

# PB007: Ilustrační písemka - Test

Jméno a příjmení - pište do okénka	UČO	Číslo zadání
		1

90 minut celkem; A>=82, 82>B>=74, 74>C>=66, 66>D>=58, 58>E>=50, 50>F

- 1** (5 b.) Jaká z následujících tvrzení týkajících se softwarových procesů jsou pravdivá? // Bodování: součet za jednotlivé
- A** Vodopádový model umožňuje reflektovat časté změny uživatelských požadavků.
  - B** Boehmův spirálový model zahrnuje analýzu rizik a prototypování.
  - C** Reuse-oriented software engineering řeší integraci dříve vyvinutých komponent do nového systému.
  - D** Během iterativního vývoje jsou malé funkční celky často dodávány zákazníkovi a tak zvyšují rychlou použitelnost systému před dokončením jeho vývoje.
  - E** Během inkrementálního vývoje jsou malé funkční celky často dodávány zákazníkovi a tak zvyšují rychlou použitelnost systému před dokončením jeho vývoje.
- 
- 2** (5 b.) Které z následujících typů UML diagramů se používají k zachycení strukturálního pohledu na systém [structural perspective]? // Bodování: úplně správně
- A** Komunikační diagram [Communication diagram]
  - B** Objektový diagram [Object diagram]
  - C** Komponentový diagram [Component diagram]
  - D** Diagram balíčků [Package diagram]
  - E** Diagram případů užití [Use case diagram]
- 
- 3** (5 b.) U kterých z následujících požadavků jde o funkční požadavky? // Bodování: součet za jednotlivé
- A** Uživatel má možnost vypsát historii svých výpůjček.
  - B** Systém musí být vyvinut v jazyce Java.
  - C** Během výpisu historie výpůjček systém dokáže zpracovat 1000 transakcí za sekundu.
  - D** Systém bude každé ráno generovat přehled výpůjček za předchozí den.
  - E** Pravděpodobnost chybného výstupu algoritmu X nepřesáhne 0.2%.
- 
- 4** (5 b.) Uvažujte následující situaci: Aktor v UML diagramu případů užití [Use Case diagram] komunikuje se dvěma případy užití, které sdílí společnou funkcionalitu definovanou samostatným případem užití. Jaký vztah z popisované situace vzniká? // Bodování: úplně správně
- A** Vztah <<include>> mezi případy užití.
  - B** Vztah <<extend>> mezi případy užití.
  - C** Generalizaci mezi případy užití.
  - D** Generalizaci mezi aktory.
- 
- 5** (5 b.) Která z následujících tvrzení jsou pravdivá? // Bodování: součet za jednotlivé
- A** Asociace může existovat mezi třídami i objekty, zatímco link existuje pouze mezi objekty.
  - B** Link je instancí asociace.
  - C** Pokud je mezi dvěma třídami asociace, pak musí existovat i link mezi objekty těchto tříd.
  - D** Pokud je mezi dvěma objekty link, pak musí existovat i asociace mezi třídami těchto objektů.
- 
- 6** (5 b.) Jak se nazývá architektonický vzor, který nejlépe charakterizuje to, že zdůrazňuje rozhraní k subsystémům zapouzdřujícím funkcionalitu na obdobné úrovni abstrakce (v blízkosti UI či HW)? // Bodování: úplně správně
- A** Repository architecture pattern
  - B** Client-server architecture pattern
  - C** Layered architecture pattern
  - D** Pipe and filter architecture pattern
- 
- 7** (5 b.) Jaké jsou výhody softwarových inspekcí oproti testování během validace a verifikace softwarového produktu? // Bodování: součet za jednotlivé
- A** Jsou snáze použitelné pro nekompletní programy.
  - B** Mohou být založeny na analýze jiných dokumentů než kódu.
  - C** Umožňují odhalení nevhodných programátorských stylů, které by mohly v budoucnu způsobit problémy.
  - D** Jsou snadno automatizovatelné.