

Vzorová zkouška – odpovědi

Sada B
Verze 1.7

Učební osnovy ISTQB® Učební osnovy pro základní stupeň Kompatibilní učebními osnovami verze 4.0.1

International Software Testing Qualifications Board



Autorská práva

Informace o autorských právech © International Software Testing Qualifications Board (dále jen ISTQB®).

ISTQB® je registrovaná ochranná známka organizace International Software Testing Qualifications Board.

Všechna práva vyhrazena.

Autoři tímto převádějí autorská práva na ISTQB®. Autoři (jako současní držitelé autorských práv) a ISTQB® (jako budoucí držitel autorských práv) souhlasili s následujícími podmínkami užití:

Výňatky z tohoto dokumentu pro nekomerční použití mohou být kopírovány, je-li uveden jako zdroj.

Jakýkoliv akreditovaný poskytovatel školení může tuto vzorovou zkoušku použít ve svém školicím kurzu v případě, pokud jsou autoři a ISTQB® uvedeni jako zdroj a vlastník autorských práv k této vzorové zkoušce. Dále musí být zajištěno, že jakákoliv propagace takového kurzu bude prováděna až po obdržení oficiální akreditace školicích materiálů od členského výboru ISTQB®.

Kterýkoli jednotlivec nebo skupina jednotlivců může použít tuto vzorovou zkoušku v člancích a knihách, pokud jsou autoři a ISTQB® uvedeni jako zdroj a vlastníci autorských práv.

Jakékoli jiné použití této vzorové zkoušky je bez předchozího písemného souhlasu ISTQB® zakázáno.

Kterýkoli členský výbor ISTQB® může přeložit tuto vzorovou zkoušku za předpokladu, že v přeložené verzi vzorové zkoušky budou uvedena a přeložena výše uvedená upozornění na autorská práva.

Odpovědnost za dokument

Za tento dokument je zodpovědná pracovní skupina ISTQB® Examination Working Group.

Tento dokument je vytvořen a udržován společným týmem ISTQB®, který se skládá z pracovní skupiny Syllabus WG a pracovní skupiny Exam WG.

Poděkování

Tento dokument byl vytvořen hlavním týmem z ISTQB®: Stuart Reid a Adam Roman

Hlavní tým děkuje reviznímu týmu Exam WG, týmu Syllabus WG a členským výborům za jejich návrhy a příspěvky.

Český překlad: David Janota, Petr Neugebauer, Lukáš Piška.

Historie změn

Vzorová zkouška – použitá šablona pro odpovědi Verze 2.11 Datum: 16.10.2023

Verze	Datum	Poznámky
1.7	1.4.2025	Oprava termínů dle slovníku ISTQB.
1.6	8.7.2024	Oprava otázky 32.
1.5	29.5.2024	Drobná oprava odpovědi 28.
1.4	12.4.2024	Oprava odpovědi 2.
1.3	8.1.2024	Koordinace s aktuální verzí otázek.
1.2	5.12.2023	Oprava studijního cíle pro otázku 15.
1.1	15.11.2023	Koordinace s aktuální verzí otázek.
1.0	16.10.2023	První verze.

Obsah

Autorská práva.....	2
Odpovědnost za dokument	2
Poděkování.....	2
Historie změn.....	3
Obsah	4
Úvod	5
Účel tohoto dokumentu.....	5
Instrukce.....	5
Klíč odpovědí	6
Odpovědi	7
1.....	7
2.....	8
3.....	9
4.....	10
5.....	11
6.....	12
7.....	13
8.....	14
9.....	15
10.....	16
11.....	16
12.....	17
13.....	18
14.....	18
15.....	19
16.....	20
17.....	21
18.....	22
19.....	23
20.....	24
21.....	25
22.....	26
23.....	27
24.....	28
25.....	29
26.....	30
27.....	31
28.....	31
29.....	32
30.....	32
31.....	33
32.....	34
33.....	34
34.....	35
35.....	35
36.....	36
37.....	36
38.....	37
39.....	37
40.....	38

Úvod

Účel tohoto dokumentu

Vzorové otázky a odpovědi a související zdůvodnění v této vzorové zkoušce byly vytvořeny týmem odborníků na danou problematiku a zkušených autorů otázek s cílem:

- Pomoci členským výborům ISTQB® a zkušebními komisím při tvorbě zkušebních otázek.
- Poskytnout příklady zkušebních otázek poskytovatelům školení a uchazečům o zkoušky.

Otázky nelze v této podobě použít v jakékoliv oficiální zkoušce.

Skutečné (ostré) zkoušky mohou zahrnovat širokou škálu otázek a tato vzorová zkouška není určena k tomu, aby obsahovala příklady všech možných typů, stylů nebo délek otázek. Dále platí, že tato vzorová zkouška může být těžší, ale i lehčí než zkouška skutečná.

Instrukce

V tomto dokumentu najdete:

- Tabulku odpovědí, kdy je u každé odpovědi uvedeno:
 - K-úroveň, studijní cíl a bodové hodnocení,
- Sady odpovědí, kdy u každé otázky je uvedeno:
 - správná odpověď,
 - odůvodnění každé odpovědi,
 - K-úroveň, studijní cíl a bodové hodnocení.

Otázky jsou obsaženy v samostatném dokumentu.

