Најверојатно 2^{op} Колоквиум

1. Дизајн е:

a. Моделирање на ниво на домен на објекти од „реалниот свет“ (ова е анализа)
b. Ниту еден од понудените одговори
c. Моделирање на имплементациски објекти ***

2. Објектно ориентираните системи ги искористуваат повторувачките дизајн структури кои промовираат:

a. Грануларност
b. Апстракција ***
c. Елеганција ***
d. Флексибилност ***
e. Ниту еден од понудените одговори
f. Модуларност ***

3. Кој од следните својства НЕ е својства на структурните на Објектно-ориентираниот дизајн?

a. Апстракција
b. Елеганција
c. Ригидност ***
d. Флексибилност

4. Апстракција е отстранување на детали а задржување на основните карактеристики на одредена структура

a. True ***
b. False

5. Дизајн принципот на Отворено/Затворено значи дека:

a. Ниту еден од понудените одговори
b. Класите треба да бидат отворени за проширување а затворени за модификација ***
c. Класите треба да бидат затворени за проширување а отворени за модификација

6. Изберете ги главните тополошки цели на една архитектура:

a. Максимална кохезија во секоја компонента *** (high cohesion)
b. Минимална кохезија во секоја компонента
c. Максимално впарување помеѓу компонентите
d. Минимално впарување помеѓу компонентите *** (minimal/loose coupling)

7. Дизајн шаблони се:

a. Се состојат од класи и/или објект ***
b. Чести решенија за повторувачки дизајн проблеми ***
c. Ниту еден од понудените одговори

8. Дизајн шаблоните се:

a. Јазично зависни
b. Се „микро архитектури“ ***
c. Ниту еден од понудените одговори
d. Имаат 4 основни делови: име, проблем, решение, последици ***

9. Што НЕ содржат шаблоните за дизајн на софтвер?

a. Конвенции.
b. Класи и објекти.
c. Специфични имплементации. ***
d. Зависности.

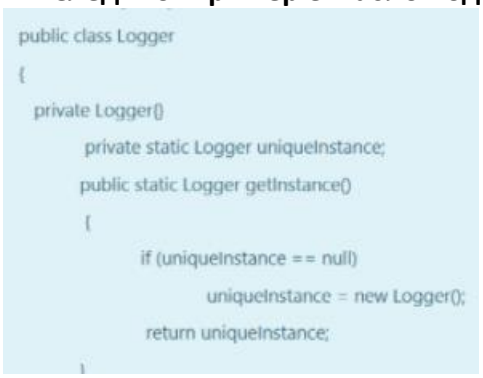
10. Шаблоните за дизајн на софтвер се:

a. Имплементација на специфичен проблем.
b. Дефиниции на функции.
c. Сложени архитектури на целиот систем.
d. Независни од програмскиот јазик. ***

11. Кој од следните шаблони е структурен:

a. Prototype (creational)
b. Abstract Factory (creational)
c. Adapter ***
d. Singleton (creational)

12. Следниот пример е шаблон од типот?



```
public class Logger
{
  private Logger()
        private static Logger uniqueInstance;
        public static Logger getInstance()
        {
              if (uniqueInstance == null)
                    uniqueInstance = new Logger();
              return uniqueInstance;
        }
```

a. Инстанца
b. Ниту еден од понудените одговори
c. Декоратер
d. Синглтон ***
e. Логер
f. Фабрика

13. Singleton овозможува креирање на една инстанца од класата преку повикување на public конструкторот на класата.

a. Да.
b. Не. ***

14. Синглтон шаблонот обезбедува

a. Една класа да има една инстанца и обезбедува статичка точка за пристап до неа
b. Една класа да има една инстанца и обезбедува виртуелна точка за пристап до неа
c. Ниту еден од понудените одговори
d. Една класа да има една инстанца и обезбедува глобална точка за пристап до неа ***
e. Една класа да има повеќе инстанци но обезбедува глобална точка за пристап до само една

15. Следниот код обезбедува

```
public class Singleton
{
  private Singleton() {}
  private static Singleton uniqueInstance;
  public static Singleton getInstance()
  {
     synchronized(Singleton.class) {
       if (uniqueInstance == null)
            uniqueInstance = new Singleton();
     }
     return uniqueInstance;
  }
}
```

a. Ниту еден од понудените одговори
b. Едноставно заклучување при нишки ***
c. Синхронизација на инстанца
d. Двојна синхронизација на инстанца
e. Заклучување со двојна проверка при нишки

16. Бидејќи објектот се инстанцира преку една функција од класата, Singleton по природа е threadsafe и не се потребни дополнителни проверки или механизми за пристап од повеќе тредови.

a. Не. ***
b. Да.

17. Шаблонот Singleton е наједноставниот шаблон бидејќи овозможува едноставно тестирање со Unit тестови.

a. Не. ***
b. Да.

