

Ćwiczenia 8 i 9 - wzorce operacyjne

Interpreter

<http://devman.pl/pl/techniki/wzorce-projektowe-interpreterinterpreter/>

<https://www.dotnettricks.com/learn/designpatterns/interpreter-design-pattern-c-sharp>

<https://www.geeksforgeeks.org/interpreter-design-pattern/>

<https://www.c-sharpcorner.com/UploadFile/20c06b/learn-design-patterns-interpreter/>

J1. Stwórz projekt implementujący wzorec Interpretera do zamiany liczby z systemu szesnastkowego na dziesiętny.

Iterator (kursor)

<http://devman.pl/pl/techniki/wzorce/wzr-opr/wzorce-projektowe-iteratoriterator/>

<https://refactoring.guru/design-patterns/iterator/csharp/example>

<https://www.c-sharpcorner.com/article/iterator-design-pattern/>

J2. Zmodyfikuj kod ze wzorca Kompozytu z poprzednich ćwiczeń o strukturę Iteratora.

Łańcuch zobowiązań

<http://lukaszkosiorowski.pl/programowanie/lancuch-zobowiazan-chain-of-responsibility/>

<https://refactoring.guru/design-patterns/chain-of-responsibility/csharp/example>

<https://www.codeproject.com/Articles/743783/Reusable-Chain-of-responsibility-in-Csharp>

<https://www.dofactory.com/net/chain-of-responsibility-design-pattern>

J3. Stwórz projekt zamieniający liczby z systemu dziesiętnego (cyry arabskie) na system rzymski z wykorzystaniem Łańcucha Zobowiązań.

Mediator

<https://refactoring.guru/design-patterns/mediator/csharp/example>

<https://www.dofactory.com/net/mediator-design-pattern>

<http://devman.pl/pl/techniki/wzorce-projektowe-mediatormediator/>

J4. Stwórz dwie testowe klasy i mediatora między nimi.

Metoda szablonowa

<https://refactoring.guru/design-patterns/template-method/csharp/example>

<https://www.dotnettricks.com/learn/designpatterns/template-method-design-pattern-c-sharp>

J5. Stwórz projekt zgodnie ze wzorcem metody szablonowej do przechowywania liczb w różnych systemach pozycyjnych.

Obserwator

<https://www.plukasiewicz.net/Artykuly/Obserwator>

<http://devman.pl/pl/techniki/wzorce-projektowe-obszawatorobserver/>

<https://refactoring.guru/design-patterns/observer/csharp/example>

J6. Stwórz projekt do obserwowania cen produktów w serwisie aukcyjnym (za pomocą wzorca Obserwatora).

Odwiedzający (wizytator)

<http://devman.pl/pl/techniki/wzorcow/wzorcowe-projektowe-odwiedzajacyvisitor/>

<https://refactoring.guru/design-patterns/visitor/csharp/example>

<http://dotnetpattern.com/visitor-design-pattern>

J7. Wzorując się na przykładzie https://en.wikipedia.org/wiki/Visitor_pattern#C%23_example , stwórz kalkulator implementujący wzorzec Obserwatora.

Pamiętka (zaczni)

<http://devman.pl/pl/techniki/wzorcowe-projektowe-pamiatkamemento/>

<https://refactoring.guru/design-patterns/memento/csharp/example>

J8. Stwórz klasę Szkoła z polem adres typu string, numer typu int i polem uczniowie typu `List<string>`. Zaimplementuj wzorzec pamiętka.

Polecenie

<http://www.algorytm.org/wzorcowe-projektowe/polecenie-command/command-cs.html>

<https://www.dofactory.com/net/command-design-pattern>

<https://www.c-sharpcorner.com/article/command-patterns-in-C-Sharp/>

J9. Stwórz projekt zgodnie ze wzorcem Polecenie pokazujący operacje na liczbach zespolonych.

Stan

<http://devman.pl/pl/techniki/wzorcowe-projektowe-stanstate/>

<https://refactoring.guru/design-patterns/state/csharp/example>

<https://www.dofactory.com/net/state-design-pattern>

J10. Stwórz projekt symulujący działanie Drukarki na podstawie jej stanu. Przykładowy diagram UML <https://dotnettrickscloud.blob.core.windows.net/img/designpatterns/real-life-example-of-state-design-pattern.png>

Strategia (polityka)

https://www.plukasiewicz.net/Artykuly/Strategy_Pattern

[https://www.altcontroldelete.pl/artykuly/worzec-strategia-przykladowa-implementacja-w-c-/](https://www.altcontroldelete.pl/artykuly/worzec-strategia-przykladowa-implementacja-w-c/)

<https://refactoring.guru/design-patterns/strategy/csharp/example>

J11. Stwórz przykładową klasę `Test` z polem `int`. Następnie stwórz klasę na przechowywanie sortowanej listy na obiekty typu `Test`. Różne mechanizmy sortowania umieść w różnych klasach aby było to zgodne ze wzorcem strategii.