Klíč odpovědí

Číslo otázky	Správná odpověď	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
1	d	FL-1.2.1	K2	1
2	b	FL-1.2.2	K1	1
3	d	FL-1.3.1	K2	1
4	a	FL-1.4.1	K2	1
5	c	FL-1.4.2	K2	1
6	b	FL-1.4.4	K2	1
7	b	FL-1.5.1	K2	1
8	d	FL-1.5.2	K1	1
9	b	FL-2.1.1	K2	1
10	b	FL-2.1.2	K1	1
11	a	FL-2.1.3	K1	1
12	b	FL-2.1.4	K2	1
13	a	FL-2.2.1	K2	1
14	d	FL-2.2.3	K2	1
15	b	FL-3.1.3	K2	1
16	c	FL-3.2.1	K1	1
17	d	FL-3.2.2	K2	1
18	c	FL-3.2.3	K1	1
19	d	FL-4.1.1	K2	1
20	a	FL-4.2.1	K3	1

Číslo otázky	Správná odpověď	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
21	d	FL-4.2.2	K3	1
22	b	FL-4.2.3	K3	1
23	c	FL-4.2.4	K3	1
24	b	FL-4.3.1	K2	1
25	c	FL-4.3.2	K2	1
26	a, e	FL-4.4.2	K2	1
27	d	FL-4.4.3	K2	1
28	b	FL-4.5.2	K2	1
29	d	FL-4.5.3	K3	1
30	a	FL-5.1.3	K2	1
31	b	FL-5.1.4	K3	1
32	b	FL-5.1.5	K3	1
33	d	FL-5.1.7	K2	1
34	c	FL-5.2.4	K2	1
35	a	FL-5.3.1	K1	1
36	a	FL-5.3.3	K2	1
37	a	FL-5.4.1	K2	1
38	b	FL-5.5.1	K3	1
39	c	FL-6.1.1	K2	1
40	a	FL-6.2.1	K1	1

Odpovědi

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
1	d	<p>a) Není správně. Často je možné využít dynamické testování k vyvolání selhání testovaného objektu způsoby, ke kterým by se uživatelé nikdy nedostali, například pomocí vkládání chyb (fault injection). Nicméně, pokud se nemůže takové selhání nikdy vyskytnout u reálných koncových uživatelů, pak není jeho identifikace nijak zvlášť hodnotná, protože testování je v konečném důsledku zaměřeno na zlepšení pracovního produktu pro koncové uživatele. Trávit čas testováním takových selhání není efektivní využití času testerů.</p> <p>b) Není správně. Statické testování ve formě statické analýzy používají vývojáři k identifikaci defektů v jejich kódu dříve, než toho lze dosáhnout prostřednictvím dynamického testování. Pořád ale platí, že statické testování (a statická analýza) se používá k detekci defektů, nikoli selhání (která jsou identifikována dynamickým testováním). Důvodem, proč je tato odpověď nesprávná, je použití termínu "selhání".</p> <p>c) Není správně. Statická analýza přímo detekuje defekty v kódu, a její výsledky obvykle využívají vývojáři, nikoliv zákazník.</p> <p>d) Je správně. Revize jsou formou statického testování, které lze aplikovat od úplného začátku životního cyklu vývoje softwaru. Nalezení a odstranění defektů v požadavcích tak ušetří pracnost, která by se vyplývala při činnostech vývoje, které by s těmito chybnými požadavky souvisely. Pokud by nebyly tyto defekty včas zjištěny a odstraněny a došlo by k jejich detekci v navazujících pracovních produktech (jako je návrh a kód), muselo by dojít ke změně požadavků.</p>	FL-1.2.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
2	b	<p>a) Není správně. QA se zaměřuje na zlepšování a implementaci procesů s cílem vyhnout se chybám a defektům preventivním způsobem, zatímco testování je jednou z forem QC, která se používá k detekci defektů.</p> <p>b) Je správně. QC se snaží dosáhnout odpovídající úrovně kvality zaměřením se na identifikaci a opravu defektů v produktu. Testování je významnou součástí QC a pomáhá tyto defekty odhalit.</p> <p>c) Není správně. Ačkoli je testování významnou součástí QC a pomáhá odhalovat defekty, existují i další (netestovací) techniky používané v QC, které zahrnují formální metody jako jsou ověření modelu, důkazy správnosti, simulace nebo prototypování.</p> <p>d) Není správně. QA se zaměřuje na zlepšování a implementaci procesů s cílem vyhnout se chybám a defektům preventivním způsobem, zatímco testování je jednou z forem QC, která se používá k detekci defektů.</p>	FL-1.2.2	K1	1

3	d	<p>Princip "kompletní testování není možné" znamená, že není možné otestovat všechny možné kombinace testovacích vstupů za všech okolností s výjimkou triviálních případů. Je lepší zaměřit úsilí na vhodné techniky testování, stanovení priorit testovacích případů a testování založené na rizicích s cílem vybrat z celé množiny vstupů pouze určité vzorky.</p> <p>a) Není správně. Tento princip říká, že není možné otestovat vše kromě triviálních případů. Testování všeho by vyžadovalo testování všech možných variant <u>vstupů</u> za všech okolností, což je obecně neproveditelné, protože existuje prakticky nekonečné množství možností. Testování každého možného <u>očekávaného výsledku</u> tento problém nevyřeší, protože vztah mezi vstupy a očekávanými výsledky se může u každého testovaného objektu lišit. Někdy může existovat prakticky nekonečný počet možných očekávaných výsledků (např. když existuje několik proměnných reprezentujících reálná čísla), zatímco jindy mohou být jen dva (například s jednou proměnnou, kdy je výsledek pravda nebo nepravda).</p> <p>b) Není správně. Princip říká, že není možné testovat všechny možné varianty vstupů za všech okolností. Je to proto, že pro netriviální systémy existuje prakticky nekonečné množství kombinací. V praxi by proto bylo zdokumentování všech možných variant vstupů nereálné, protože by to trvalo nekonečně dlouho.</p> <p>c) Není správně. Zahájení testování co nejdříve s využitím revizí a dalších technik statického testování nevyřeší problém s příliš mnoha možnými testovacími případy. Princip "včasné testování šetří čas a peníze" se týká včasného odstranění defektů s cílem zabránit vzniku následných defektů odvozených pracovních produktů, čímž se sníží náklady a pravděpodobnost selhání.</p> <p>d) Je správně. Použití rozdělení tříd ekvivalence a analýzy hraničních hodnot ke tvorbě testovacích případů je jedním ze způsobů, jak tento princip uplatnit, protože tyto testovací techniky poskytují systematický</p>	FL-1.3.1	K2	1
---	---	---	----------	----	---

