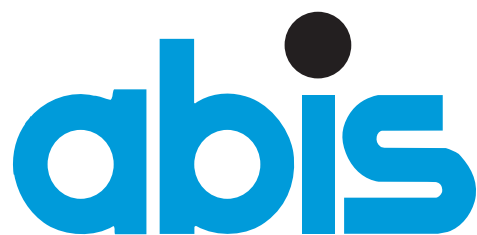


# Zelftest OOAD/UML

Document: N0767Test.fm

30/08/2010

ABIS Training & Consulting  
P.O. Box 220  
B-3000 Leuven  
Belgium



---

TRAINING & CONSULTING

# INTRODUCTIE

Deze test is gebaseerd op de inhoud van onze cursus [OO analyse en design met UML](#) en laat u toe uw kennis over dit onderwerp in te schatten.

De test bestaat uit 20 vragen. Voor elke vraag moet u één antwoord aankruisen (tenzij anders vermeld). Een vraag is pas correct beantwoord wanneer alle correcte antwoordmogelijkheden zijn aangekruist.

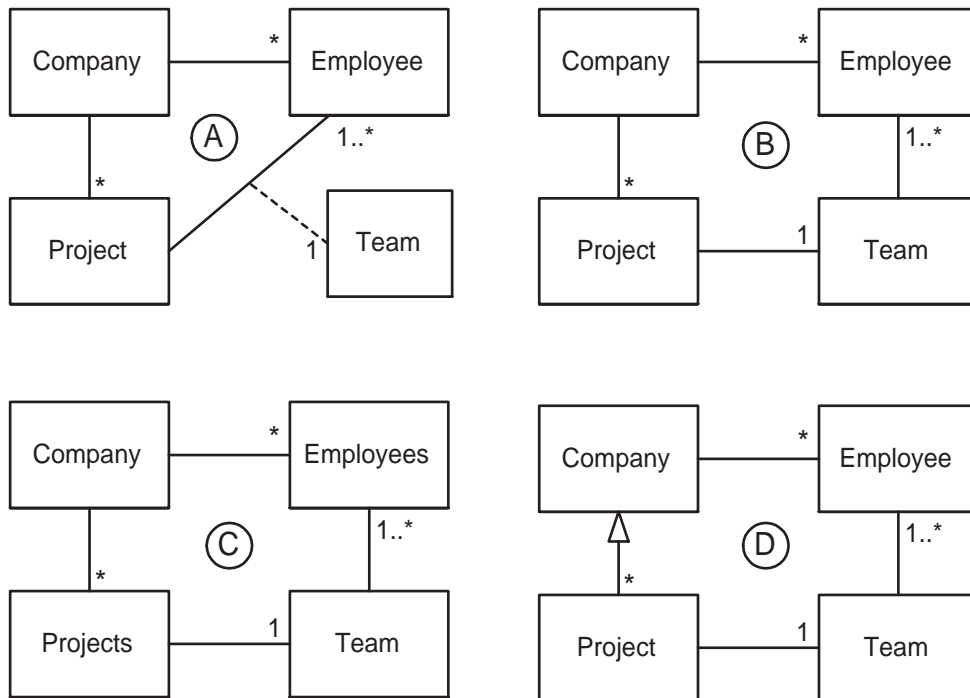
Reken 10 à 15 minuten voor het invullen.

De juiste antwoorden en richtlijnen voor de evaluatie vindt u achteraan dit document.

# VRAGEN

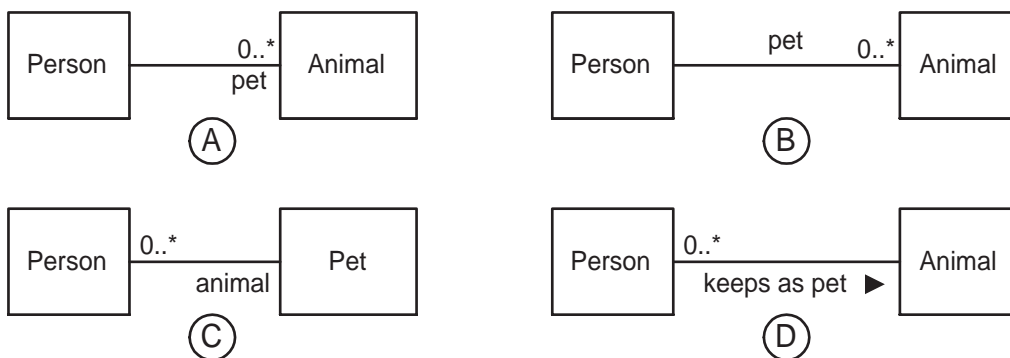
1. Stel: je moet voor een project activiteiten plannen zoals het ontwikkelen van nieuwe functionaliteiten of het maken van test-scenario's, welk OOAD 'artefact' is dan een nuttig hulpmiddel?
  - (a) Sequence diagrams
  - (b) Use cases
  - (c) Domain model
  - (d) Package diagrams
  