18. Последици од користење Синглтон се:

a. Предизвици при наследување ***
b. Ниту еден од понудените одговори
c. Мрзеливо инстанцирање ***
d. Предизвици при тестирање единици ***
e. Нишки ***
f. Скриени зависности ***

19. При користење на наследување, однесувањето се одлучува за време на

a. Ниту еден од понудените одговори
b. Извршување
c. Компајлирање ***
d. Динамичка композиција

20. Кај шаблонот Стратегија важи:

a. Програмирај на интерфејс не на апстракција
b. Програмирај на апстракција, не на имплементација ***
c. Програмирај на интерфејс, не на имплементација ***
d. Ниту еден на понудените одговори
e. Програмирај на имплементација и на интерфејс

21. Шаблонот Стратегија овозможува

a. Алгоритмите да се енкапсулираат и да се направат меѓусебно заменливи ***
b. Да се дефинира фамилија на алгоритми ***
c. Ниту еден од понудените одговори
d. Алгоритмот да се менува независно од клиентот ***

22. Шаблонот Strategy овозможува:

a. Користење на една имплементација на алгоритам за сите проблеми.
b. Единствена инстанца на класата.
c. Користење на различни алгоритми од страна на објектот во runtime. ***
d. Користење на наредби за условно гранење за избор на соодветниот алгоритам.

23. Шаблонот Стратегија

a. Не овозможува промена на однесувањето за време на извршување
b. Дефинира фамилија на класи кои користат ист интерфејс ***
c. Ниту еден од понудените одговори

24. Hollywood Principle можеме да го забележиме кај:

a. Template. ***
b. Strategy.
c. Ниеден од наведените.
d. Singleton.

25. Холивуд принципот означува дека:

a. Максимизира директно впарување помеѓу конкретни класи
b. Ниту еден од понудените одговори
c. Методите на повисоко ниво не треба да зависат од модулите на пониско ниво ***

26. Шаблонот Темплејт методот

a. Овозможува подкласите да редефинираат одредени чекори на алгоритмот без да ја сменат алгоритамската структура ***

b. Овозможува подкласите да редефинираат одредени чекори на алгоритмот за да сменат алгоритамската структура

c. Дефинира 'рбет на алгоритам во метод, пренесувајќи некој чекори во подкласи ***

d. Ниту еден од понудените одговори

27. Шаблонот Template овозможува менување на чекорите од даден алгоритам од страна на децата класи:

a. Да

b. Не

28. Креирате софтвер за интернет продавница. Продавницата за продажба на книги и продажба на играчки имаат ист алгоритам, но продавницата за играчки мора да има дополнителна можност за приказ на возраста за која е соодветна играчката. За изработка ќе користите:

a. Ниту еден од понудените одговори

b. Фабрика

c. Темплејт ***

d. Синглтон

e. Стратегија

29. Factory Method дефинира:

a. Интерфејс за инстанцирање на апстрактни објекти.

b. Интерфејс за инстанцирање објекти чиј тип е однапред определен со константа.

c. Интерфејс за инстанцирање објекти чиј тип го одлучува подкласата. ***

d. Интерфејс за инстанцирање на објекти чиј тип се определува во runtime.

30. Методот Фабрика мора да врати:

a. Конкретен креатор

b. Креатор

c. Ниту еден од понудените одговори

d. Конкретен продукт ***

e. Продукт

31. Abstract Factory овозможува:

a. креирање на фамилии од поврзани или зависни објекти без да се специфицира конкретната класа. ***

b. креирање на фабрики за објекти.

c. креирање на композиции од објекти.

d. креирање на фамилии од независни објекти.

32. Апстрактна Фабрика всушност содржи еден или повеќе Фабрика методи

a. True ***

b. False

33. Шаблонот Команда енкапсулира

a. Барање како апстрактен интерфејс
b. Ниту еден од понудените одговори
c. Барање како објект ***
d. Барање како параметар на статичка класа

34. Адаптер шаблонот конвертира

a. Објект на дадена класа во друг објект кој клинетот го очекува
b. Интерфејс на дадена класа во друг интерфејс кој клиентот го очекува ***
c. Ниту еден од понудените одговори

35. Шаблонот Adapter овозможува:

a. Промена на интерфејстот на класата во универзален интерфејс кој можа да го користи било која класа
b. Промена на алгоритмите кои се користат од методите на класата во runtime
c. Промена на интерфејсот на класата во некомпатабилен интерфејс
d. Промена на интерфејсот на класата во компатабилен интерфејс кој го очекува клиенетот ***

36. Шаблонот Набљудувач дефинира „многу кон многу“ зависност помеѓу објекти така што кога повеќе објекти ќе променат состојба, еден е известен и автоматски освежен

a. True
b. False ***

37. Во шаблонот MVC, кој друг шаблон се користи за да се имплементира ажурирањето на View-то кога се прави промена на податоците во Model-от:

a. Decorator.
b. Data Loader.
c. Informer.
d. Adapter.
e. Observer. ***

38. Декоратер

a. Динамички додава дополнителни одговорности на соодветен објект ***
b. Ниту еден од понудените одговори
c. Флексибилна алтернатива на користењето подкласи за потреба на проширена функционалност ***

39. Кај MVC шаблон:

a. Промените во моделот не се отсликуваат со погледот
b. Промените во моделот ги пренесува контролерот во погледот
c. Промените во моделот погледот ги добива директно од моделот
d. Контролерот секогаш ги прави промените во моделот???