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
		návod, jak ze všech možných testovacích případů odvodit jejich konečnou podmnožinu.			
4	a	<p>a) Je správně. Návrh testů zahrnuje použití testovacích podmínek k vytvoření testovacích případů a dalšího testwaru jako jsou požadavky na testovací data a testovací listiny pro průzkumné testování. Specifikovány jsou také požadavky na testovací prostředí včetně potřebné infrastruktury a nástrojů.</p> <p>b) Není správně. Provedení testů zahrnuje provedení testovacích případů (jako součást testovacích procedur), ale přímo nepokrývá ostatní testware uvedený v otázce, tzn. požadavky na testovací data, požadavky na testovací prostředí a testovací podmínky.</p> <p>c) Není správně. Testovací analýza se používá k identifikaci funkcionalit, které vyžadují testování. Je analyzována a definována testovací báze, kdy výstupem jsou prioritizované testovací podmínky spolu se souvisejícími riziky. I když tato aktivita zahrnuje práci s testovacími podmínkami, nepokrývá další testware zmíněný v otázce, tzn. požadavky na testovací data, požadavky na testovací prostředí a testovací případy.</p> <p>d) Není správně. Implementace testování zahrnuje vytváření testovacích procedur jako jsou manuální a automatizované testovací skripty, které se vytvářejí z testovacích případů a mohou být sestaveny do testovacích sad. Testovací procedury jsou prioritizovány a uspořádány v harmonogramu provádění testů. Vytváří se testovací data, nastaví se testovací prostředí a ověří se jeho nastavení. Přestože tato aktivita explicitně pracuje s testovacími případy a může využívat požadavky na testovací data a testovací prostředí k jejich vytvoření, nezahrnuje testovací podmínky.</p>	FL-1.4.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
5	c	<p>a) Není správně. Je nepravděpodobné, že by marketingový tým organizace prováděl testování (i když v některých organizacích může být zapojen do akceptačního testování), takže jeho průměrná úroveň zkušeností (která by byla většinově v oblasti marketingu) pravděpodobně nebude mít vliv na to, jak se provádí testování pro daný testovaný objekt.</p> <p>b) Není správně. Je nepravděpodobné, že úroveň znalostí uživatelů, pro které je systém vyvíjen, ovlivní způsob provádění testování. Jakékoli zapojení uživatelů, které by mohlo ovlivnit způsob provádění testování, bude s větší pravděpodobností výsledkem rozhodnutí učiněných testery, zákazníkem a projektovým manažerem.</p> <p>c) Je správně. Délka praxe členů testovacího týmu pomůže určit potřebné schopnosti a znalosti (např. různých nástrojů a typů defektů), které členové týmu uplatní při testování.</p> <p>d) Není správně. Organizační struktura různých koncových uživatelů (kteří se mohou měnit) může být odlišná. Při testování aplikace tedy nemusí být známo, kdo bude koncovým uživatelem, a organizační struktura tohoto uživatele tak může mít jen malý vliv na způsob provádění testování.</p>	FL-1.4.2	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
6	b	<p>a) Není správně. Trasovatelnost mezi zmírněnými riziky a úspěšnými (passed) testovacími případy neposkytuje dostatek informací. Aby byla rizika testováním zmírněna, musel by k nim být přiřazen úspěšný (passed) testovací případ. Pro posouzení zbytkového (reziduálního) rizika je nezbytná sledovatelnost mezi všemi riziky a výsledky testů, což umožňuje identifikovat jako zbytková ta rizika, která nejsou pokryta žádným úspěšným testem.</p> <p>b) Je správně. Trasovatelnost mezi uživatelskými požadavky a výsledky testů poskytuje informace o tom, které požadavky uživatelů již byly testovány. Tím měří postup projektových prací (v kontextu testování) oproti byznysovým cílům.</p> <p>c) Není správně. Není jasné, zda neúspěšné testovací případy vypovídají o schopnostech testera více než úspěšné testovací případy. Částečně by to záviselo na cílech testování (např. budování důvěry nebo vyvolání selhání). Měření testerů ve smyslu úspěšných a neúspěšných testovacích případů může být také kontraproduktivní, protože by mohlo způsobit to, že testeři optimalizují své testování podle této metriky a nikoliv podle cílů testování.</p> <p>d) Není správně. Trasovatelnost mezi identifikovanými riziky a vytvořenými testovacími podmínkami umožňuje určit, jaké další testovací podmínky je nutné napsat. Určení, která rizika stojí za to testovat, je součástí managementu rizik, zejména pak zmírňování rizik.</p>	FL-1.4.4	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
7	b	<p>a) Není správně. Dobré komunikační dovednosti, schopnost aktivního naslouchání a týmové spolupráce umožňují testerovi efektivně komunikovat se všemi zainteresovanými stranami, nicméně hluboká znalost různých počítačových her, která pomáhá dobře vycházet s jedním z vývojářů, není příkladem obecné dovednosti užitečné pro testery.</p> <p>b) Je správně. Znalost oboru, kterou lze využít pro porozumění a komunikaci s koncovými uživateli a zástupci byznysu, je jednou z obecných dovedností požadovaných po testerech. Tester se zkušenostmi pilota může být schopen lépe posoudit akceptační kritéria řídicího systému vrtulníku.</p> <p>c) Není správně. Ačkoli by schopnost programovat mohla být považována za technickou znalost, která může zvýšit efektivitu používání některých testovacích nástrojů, je nepravděpodobné, že by tato dovednost zlepšila komunikaci s byznysovými analytiky.</p> <p>d) Není správně. Ačkoli důkladnost, smysl pro detail, zvědavost a metodický přístup k identifikaci těžko zjistitelných defektů jsou pro testery užitečné obecné dovednosti, lze pochybovat o tom, že by testeři vygenerovali testovací případy již před zahájením průzkumného testování. Je to proto, že jednou z hlavních zásad průzkumného testování je, že testovací případy jsou generovány během testování (nejsou napsány předem).</p>	FL-1.5.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
