Кучма"));

contests.push_back(Contest("Розсміши Коміка", "Володимир
Зелемський"));

contests.push_back(Contest("Найкращий вокал", "Олександр
Великий"));

```

```

vector<Organizer> organizers;

organizers.push_back(Organizer("Іван Хрущ", 32));

organizers.push_back(Organizer("Дмитро Паровозов", 88));

organizers.push_back(Organizer("Олександр Авраменко", 12));

organizers.push_back(Organizer("Лев Волощук", 114));

organizers.push_back(Organizer("Григорій Пачка", 223));

```

```

while (requestNum != 13)
{
    mainMenu();

    if (requestNum == 1)
    {
        system("cls");
        cout << "|-----|" << endl;
        cout << "|-----Культурні споруди-----|" << endl;
        cout << "|-----|\n\n" << endl;
        for (Building b : buildings)
        {
            b.PrintType();
        }
        system("pause");
    }

    else if (requestNum == 2)
    {
        system("cls");
        cout << "|-----|" << endl;
        cout << "|-----Артисти та їх жанри-----|" << endl;
        cout << "|-----|\n\n" << endl;
        for (Artist a : artists)
        {

```

```

        a.PrintGenre();
    }
    system("pause");
}

else if (requestNum == 3)
{
    system("cls");
    cout << "|-----|" << endl;
    cout << "|----Артисти та їх імпресаріо----|" << endl;
    cout << "|-----|\n\n" << endl;
    for (Artist a : artists)
    {
        a.PrintImpresario();
    }
    system("pause");
}

else if (requestNum == 4)
{
    system("cls");
    cout << "|-----|" << endl;
    cout << "|--Артисти з декількома жанрами--|" << endl;
    cout << "|-----|\n\n" << endl;
    for (Artist a : artists)
    {

```

```

        if (a.getGenreNum() > 1)
        {
            a.PrintGenre();

        }
    }
    system("pause");
}

else if (requestNum == 5)
{
    system("cls");
    cout << "|-----|" << endl;
    cout << "|-----Імпресаріо артиста-----|" << endl;
    cout << "|-----|\n\n" << endl;
    string name = "";
    cout << "Введіть артиста: ";
    cin >> name;

    for (Artist a : artists)
    {
        if (strcmp(a.getName().c_str(), name.c_str()) == 0)
        {
            a.PrintImpresario();
        }
    }
}

```



```

        system("pause");
    }

    else if (requestNum == 6)
    {
        system("cls");
        cout << "|-----|" << endl;

        cout << "|-----Концертні заходи-----|" << endl;
        cout << "|-----|\n\n" << endl;
        for (Concert c : concerts)
        {
            c.PrintInfo();
        }
        system("pause");
    }

    else if (requestNum == 7)
    {
        system("cls");
        cout << "|-----|" << endl;
        cout << "|-----Призери конкурсів-----|" << endl;
        cout << "|-----|\n\n" << endl;
        for (Contest c : contests)
        {
            c.PrintInfo();
        }
    }

```

```

    }

    system("pause");
}

else if (requestNum == 8)
{
    system("cls");
    cout << "|-----|" << endl;
    cout << "|-----Концерти в спорудах-----|" << endl;
    cout << "|-----|\n\n" << endl;
    string building = "";

    cout << "Введіть назву споруди: ";
    cin >> building;

    for (Concert c : concerts)
    {
        if (strcmp(c.getBuilding().c_str(), building.c_str()) == 0)
        {
            c.PrintInfo();
        }
    }

    system("pause");
}

```

```

else if (requestNum == 9)
{
    system("cls");
    cout << "|-----|" << endl;
    cout << "|-----Імпресаріо та їх жанри-----|" << endl;
    cout << "|-----|\n\n" << endl;
    for (Artist a : artists)
    {
        a.PrintImpresarioGenre();
    }
    system("pause");
}

```

```

else if (requestNum == 10)
{

    system("cls");
    cout << "|-----|" << endl;
    cout << "|---Участь артистів в конкурсах--|" << endl;
    cout << "|-----|\n\n" << endl;
    for (Artist a : artists)
    {
        a.PrintContest();
    }
    system("pause");
}

```