2. Welk van onderstaande is iteratief, incrementeel, use case driven en architecture centric?
  - (a) V-method
  - (b) UML
  - (c) Component Based Development
  - (d) RUP
  
3. Welk is een juiste uitspraak over UML stereotypes?
  - (a) Een stereotype wordt gebruikt voor uitbreiding van UML
  - (b) Een stereotyped class moet abstract zijn.
  - (c) Het stereotype {frozen} geeft aan dat een UML element niet kan gewijzigd worden
  - (d) Profielen kunnen gestereotypeerd worden voor compatibiliteitsredenen.
  
4. Stel: je hebt een drankenautomaat (koffie, soep, ...) met een 'customer' als actor en 'machine' als scope. Wat wil je dan aantreffen in het 'main scenario' van de use case 'get drink'?
  - (a) - enter choice  
- if drink available then show price  
- put in coins  
- if paid enough then deliver drink
  - (b) - customer enters choice  
- machine shows price  
- customer puts in coins  
- machine delivers drink
  - (c) - enter choice  
- show price  
- put in coins  
- deliver drink
  - (d) - ...  
- machine sends price to LCD display  
- customer puts coins in slot  
- coin mechanism verifies amount and tells machine controller  
- machine controller activates boiler  
- etc...

5. Beschouw volgende situatie:  
 een company realiseert projecten; elk project wordt uitgevoerd door een team employees.  
 Welk is het meest geschikte conceptueel UML diagram?



- (a) diagram A
- (b) diagram B
- (c) diagram C
- (d) diagram D

6. Hoe druk je uit dat sommige personen dieren houden als huisdieren ('some persons keep animals as pets')?

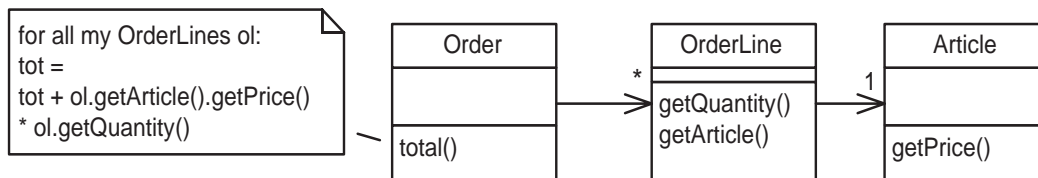


- (a) diagram A
- (b) diagram B
- (c) diagram C
- (d) diagram D

7. Waarvoor kan je UML interfaces gebruiken?

- (a) om concrete klassen te voorzien van de stereotype <<interface>>
- (b) om te programmeren in Java en C++, maar niet in C#
- (c) om uitvoerbare logica te definiëren, die kan hergebruikt worden in verschillende klassen
- (d) om services te specificeren, die door bepaalde objecten vereist worden

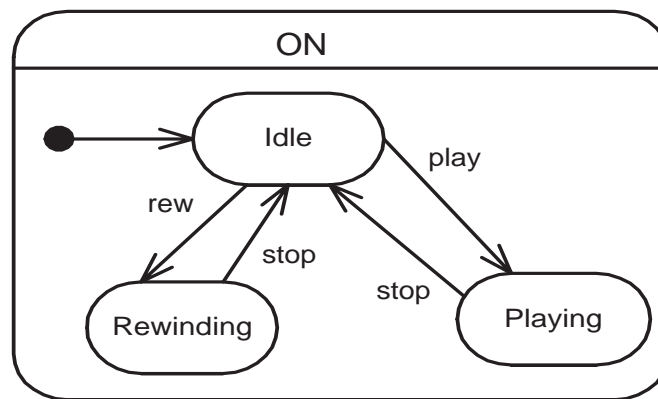
8. Beschouw het volgende ontwerp:.



Hoe zou de invoering van een subtotal() methode in OrderLine het ontwerp verbeteren?