8	d	<p>a) Není správně. Týmový přístup znamená, že se jakýkoliv člen týmu s potřebnými dovednostmi a znalostmi může ujmout jakéhokoliv úkolu. To ale neznamená, že může kdykoliv zastat jakoukoliv roli. Obvykle členové týmu přebírají pouze ty role, ve kterých se cítí kompetentní. Nelze obecně říci, že každý člen týmu zvládne jakoukoli roli.</p> <p>b) Není správně. Týmový přístup se vztahuje ke způsobu práce jednoho konkrétního týmu (typicky při agilním vývoji softwaru). Nevztahuje se na to, jak má pracovat více týmů na větších projektech a neříká nic o tom, že na celý projekt je zapotřebí pouze jeden „velký“ tým.</p> <p>c) Není správně. Týmový přístup nepředpokládá, že každý člen týmu bude zapojen do každého důležitého rozhodnutí. Například není nutné, aby se zástupci byznysu (např. vlastník produktu) podíleli na každém technickém rozhodnutí, které nemá vliv na výsledek z pohledu byznysu. Realizace takového přístupu by naopak zbytečně zpomalovala postup týmu.</p> <p>d) Je správně. Týmový přístup co nejefektivněji využívá rozmanité dovednosti každého člena, a tím podporuje lepší týmovou dynamiku, posiluje efektivní komunikaci a spolupráci a vytváří soudržnost (synergii), z níž těží celý projekt.</p>	FL-1.5.2	K1	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
9	b	<p>a) Není správně. Při agilním vývoji softwaru jsou výstupy vytvářeny v každé iteraci a časté dodávání přírůstků vyžaduje rozsáhlé regresní testování. Ačkoli některé (nebo všechny) testy z tohoto regresního testování mohou být automatizované, regresní testování (ať už automatizované nebo manuální) nemůže být nahrazeno automatizací systémových testů.</p> <p>b) Je správně. Pokud se používá sekvenční vývojový model, pak není v raných fázích životního cyklu k dispozici žádný spustitelný kód, a proto se během této doby provádí statické testování (např. revize). V pozdějších fázích životního cyklu (kdy už je k dispozici spustitelný kód) je možné provádět dynamické testování. Je nutno ale poznamenat, že příprava na dynamické testování často probíhá právě v těchto raných fázích životního cyklu vývoje softwaru.</p> <p>c) Není správně. Pokud se používá iterativní vývojový model (např. agilní vývoj softwaru), pak lze pro regresní testování v každé iteraci použít testy komponent. V takovém případě existuje pádný důvod pro jejich automatizaci, protože budou spouštěny často. Naopak je jen malá pravděpodobnost, že by se našel pádný argument pro to, aby je vývojáři prováděli manuálně.</p> <p>d) Není správně. Ve většině modelů inkrementálního vývoje jsou výstupy vytvářeny v každém přírůstku, což vyžaduje statické i dynamické testování ve všech úrovních testování pro každý dodaný přírůstek.</p>	FL-2.1.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
10	b	<p>a) Není správně. Testeři by měli revidovat pracovní produkty v okamžiku, kdy jsou k dispozici pracovní verze tak, aby využili včasné testování (jako součást principu shift left). Pokud by čekali do další vývojové fáze, pak by mohly být zahájeny zbytečné vývojové (a testovací) práce na nerevidovaných chybných pracovních produktech.</p> <p>b) Je správně. Testeři by měli revidovat pracovní produkty v okamžiku, kdy jsou k dispozici pracovní verze tak, aby využili včasné testování (jako součást principu shift left).</p> <p>c) Není správně. Testeři obvykle revidují ty pracovní produkty, které tvoří testovací bázi v rámci testovací analýzy, nikoli před testovací analýzou a návrhem testů.</p> <p>d) Není správně. Testeři by měli revidovat pracovní produkty v okamžiku, kdy jsou k dispozici pracovní verze (drafty) tak, aby využili včasné testování (jako součást principu shift left). Čekání na jejich zveřejnění znamená, že všechny defekty, které by mohly být nalezeny pomocí revize, budou ve zveřejněném pracovním produktu.</p>	FL-2.1.2	K1	1
11	a	<p>a) Je správně. Vývoj řízený testováním (TDD) je známým příkladem vývoje iniciovaného testy.</p> <p>b) Není správně. Vývoj řízený pokrytím není správným příkladem vývoje iniciovaného testy.</p> <p>c) Není správně. Vývoj řízený kvalitou není správným příkladem vývoje iniciovaného testy.</p> <p>d) Není správně. Vývoj řízený (užitnými) vlastnostmi není příkladem vývoje iniciovaného testy, ale jedná se o agilní metodiku vývoje softwaru založenou na dodávání užitečných vlastností (na rozdíl od uživatelských scénářů ve Scrumu).</p>	FL-2.1.3	K1	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
12	b	<p>a) Není správně. DevOps obohacuje testování několika způsoby, například poskytováním rychlé zpětné vazby o kvalitě kódu, automatizovaným regresním testováním (které minimalizuje riziko regrese) a podporou principu shift left s dodáváním vysoce kvalitního kódu a testů komponent. To je z velké části zajištěno prostřednictvím průběžné integrace, kdy vývojáři dodávají společně se svým novým kódem také testy komponent, které musí projít, jinak není kód akceptován do sestavení (buildu). Vývojáři proto musí dokončit testování komponent.</p> <p>b) Je správně. DevOps obohacuje testování několika způsoby, například poskytováním rychlé zpětné vazby o kvalitě kódu, automatizovaným regresním testováním, které minimalizuje riziko regrese, a podporou principu shift left s dodáváním vysoce kvalitního kódu a testů komponent.</p> <p>c) Není správně. DevOps obohacuje testování několika způsoby, například poskytováním rychlé zpětné vazby o kvalitě kódu, automatizovaným regresním testováním (které minimalizuje riziko regrese) a podporou principu shift left s dodáváním vysoce kvalitního kódu a testů komponent. Testeři se nesnaží vyvažovat svůj přístup k vývojářům a provozu tím, že by věnovali více času testování vydání, ačkoliv je možné tento princip (tzv. shift right neboli testování v produkci) možné klidně používat.</p> <p>d) Není správně. Automatizované procesy, jako je průběžná integrace/průběžné doručování (CI/CD) v DevOps, pomáhají se stabilizací testovacího prostředí a snižují potřebu manuálního testování, existuje však riziko přehlédnutí důležitosti manuálního testování, zejména z pohledu uživatele.</p>	FL-2.1.4	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
