**Варіант 1**

* Що називають патерном проектування

Выдповыдь;

Стандартизація коду, перевірені рішення, загальний словник програмістів.

* Що з перерахованого є правильним щодо шаблону проектування Abstract Factory?
* Шаблон відноситься до групи породжуючих шаблонів.
* Abstract Factory створює сімейства об’єктів
* Інтерфейс Abstract Factory відповідає за створення сімейства пов'язаних об'єктів без явної вказівки класів об’єктів.
* Підходить все перераховане вище.
* Що з перерахованого нижче описує шаблон Decorator правильно?
* Ця модель дозволяє користувачеві додавати нові функціональні можливості існуючого об'єкта без зміни його структури.
* Ця модель використовується там, де ми повинні ставитися до групи об'єктів подібним чином як до єдиного об'єкту.
* Ця модель приховує складність системи і забезпечує інтерфейс, за допомогою якого клієнт може отримати доступ до системи.
* Ця модель в основному використовується для скорочення кількості об'єктів, що створюються, зменшення обсягу використовуваної пам'яті і збільшення продуктивності.
* Що з наведеного нижче описує шаблон Прототип правильно?
* Цей шаблон будує складний об'єкт, використовуючи прості об'єкти і використовуючи покроковий підхід.
* Виконується створення дублікату об'єкта, маючи на увазі продуктивність.
* Цей патерн працює як міст між двома несумісними інтерфейсами.
* Застосовується тоді, коли нам потрібно відокремити абстракцію від її реалізації, щоб вони могли змінюватись незалежно.
* Який з наступних патернів використовується там, де ми повинні поводитися з групою об'єктів як з одним об'єктом?
* Composite
* Facade
* Flyweight
* Decorator
* Які з перерахованих нижче пунктів правильно описує шаблон Builder?
* Цей шаблон будує складний об'єкт, використовуючи прості об'єкти із використанням поетапного підходу до побудови.
* Шаблон відповідає за створення об'єкта- дубліката при збереженні продуктивності
* Цей шаблон використовується, якщо створення об'єкта безпосередньо є дорогим.
* Який з наступних шаблонів працює як міст між двома несумісними інтерфейсами?
* Builder
* Facade
* Adapter
* Prototype
* Який шаблон слід використовувати, щоб відокремити абстракцію від її реалізації, так що обидві можуть змінюватися незалежно?
* Adapter
* Bridge
* Facade
* Composite
* Який шаблон визначає залежність між класами один до багатьох таким чином, що коли один об'єкт змінив стан, всі залежні повідомляються та оновлюються автоматично?
* Composite
* Strategy
* Observer
* Facade
* Який шаблон використати для керуванням віддаленим об’єктом?
* Command
* Bridge
* Strategy
* Proxy
* Adapter
* Що з наведеного описує патерн Фасад правильно?
* Цей шаблон використовується, якщо створення об'єкта безпосередньо є дорогим.
* Застосовується там, коли нам потрібно розглядати групу об'єктів як єдиний об'єкт.
* Приховує складність системи та забезпечує інтерфейс для клієнта, за допомогою якого клієнт може отримати доступ до системи.
* Використовується в першу чергу для зменшення кількості створених об'єктів та підвищення продуктивності.
* Що з наведеного нижче є правильним щодо моделі дизайну Singleton.
* Цей патерн відноситься до породжуючих
* Визначає один клас, який відповідає за створення об'єкта, причому створюється лише один об'єкт.
* Клас Singleton забезпечує спосіб доступу до свого єдиного об'єкта без необхідності безпосереднього створення об'єкту класу
* Все вищеперераховане.