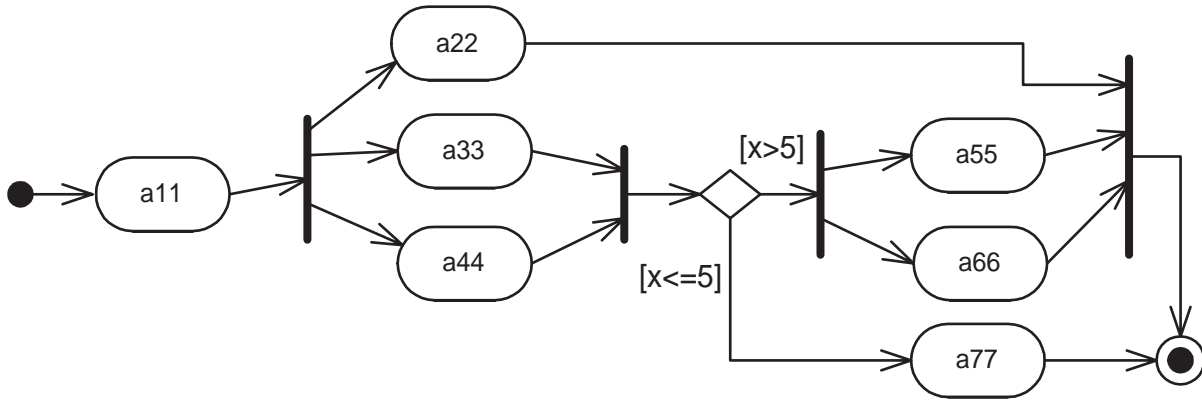
- (a) Het verhoogt de cohesie van Article.
- (b) Het reduceert de koppeling van Order.
- (c) Het reduceert de cohesie van Order.
- (d) Het verhoogt de koppeling van Article.

9. Wat is juist i.v.m. onderstaande State Diagram?



- (a) 'ON' is een 'concurrent state'.
- (b) Deze State Diagram is fout, want hij bevat geen 'final state'.
- (c) 'play', 'stop' en 'rew' zijn actions.
- (d) 'ON' is een 'superstate'.

10. Welke van volgende activiteiten kunnen gelijktijdig plaatsvinden?



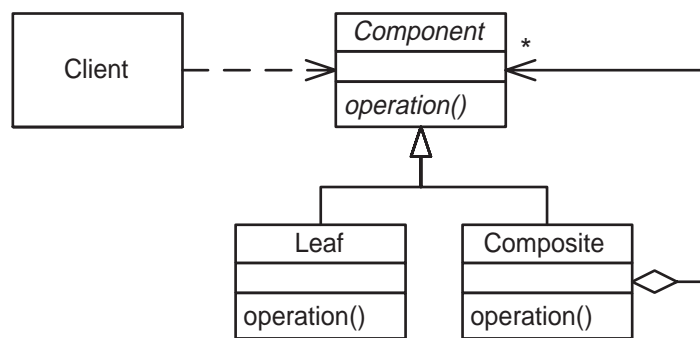
[2 antwoorden]

- [a] a44 en a66
- [b] a44, a33 en a22
- [c] a22 en a77
- [d] a77 en a66

11. Welke zijn geldige gebeurtenissen (events) in een State Diagram? [2 antwoorden]

- [a] if()
- [b] when()
- [c] close()
- [d] after()

12. Dit is de structuur van een bekend GoF pattern. Welk?



- (a) Component
- (b) Delegation
- (c) Polymorphism
- (d) Composite

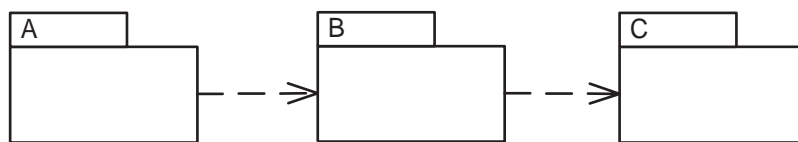
13. Welke van onderstaande zijn bekende refactorings (volgens Fowler)? [2 antwoorden]

- [a] Protect Variations
- [b] Replace Inheritance with Delegation
- [c] Replace Delegation with Inheritance
- [d] Introduce Association Class

14. Welk diagram zou je gebruiken om de fysieke locatie van software-componenten in hardware en netwerk aan te duiden?

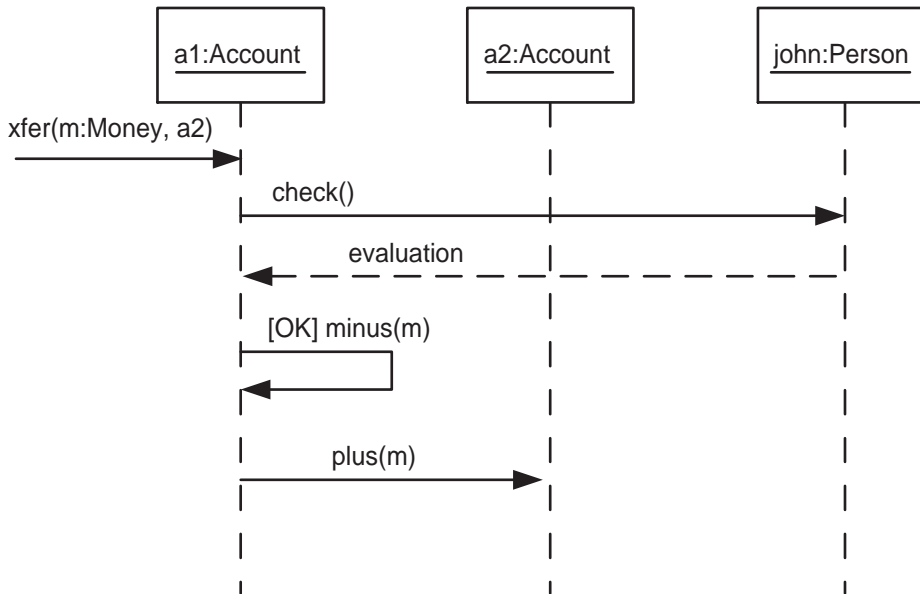
- (a) component diagram
- (b) deployment diagram
- (c) class diagram
- (d) network diagram