13	a	<p>a) Je správně. Systémové testování zkoumá chování a schopnosti celého systému a zahrnuje nefunkcionální testování kvalitativních charakteristik, například testování bezpečnosti. Tento typ testování je často prováděn nezávislým testovacím týmem na základě specifikací systému.</p> <p>b) Není správně. Systémové integrační testování zkoumá rozhraní s jinými systémy a externími službami.</p> <p>c) Není správně. Beta testování je typ akceptačního testování prováděného v externí lokalitě rolemi mimo vývojovou organizaci.</p> <p>d) Není správně. Integrační testování komponent zahrnuje testování interakcí mezi komponentami systému, což jsou například uživatelská rozhraní a databáze.</p>	FL-2.2.1	K2	1
14	d	<p>a) Není správně. Počet regresních testů v průběhu projektu roste, protože se změnami v systému jsou obvykle zapotřebí nové regresní testy. Podobně se obvykle zvyšuje také počet konfirmačních testů, protože pro každou opravu systému jsou potřeba nové konfirmační testy.</p> <p>b) Není správně. Je to naopak. Konfirmační testy jsou vytvořeny a spuštěny, když je testovaný objekt opravován, a regresní testy jsou (v ideálním případě) spuštěny vždy, když je testovaný objekt rozšiřován (měněn).</p> <p>c) Není správně. Konfirmační testování ověřuje, že defekt byl správně opraven, a proto se týká testování změn testovaného objektu. Regresní testování však zajišťuje, aby změny (včetně změn provozního prostředí) neměly negativní vliv na nezměněný software, a proto nekontroluje, zda provozní prostředí zůstává nezměněno.</p> <p>d) Je správně. Regresní testování zajišťuje, že změny nebudou mít negativní vliv na nezměněný software. Konfirmační testování ověřuje, že defekt byl opraven, týká se tedy změněného kódu.</p>	FL-2.2.3	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
15	b	<p>a) Není správně. Nedostatečnou použitelnost poskytovanou prostřednictvím uživatelského rozhraní lze zjistit prostřednictvím revize s využitím vhodného kontrolního seznamu. Nedostatečnou použitelnost lze také identifikovat tak, že několik typických uživatelů dynamicky testuje uživatelské rozhraní a poskytuje zpětnou vazbu o jeho použitelnosti.</p> <p>b) Je správně. Revize kódu může detekovat kód, který nemůže být žádným způsobem spuštěn (tzv. nedosažitelný kód). Dynamické testy mohou testovat pouze dosažitelný kód a nemohou tedy zjistit, že daný kód nelze spustit při všech možných kombinacích vstupů a stavů vstupů.</p> <p>c) Není správně. Špatnou dobu odezvy u většiny očekávaných uživatelů je obtížné určit bez spuštění kódu (tj. statickým testováním), takže v této situaci by dynamické testování defekt najít mohlo, naopak statické testování jej pravděpodobně nenajde.</p> <p>d) Není správně. Revize kódu někým, kdo zná požadované funkcionality, by mohla identifikovat, že nebyly implementovány v kódu. Dynamické testování by mohlo být použito ke zjištění, že obecně nebyly implementovány vůbec.</p>	FL-3.1.3	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
16	c	<p>a) Není správně. Zpětná vazba pochází od zainteresovaných stran (např. zástupců byznysu nebo koncových uživatelů), nikoli od vývojářů, takže pravděpodobně neinformuje manažery o tom, kteří vývojáři jsou více či méně produktivní.</p> <p>b) Není správně. Včasnou a častou zpětnou vazbu od zainteresovaných stran nepoužívají projektoví manažeři k tomu, aby upřednostnili způsob, jakým komunikují s různými zainteresovanými stranami.</p> <p>c) Je správně. Získání zpětné vazby od zainteresovaných stran včasné a často v procesu vývoje softwaru může být velmi přínosné, protože umožňuje včasnou komunikaci o potenciálních problémech s kvalitou, může zabránit nedorozuměním ohledně požadavků a zajišťuje, že jakékoli změny v požadavcích zainteresovaných stran budou pochopeny a implementovány včas.</p> <p>d) Není správně. Včasná a častá zpětná vazba může zabránit vývoji produktu, který nesplňuje potřeby zainteresovaných stran a/nebo vede k nákladnému přepracování a zmeškání termínů. V ideálním případě by tedy nemělo dojít k žádnému zpoždění. Zpětná vazba navíc směřuje od zainteresovaných stran (což jsou také koncoví uživatelé), nikoliv k nim. Samotné poskytování zpětné vazby ze strany uživatelů tedy nezlepšuje jejich vlastní porozumění.</p>	FL-3.2.1	K1	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
17	d	<p>S ohledem na každý z uvedených popisů úkolů platí, že:</p> <ol style="list-style-type: none"> Jsou vybrány hodnocené kvalitativní charakteristiky a výstupní kritéria. Plánování (C): Definování rozsahu revize, účelu, pracovního produktu, který má být přezkoumán, kvalitativních charakteristik, které mají být hodnoceny, oblastí zaměření, výstupních kritérií, podpůrných informací, což jsou např. normy, pracnost a časové rámce. Každý má přístup k pracovnímu produktu. Zahájení revize (B): Zajistit, aby všichni účastníci měli přístup k pracovnímu produktu a potřebným zdrojům, a vyjasnit jejich role a odpovědnosti. Anomálie v pracovním produktu jsou identifikovány. Individuální revize (A): Vyhodnocení kvality pracovního produktu, identifikace a zaznamenání anomálií, doporučení a otázek pomocí technik revize jako je revize založená na kontrolních seznamech a revize založená na scénářích. Anomálie jsou analyzovány a diskutovány. Komunikace a analýza (D): Analýza a diskuze o každé anomálii, určení jejího stavu, vlastnictví a požadovaných akcí, a učinit rozhodnutí při revizi, obvykle na schůzce. To by mohlo zahrnovat určení potřeby následné revize. <p>Tedy:</p> <ol style="list-style-type: none"> Není správně. Není správně. Není správně. Je správně. Správná kombinace je: 1C, 2B, 3A, 4D 	FL-3.2.2	K2	1