**Варіант 2**

* Назвіть групи патернів проектування
* Який з наступних патернів включає один клас, що відповідає за створення об'єкту, при цьому лише один об'єкт буде створений?
* Factory
* Abstract Factory
* Singleton
* Monostate
* Builder
* Що з перерахованого є правильним щодо шаблону проектування Factory Method?
* Шаблон відноситься до групи породжуючих шаблонів.
* Шаблон створює взаємозамінні сімейства об’єктів
* Відповідає за створення об’єкта без явної вказівки класу об’єкта.
* Підходить все перераховане вище.
* Який з наступних патернів використовується, щоб отримати доступ до елементів об'єкта колекції в послідовному порядку, при цьому не потрібно знати про спосіб зберігання колекції?
* Composite
* Iterator
* Flyweight
* State
* Які з перерахованих нижче пунктів правильно описує шаблон Bridge?
* Цей шаблон будує складний об'єкт, використовуючи прості об'єкти і з використанням поетапного підходу.
* Шаблон відповідає за створення об'єкта- дубліката при збереженні продуктивності
* Цей шаблон використовується, коли нам потрібно відокремити абстракції від його реалізації, так що обидві можуть змінюватися незалежно
* Цей шаблон використовується, якщо створення об'єкта безпосередньо є дорогим.
* Які з перерахованих нижче пунктів правильно описує шаблон Template method?
* В цьому шаблоні, поведінка класу змінюється на основі його стану.
* Поведінка класу або його алгоритм може бути змінений під час виконання.
* Абстрактний клас надає певним чином (и) / шаблон (и) для виконання своїх методів.
* У якому з наступних шаблонів клас представляє функціональність іншого класу?
* Strategy
* Command
* Adapter
* Iterator
* Proxy
* У якому випадку варто використовувати шаблон проектування Адаптер?
* Для приведення існуючого інтерфейсу класу в сумісний цільовий інтерфейс
* Щоб поліпшити продуктивність системи
* Для перетворення даних клієнтського коду в відповідний формат, очікуваний цільовим інтерфейсом
* Щоб класи з несумісними інтерфейсами могли працювати разом
* Що з наведеного описує патерн Компонувальник правильно?
* Цей патерн будує складний об'єкт, використовуючи прості об'єкти і використовуючи покроковий підхід.
* Застосовується там, коли нам потрібно розглядати групу об'єктів як єдиний об'єкт.
* Приховує складність системи та забезпечує інтерфейс для клієнта, за допомогою якого клієнт може отримати доступ до системи.
* Використовується в першу чергу для зменшення кількості створених об'єктів і зменшення сліду пам’яті та підвищення продуктивності.
* Proxy- шаблон найкраще використовувати для:
* Контролю за доступом до віддаленого об'єкту
* Для збірки складних об'єктів
* Створення ресурсномісткого об'єкт при запиті клієнта
* Зберігання загальних надлишкових даних між великою кількістю об'єктів
* Шаблон забезпечує засоби визначення сімейства взаємозамінних алгоритмів і інкапсулює кожен з них в якості об'єкта:
* Abstract Factory
* State
* Strategy
* Chain of Responsibility
* Template Method