15. Welke is een juiste bewering over onderstaande packages?



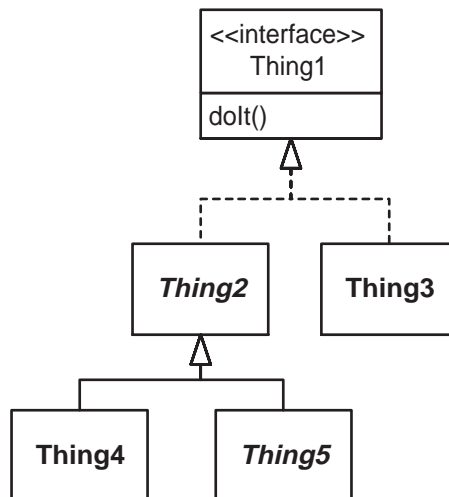
- (a) Als package C verandert, moet je package B nakijken voor de nodige wijzigingen, en misschien moet je daarna ook nog A aanpassen.
- (b) Als package B verandert, dan moeten packages A en C nagezien worden voor de nodige aanpassingen.
- (c) Packages moeten zó ontworpen worden dat een verandering in één package geen effect heeft voor de andere packages.
- (d) Als package C verandert, moet je package A nakijken (zowel als B), omdat afhankelijkheden transitief zijn.

16. Welke operaties moeten geïmplementeerd zijn voor de Account klasse?



- (a) xfer()
- (b) xfer(), plus(), minus()
- (c) check(), plus(), minus()
- (d) xfer(), evaluation(), plus(), minus()

17. Wat geldt i.v.m. de methode dolt()?



- (a) De methode dolt() moet geïmplementeerd worden door Thing3 en misschien ook door Thing4.
- (b) De methode dolt() moet alleen geïmplementeerd worden door Thing5.
- (c) De methode dolt() moet geïmplementeerd worden door Thing2, Thing3, Thing4 en Thing5.
- (d) Geen enkele klasse moet dolt() implementeren, want dat is reeds geïmplementeerd door Thing1.

18. Wat is waar i.v.m. een Sequence Diagram? [2 antwoorden]
- [a] Het beschrijft gedrag binnen meerdere Use Cases.
  - [b] Het beschrijft gedrag binnen één enkele Use Case.
  - [c] Het beschrijft het gedrag van één enkel object.
  - [d] Het beschrijft het gedrag van meerdere objecten.
19. Welk GRASP pattern gebruik je om te beslissen welke klasse de binnenkomende systeemgebeurtenissen ('system events') moet afhandelen?
- (a) Controller
  - (b) Low coupling
  - (c) Adapter
  - (d) Information Expert
20. Welk diagram is NIET gebruikelijk voor het illustreren van use cases?
- (a) system sequence diagram
  - (b) activity diagram
  - (c) use case diagram
  - (d) collaboration diagram

# EVALUATIE.

Hier zijn de correcte antwoorden op alle vragen:

1. b
2. d
3. a
4. b
5. b
6. a
7. d
8. b
9. d
10. b c
11. b d
12. d
13. b c
14. b
15. a
16. b
17. a
18. b d
19. a
20. d

Geef uzelf 1 punt voor elke correct beantwoorde vraag (d.w.z. alle correcte antwoordmogelijkheden aangeduid en geen enkele andere).

Heeft u minder dan 50% van de vragen correct beantwoord, dan raden wij u ten zeerste aan de cursus [OO analyse en design met UML](#) te volgen.

Heeft u tussen de 50% en 75% van de vragen correct beantwoord, dan weet u al heel wat, maar u zal zeker nog bijleren tijdens diezelfde cursus.

Heeft u meer dan 75% correct, dan zal u tijdens de cursus [OO analyse en design met UML](#) niet veel meer bijleren. Misschien is de cursus [Patterns in de praktijk](#) iets voor u?