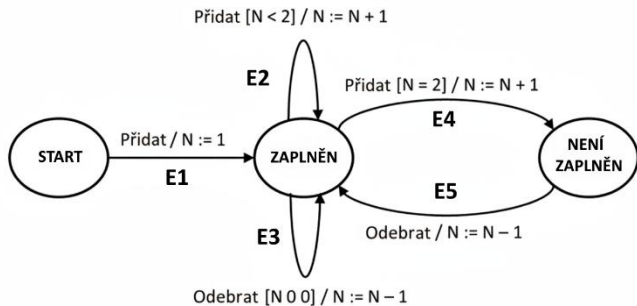
Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
18	c	<p>S ohledem na každou z uvedených rolí platí, že:</p> <ol style="list-style-type: none"> 1. Zapisovatel – zodpovídá za shromažďování zpětné vazby od revidujících a zdokumentování informací o revizi jako jsou učiněná rozhodnutí a jakékoli nové anomálie zjištěné během revizní schůzky. B: Zaznamenává informace o revizi, jako jsou rozhodnutí a nové anomálie zjištěné během revizní schůzky. 2. Vedoucí revize – zodpovídá za dohled nad procesem revize jako je výběr členů revizního týmu, plánování revizních schůzek a zajištění úspěšného dokončení revize. D: Přebírá celkovou odpovědnost za revizi, například organizuje, kdy a kde se bude revize konat. 3. Facilitátor (nebo moderátor) - odpovídá za zajištění efektivního průběhu revizních schůzek včetně řízení času, vedení diskusí a vytvoření komfortního prostředí, kde může každý svobodně vyjádřit svůj názor. A: Zajišťuje efektivní průběh revizních schůzek a nastavení komfortního prostředí pro revizi. 4. Manažer – odpovídá za rozhodování o tom, co je třeba revidovat, a za přidělování zdrojů pro revizi (např. lidé nebo čas. C: Manažer – rozhoduje o tom, co má být revidováno, a rozhoduje o zdrojích jako jsou lidé nebo čas. <p>Tedy:</p> <ol style="list-style-type: none"> a) Není správně. b) Není správně. c) Je správně. Správná kombinace je: 1B, 2D, 3A, 4C. d) Není správně. 	FL-3.2.3	K1	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
19	d	<p>a) Není správně. Testování dle rozhodovací tabulky je technika testování černé skříňky, nikoli technika testování bílé skříňky. Testovací případy nejsou tedy založeny na rozhodnutích ve zdrojovém kódu. Při testování větví jsou testovací případy odvozeny ze znalosti řídicích toků testovaného objektu.</p> <p>b) Není správně. Předvídání potenciálních defektů se používá při odhadování chyb (technika testování založená na zkušenostech), nikoli při testování větví (technika testování bílé skříňky). Při testování dle rozhodovací tabulky jsou testovací případy odvozeny ze specifikace, která popisuje byznysovou logiku.</p> <p>c) Není správně. Pokud je testovací případ založen na znalosti řídicího toku testovaného objektu, jedná se o techniku testování bílé skříňky. Testování dle rozhodovací tabulky je obvykle založeno na analýze byznysové logiky, jedná se tedy o techniku testování černé skříňky. Při testování větví nejsou testovací případy odvozeny ze specifikace – to by z něj udělalo techniku testování černé skříňky. Testování větví je technika testování bílé skříňky, kdy jsou testovací případy odvozeny na základě struktury zdrojového kódu.</p> <p>d) Je správně. Testování dle rozhodovací tabulky je technika testování černé skříňky, takže je založena na analýze specifikovaného chování testovaného objektu bez odkazu na jeho vnitřní strukturu. Testovací případy jsou tedy nezávislé na tom, jak je software implementován. Testování větví je technika testování bílé skříňky, takže testovací případy jsou založeny na analýze vnitřní struktury a procesů testovaného objektu. Vzhledem k tomu, že testovací případy jsou závislé na návrhu a implementaci daného softwaru, mohou být vytvořeny pouze po jeho ukončení nebo po jeho implementaci (pokud není použita technika TDD).</p>	FL-4.1.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
20	a	<p>a) Je správně. 19 pokrývá třídu "bez slevy", 20 pokrývá třídu "50 % sleva" a 30 pokrývá třídu "10 % sleva". Tyto tři hodnoty pokrývají všechny tři platné třídy ekvivalence.</p> <p>b) Není správně. 11 a 12 pokrývají třídu "bez slevy", zatímco 20 pokrývá třídu "50 % sleva", takže celkově pokrývají dvě ze tří platných tříd ekvivalence.</p> <p>c) Není správně. 1 pokrývá třídu "bez slevy", zatímco 10 a 50 pokrývají třídu "10 % sleva". Třída "50 % sleva" není pokryta, takže celkově jsou pokryty dvě ze tří platných tříd ekvivalence.</p> <p>d) Není správně. 29 a 31 pokrývají třídu "bez slevy", zatímco 10 a 30 pokrývají třídu "10 % sleva". Třída "50 % sleva" není pokryta, takže celkově jsou pokryty dvě ze tří platných tříd ekvivalence.</p>	FL-4.2.1	K3	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
21	d	<p>Obor hodnot pro délku hesla má tři třídy ekvivalence:</p> <ul style="list-style-type: none"> hesla příliš krátká {0, 1, ..., 4, 5} správná hesla {6, 7, ..., 11, 12} hesla příliš dlouhá {13, 14, ...} <p>Abychom dosáhli plného pokrytí pro 3-hodnotovou BVA, musíme otestovat následující hodnoty: 0, 1, 4, 5, 6, 7, 11, 12, 13, 14.</p> <p>Vzhledem k tomu, že 2-hodnotová BVA je již pokryta, znamená to, že jsme již otestovali hesla délky: 0, 5, 6, 12 a 13.</p> <p>To znamená, že další délky, které je potřeba pokrýt, aby bylo možné přejít z 2-hodnotové na 3-hodnotovou BVA, jsou: 1, 4, 7, 11 a 14.</p> <p>Tedy:</p> <ol style="list-style-type: none"> Není správně. Není správně. Není správně. Je správně. 	FL-4.2.2	K3	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
22	b	<p>V rozhodovací tabulce je pět sloupců. Každý testovací případ pokrývá jeden z nich.</p> <p>TC1 a TC2 pokrývají Pravidlo 4. TC3 a TC4 pokrývají Pravidlo 2. TC5 pokrývá Pravidlo 5.</p> <p>Těchto pět testovacích případů tedy pokrývá tři z pěti sloupců a dosahuje pokrytí $(3/5) \cdot 100 \% = 60 \%$. Proto je správně možnost b).</p> <p>Tedy:</p> <p>a) Není správně. b) Je správně. c) Není správně. d) Není správně.</p>	FL-4.2.3	K3	1