```

}

else if (requestNum == 11)
{
    system("cls");
    cout << "|-----|" << endl;
    cout << "|-----Організатори культурних заходів----|" << endl;
    cout << "|-----|\n\n" << endl;
    for (Organizer o : organizers)
    {
        o.PrintInfo();
    }
    system("pause");
}

else if (requestNum == 12)
{
    system("cls");
    cout << "|-----|" << endl;
    cout << "|---Культурні споруди та заходи--|" << endl;

    cout << "|-----|\n\n" << endl;
    for (int i = 0; i < 6; i++)
    {
        buildings[i].PrintConcert();
        concerts[i].PrintInfo();
    }
}

```

```

    }
    system("pause");
}

else if (requestNum == 13)
{
    system("cls");
    cout << "|-----|" << endl;
    cout << "|-----Завершення роботи-----|" << endl;
    cout << "|-----|" << endl;
    cout << "|-----Натисніть будь-яку клавішу-----|" << endl;
    cout << "|-----|" << endl;
    system("pause");
}
}

return 0;
}

```

## Файл Header1.h

```
#pragma once
```

```
#include <iostream>
```

```
#include <string>
```

```
#include <vector>
```

```
#include <Windows.h>
```

```
using namespace std;
```

```
class Building {
```

```
    private: string name;
```

```
    string type;
```

```
    public: Building(string name, string type) {
```

```
        this -> name = name;
```

```
        this -> type = type;
```

```
    }
```

```
    void PrintType() {
```

```
        cout << "|-----" << endl;
```

```
        cout << "| Назва: " << this -> name << endl;
```

```
        cout << "| Тип: " << this -> type << endl;
```

```
    }
```

```
    void PrintConcert() {
```

```
        cout << "|-----" << endl;
```

```
        cout << "| Назва: " << this -> name << endl;
```

```
        cout << "| Тип: концертний зал" << endl;
```

					ІІІ-220424	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    }
};

class Artist {
    private: string name;
    string impresario;
    int genreNum;
    string genres;
    bool hasBeenInContest;

    public: Artist(string name, string impresario, int genreNum, string genres, bool
hasBeenInContest) {
        this -> name = name;
        this -> impresario = impresario;
        this -> genreNum = genreNum;
        this -> genres = genres;
        this -> hasBeenInContest = hasBeenInContest;
    }

    void PrintGenre() {

        cout << "|-----" << endl;
        cout << "| Имя: " << this -> name << endl;
        cout << "| Жанр: " << this -> genres << endl;
    }

    void PrintImpresario() {
        cout << "|-----" << endl;
        cout << "| Имя: " << this -> name << endl;
        cout << "| Импресарио: " << this -> impresario << endl;
    }
}

```

					III-220424	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

```

void PrintImpresarioGenre() {
    cout << "|-----" << endl;
    cout << "| Імпреcapio: " << this -> impresario << endl;
    cout << "| Жанр: " << this -> genres << endl;
}

```

```

void PrintContest() {
    cout << "|-----" << endl;
    cout << "| Ім'я: " << this -> name << endl;
    cout << "| Брав(ла) участь в конкурсах: ";
    if (this -> hasBeenInContest == 1) {
        cout << "так" << endl;
    } else {

        cout << "ні" << endl;
    }
}

```

```

int getGenreNum() {
    return genreNum;
}

```

```

string getName() {
    return name;
}

```

```
};
```

```

class Concert {
private:

```

					III-220424	Арк.
						48
Змн.	Арк.	№ докум.	Підпис	Дата		



```

    string name;

    double time;

    string organiser;

    string building;


    public: Concert(string name, double time, string organiser, string building)

    {

        this -> name = name;

        this -> time = time;

        this -> organiser = organiser;

        this -> building = building;

    }


    void PrintInfo() {

        cout << "|-----" << endl;

        cout << "| Назва концерту: " << this -> name << endl;

        cout << "| Дата: " << this -> time << endl;

        cout << "| Організатор: " << this -> organiser << endl;

    }


    string getBuilding() {

        return building;

    }

};


class Contest {

    private: string name;

    string winner;


    public:

```

```

Contest(string name, string winner) {
    this -> name = name;
    this -> winner = winner;
}

void PrintInfo() {
    cout << "|-----" << endl;
    cout << "| Назва: " << this -> name << endl;
    cout << "| Переможець: " << this -> winner << endl;
}

};

class Organizer {
private: string name;
int concertNum;
public: Organizer(string name, int concertNum) {
    this -> name = name;
    this -> concertNum = concertNum;
}

void PrintInfo() {

    cout << "|-----" << endl;
    cout << "| Ім'я: " << this -> name << endl;
    cout << "| Кількість проведених концертів: " << this -> concertNum << endl;
}

};

```

					<div>III-220424</div>	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		