**Практика**

* Програма. Використати шаблон Абстрактна фабрика для створення двох фабрик по пошиттю спортивного та щоденного одягу.
* На кожній фабриці повинно вироблятися не менше як два продукти.
* "Secure Door". Створити програму з використанням патерну Замісник.
* \*Створення різних видів зброї для деякої гри. У програмі визначити масив з об’єктів зброї(прототипів). Застосувати патерн Прототип.
* Створити класи, що відповідають представленню точки у декартовій та полярній системах координат(DecPoint, PolarPoint). Визначити у класі DecPoint метод переміщення точки Move(dx, dy). Визначити у класі PolarPoint метод зміни кута точки Rotate(angle). Створити клас Point, який надає можливість використовувати обидва представлення точки. Використати патерн Фасад.
* Створити клас Time24(hour, min), що зберігаї час у 24 годинному форматі. Визначити у класі методи збільшення часу на певну кількість хвилин, годин. Адаптувати клас Time до інтерфейсу Time12.
* Створити класи, що відповідають окремим поштовим сервісам відправки листа та відправки посилки. Створити клас Пошта, який надає можливість використовувати обидва вказані сервіси. Використати патерн Фасад.
* Створити абстрактний клас Sort для впорядкування масиву цілих чисел за зростанням чи спаданням. Визначити у класі Sort чисту віртуальну функцію порівняння двох цілих. Створити два похідні класи(SortUp, SortDown), що реалізують абстрактний метод порівняння двох цілих чисел за зростанням та спаданням відповідно. Використати патерн проектування Шаблонний Метод(Template Method).
* Створити класи, що відповідають окремим сервісам автосалону: придбання авто, ремонт авто. Створити клас Автосалон, який надає можливість використовувати обидва вказані сервіси. Використати патерн Фасад
* Скласти програму. Створити класи Товар(назва, кількість), Магазин(масив товарів). Визначити клас Покупка(номер товару, кількість), що реалізує сервіс купівлі товару. Використати патерн проектування Proxy.
* Створити клас Масив. Адаптувати клас Масив до інтерфейсу Стек. Використати патерн проектування Адаптер
* Скласти програму. Визначити клас Квадрат. Визначити клас Коло. Використати патерн проектування Адаптер для можливості «перетворення» Квадрата у найближче за розміром Коло
* Скласти програму. Створити клас Студент(прізвище, ім’я, оцінка у 100 бальній шкалі). Створити клас-адаптер класу Студент, який надає інший спосіб виведення оцінки студента (у шкалі A-F).
* Скласти програму. Створити класи Замовлення квитків та Бронювання готелю. Використати патерн проектування Фасад для доступу до підсистем(функцій класів Замовлення квитків та Бронювання готелю).
* Скласти програму. Створити клас Журнал подій програми. Клас повинен надавати можливість запису подій у деякий текстовий файл. Забезпечити існування одного екземпляру журналу.
* Скласти програму. Конвертація двовимірного масиву у текстовий формат та у одновимірний масив(елементи одновимірного масиву - суми рядків матриці). Скористатися шаблоном Будівельник.
* Скласти програму. Створення блоків різної форми для гри «Тетріс». У програмі визначити "склад" з об’єктів блоків-прототипів. Застосувати патерн Прототип.
* Виведення кроків приготування піци у файл(основа, начинка, випікання, пакування) та обчислення вартості створеної піци. Builder.
* Створення різних видів реклами(відео, аудіо, текст). Factory method.
* Адаптувати клас deque<> до інтерфейсу Черга.
* Створити клас ShoppingCart(Корзина покупок) з можливістю
* додавати у корзину Товар
* видаляти Товар
* рахувати загальну вартість
* оплатити покупку
* Оплата покупки може здійснюватися двома способами
* Кредитною Карткою(імя, номер картки, cvvб дата дії картки)
* у системі PayPal(emailId, password)

Використати патерн Стратегія.

* Створити класи Працівник(Employee) та Відділ(Department). Відділ може містити працівників та інші відділи. Передбачити можливість
* доповнення відділу працівником та іншим відділом
* вилучення з відділу працівника чи під-відділу
* обчислення заробітної плати працівника чи відділу

Використати патерн Компонувальник.

* Патерн Шаблонний метод. Створити абстрактний клас Опитування, що містить методи проведення певних етапів опитування(початок, теоретична частина, практична частина, завершення). Створити класи Залік та Екзамен, які кожен по своєму реалізує певний етап.

Загальний алгоритм опитування для заліку та екзамену однаковий.

* Створити клас Fighter, який дозволяє герою перемикати стратегію руху в залежності від наявності(чи ні) перешкод різного виду(гірка, яма, озеро, ворог). Можливі стратегії руху Стрибати, Пливти, Стріляти чи ін.

\*Створити клас Текст та декілька класів -обгорток(колір, гарнітура шрифту, розмір).

--------------------------------------------------------------------------------------

Реализуйте с использованием паттернов проектирования простейшую систему планирования задач. Должна быть возможность создания списка дел, установки приоритетов, установки дат выполнения, удаление и изменения дел.

Каждому делу можно установить тег. Список дел можно загружать и сохранять в файл. Необходимо реализовать возможность поиска конкретного дела. Критерии поиска: по датам, по тегам, по приоритету и так далее.