23	c	 <p>Označme přechody E1 – E5 (viz obrázek). Proměnná N označuje počet aktuálně uložených prvků. Každá událost "Přidat" ji zvyšuje o 1 a každá událost "Odebrat" ji snižuje o 1. Pokud dojde k události "Přidat" ve stavu NENÍ ZAPLNĚN, stav se změní na ZAPLNĚN pouze v případě, že $N=2$. Pokud je $N < 2$, systém zůstane ve stavu NENÍ ZAPLNĚN. Pokud $N=0$, není možná žádná akce "Odebrat". Podobně, pokud $N=3$, není možná žádná akce "Přidat".</p> <p>Test a) lze zapsat jako E1, E3, E3, E2, E4 (pokrývá tedy 4 z 5 platných přechodů a dosahuje 80% pokrytí platných přechodů). Test b) je neproveditelný, protože po prvních třech akcích "Přidat" je systém ve stavu ZAPLNĚN a nedochází k platnému přechodu ze ZAPLNĚN vyvolanému událostí "Přidat". Po prvních třech přechodech je dosaženo pouze 60% pokrytí platných přechodů. Test c) lze zapsat jako E1, E2, E4, E5, E3 (pokrývá tedy 5 z 5 platných přechodů, čímž dosahuje 100% pokrytí platných přechodů). Test d) lze zapsat jako E1, E2, E4, E5, E4 (pokrývá tedy 4 z 5 platných přechodů, dosahuje 80% pokrytí platných přechodů).</p> <p>Tedy:</p>	FL-4.2.4	K3	1
----	---	--	----------	----	---