40. Кај MVC набљудувачот е:

a. Погледот ***
b. Контролерот
c. Ниту еден од понудените одговори
d. Моделот

41. Познати рамки на MVC се:

a. PHP MVC Leopard
b. ASP.NET MVC Framework ***
c. Java Swing ***
d. Apache Struts ***
e. Apple Macbook MVC
f. Ниту еден од понудените одговори

42. Model View Presenter е:

a. Ниту еден од понудените одговори
b. Базиран на MVC ***
c. Се фокусира на презентацискиот слој ***

43. Кој од следниве шаблони би го користеле доколку сакате да претставите хиерархија од објекти:

a. Composite. ***
b. Abstract Factory.
c. Decorator.
d. Facade.

44. Кој шаблон би го користеле доколку промените во однесувањето се случуваат како последица на надворешни фактори:

a. State.
b. Strategy. ***

45. Кој шаблон овозможува справување со мемориски ограничувања во програмата

a. Flyweight ***
b. Factory Method
c. Facade
d. Decorator

46. Кое од следните НЕ е шаблон за дизајн на софтвер:

a. Builder.
b. Observer.
c. Commander. ***
d. Decorator.

47. Бенефити на Облак се

a. Надежност **
b. Ниту еден од понудените одговори
c. Скалабилност **
d. Агилност ***
e. Сигурност **
f. Цена на чинење ***

48. Amazon EC2 е пример за PaaS

a. True
b. False

49. Главни REST операции се

a. GET ***
b. DIR
c. HASH
d. POP
e. POST ***
f. PUT ***
g. DELETE ***

50. Софтверска архитектура е множество на принципиелни дизајн одлуки на системот

a. Точно ***
b. Неточно

51. Наведете и кусо објаснете 5 архитектурни стилови:

- Слоевита архитектура – архитектура поделена во слоеви. Погорните слоеви служат како клиенти, додека подолните служат како сервиси. Во строга слоевита архитектура само соседните слоеви знаат едни за други
- Event based архитектура – клиентите на одреден временски период проверуваат дали има нови информации од системот (pull), додека самиот систем ги објавува новите информации (push). Овој систем не е ефикасен доколку често има нови настани.
- Language based архитектура – оваа архитекура е дефинирана од јазикот кој се користи за изградба на софтверот.
- Интерпретер – типичен пример е Java Virtual Machine
- Data flow – оваа архитектура е одредена од податоците кои течат низ системот.

All:
- Client/Server: N-tier separation, clients independent
- Layered: Strong separation between layers, clear interfaces
- Pipe & Filter: Dynamic reordering, best for stream data
- Peer to Peer: No centralized control, security & dist. Challenges
- Mobile Code: Dynamically move computation to other nodes
- Blackboard: Independent workers collaborate. BB hard to design
- Interpreter: Dynamic evaluation of provided language
- Event-Based: Implicit evaluation of actions, +obliv, -order
- Publish-Subscribe: Excellent broadcast mech.

IDK (not sure where the answers to these can be found, whether they are correct or whether they are from the 2nd midterm)

1. Од аспект на проширливост(скалабилост) топологијата треба да ги има следните карактеристики:
a. Избегнување на тесни грла **
b. Локациска транспарентност**
c. Еластичност на компонентните барања
d. Позиционирање на податоците поблиску до корисникот на истите **

2. Објаснете го архитектурниот стил базиран на настани (event based)?
- Стилот може да биде дефиниран како „промена на состојба“. Истиот се состои од агенти (емитери), потрошувачи (consumers) и канали на настанот. Агентите имаат улога да ги приберат евентите и да ги препратат. Понатаму агентот незнаејќи на кого е наменет овој настан, потрошувачот го прифаќа и работи со истиот. Понекогаш потрошувачот не е последниот со настанот, туку го филтрира, трансформира и препраќа на некоја друга компонента. Каналот на настанот е тој што го поврзува потрошувачот и агентот (emitter/host)
.
3. Наведете пример за софтвер кој би користел архитектурен стил од типот публикувај/претплати се?
- Во cloud-based систем каде што компонентите меѓу себе си праќаат информации каде што наменетите пораки се стават во редица или пак каде што примачите се само заинтересирани за дел од пораката(информацијата). Во овој систем се јавува потреба за публикувај/претплати се. Пораките ќе се праќаат преку канал каде што испраќачот(publisher) ги става настаните во пораки користејќи некој формат. Се користи еден канал од потрошувач(subscriber). Операцијата кај која што се копира една порака од праќачки канал во примачки канал е раководена од message broker или event bus.

4. Каков тип на апстракција е декомпозиција?
a. Латерална
b. Од долу нагоре
c. Лонгитудинална
d. Од горе надолу **

5. Софтвер кој го имплементира TCP/IP протоколот треба да е испрограмиран со кој архитектурен стил и зошто?
- Софтверот треба да биде испрограмиран со Layered architecture – самиот TCP/IP протокол е слоевит протокол, па така оваа архитектура е најпогодна за истиот.