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
		a) Není správně. b) Není správně. c) Je správně. d) Není správně.			
24	b	a) Není správně. Pokrytí je vždy definováno jako procento pokrytých prvků, proto nemůže překročit 100 %. b) Je správně. Pokud by příkazy prováděné T1 a T2 byly disjunktní, pokrytí testovací sadou {T1, T2} by bylo 105 %, což je nemožné (viz odpověď a). Proto musí být alespoň 5 % spustitelných příkazů provedeno T1 i T2. c) Není správně. Pokrytí příkazů nám neříká nic o počtu nespustitelných příkazů v kódu. d) Není správně. I v případě, že testovací sada dosáhne úplného pokrytí příkazů, neznamená to, že dosáhne úplného pokrytí větví.	FL-4.3.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
25	c	<p>Testování větví je technika testování bílé skříňky, ve které jsou položkami pokrytí větve. Větev je přenos řízení mezi dvěma uzly v grafu řídicího toku zobrazujícího možné sekvence, ve kterých jsou příkazy zdrojového kódu v testovacím objektu prováděny. Každý přenos řízení může být buď nepodmíněný (tj. lineární kód) nebo podmíněný (tj. výsledek rozhodnutí). Pokrytí se měří jako počet větví pokrytých testy k celkovému počtu větví (obvykle vyjádřeno v procentech).</p> <p>Tedy:</p> <ul style="list-style-type: none"> a) Není správně. Výsledek rozhodnutí je podmíněná větev. Pro testování větví se do X započítávají nejen podmíněné, ale i nepodmíněné větve. b) Není správně. Pokrytí větví počítá nejen podmíněné, ale i nepodmíněné větve. c) Je správně. Pokrytí se měří jako počet větví pokrytých testy k celkovému počtu větví (obvykle vyjádřeno v procentech). d) Není správně. X i Y počítají pouze podmíněné větve a neberou v úvahu nepodmíněné větve. 	FL-4.3.2	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
26	a, e	<p>Průzkumné testování je nejvíce užitečné v případech, kdy je specifikace nedostatečná nebo úplně chybí, taktéž je vhodnou metodou v případě výrazného časového tlaku na testování. Průzkumné testování je také výhodné jako doplněk jiných (formálnějších) technik testování. Průzkumné testování bude účinnější v případech, kdy je tester zkušený, má znalosti z daného odvětví a vysokou úroveň klíčových dovedností jako jsou analytické schopnosti, zvědavost a kreativita.</p> <p>Tedy:</p> <ul style="list-style-type: none"> a) Je správně. Průzkumné testování je nejvíce užitečné v případech, kdy je specifikace nedostatečná nebo úplně chybí, taktéž je vhodnou metodou v případě výrazného časového tlaku na testování. b) Není správně. Průzkumné testování není technikou testování černé skříňky. c) Není správně. Průzkumné testování je užitečné, když jsou specifikace napsané špatně. d) Není správně. Programátorské dovednosti v zásadě nemají s průzkumným testováním nic společného. e) Je správně. Průzkumné testování bude účinnější v případech, kdy je tester zkušený, má znalosti z daného odvětví a vysokou úroveň klíčových dovedností jako jsou analytické schopnosti, zvědavost a kreativita. 	FL-4.4.2	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
27	d	<p>a) Není správně. Kontrolní seznamy by měly obsahovat testovací podmínky, které mají být ověřeny. Tato odpověď je příkladem chyby, nikoli testovací podmínky. I kdyby byl tester schopen odvodit některé možné testovací podmínky z možných chyb, je tento popis chyby i tak příliš obecný.</p> <p>b) Není správně. Kontrolní seznamy by neměly obsahovat položky, které jsou vhodnější jako výstupní kritéria. Tato odpověď je příkladem výstupního kritéria.</p> <p>c) Není správně. Kontrolní seznamy by neměly obsahovat položky, které jsou příliš obecné. Tato odpověď je velmi obecná věta, která vlastně popisuje cíl testování.</p> <p>d) Je správně. Toto je příklad testovací podmínky, kterou může prověřit člověk.</p>	FL-4.4.3	K2	1
28	b	<p>a) Není správně. Formát orientovaný na pravidla zahrnuje formáty, jako jsou ověřovací seznamy (checklisty) ve formě odrážek nebo tabulky mapující vstupy na výstupy zobrazující explicitně pravidla, která je třeba dodržovat. Given/When/Then je formát orientovaný na scénáře, protože popisuje scénář, který má být ověřen.</p> <p>b) Je správně. Jedná se o formát Given/When/Then, který je orientován na scénáře.</p> <p>c) Není správně. Neexistuje žádný formát akceptačních kritérií orientovaný na produkt.</p> <p>d) Není správně. Neexistuje žádný formát akceptačních kritérií orientovaný na proces.</p>	FL-4.5.2	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
29	d	<p>a) Není správně. Testovací případ souvisí se zobrazením předchozích objednávek v historii objednávek.</p> <p>b) Není správně. Testovací případ souvisí se zobrazením předchozích objednávek.</p> <p>c) Není správně. Testovací případ souvisí se zobrazením předchozích objednávek v historii objednávek.</p> <p>d) Je správně. Testovací případ souvisí s procesem registrace, který ale není v tomto uživatelském scénáři popisován (tento uživatelský scénář popisuje zobrazení předchozích objednávek).</p>	FL-4.5.3	K3	1
30	a	<p>a) Je správně. To je něco, co může (a mělo by) být zkontrolováno před odesláním kódu do repozitáře.</p> <p>b) Není správně. To je něco, co lze zkontrolovat po provedení kroku (2), protože hlášení konfliktů při sloučení lze provést po odeslání a sloučení kódu.</p> <p>c) Není správně. To se lépe hodí jako vstupní kritérium pro krok (3).</p> <p>d) Není správně. To se lépe hodí jako výstupní kritérium pro krok (3).</p>	FL-5.1.3	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
31	b	<p>Průměrná pracnost vývoje je 900 000 Kč a průměrná pracnost testování je 90 000 Kč (počítáno ze čtyř projektů). Průměrný poměr pracnosti mezi testováním a vývojem je 1:10 (90 000 Kč : 900 000 Kč), což znamená, že historicky tvoří testování v průměru 10 % pracnosti vývoje. Pokud se tedy pracnost vývoje odhaduje na 800 000 Kč, pracnost testování se odhaduje takto: $10\% \cdot 800\,000\text{ Kč} = 0,1 \cdot 800\,000\text{ Kč} = 80\,000\text{ Kč}.$</p> <p>Tedy:</p> <ul style="list-style-type: none"> a) Není správně. b) Je správně. c) Není správně. d) Není správně. 	FL-5.1.4	K3	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
32	b	<p>Podle závislostí musí být nejprve provedeny testy VYHLEDAT, ZOBRAZIT a PŘIDAT, a to před provedením VYTVOŘIT. Před spuštěním VYTVOŘIT je možné přidat více produktů (použitím stejného postupu).</p> <p>Na základě toho musí být TC1 nebo TC2 proveden jako první. Prvními testy by měly být ZOBRAZIT a PŘIDAT produkt B, protože jeho testovací případy (TC6, TC4) mají přiřazenou vyšší prioritu. První 3 testy k provedení jsou tedy TC2 -> TC4 -> TC6.</p> <p>Nyní musíme zvážit, zda spustit TC7 a poté celý průběh pro produkt A, nebo nejprve spustit všechny testy pro produkt A. Pokud má TC7 nižší prioritu než ostatní testy, měly by být tyto testy provedeny jako první. Proto by celý postup měl být: TC2 -> TC4 -> TC6 -> TC1 -> TC3 -> TC5 -> TC7.</p> <p>a) Není správně. TC1 musí být proveden před TC3. b) Je správně. c) Není správně. Jak je uvedeno výše, TC7 je v provedení testů poslední. d) Není správně. Produkt B musí být proveden před produktem A.</p>	FL-5.1.5	K3	1
33	d	<p>a) Není správně. Testování použitelnosti je testování zaměřené na byznys a kritiku produktu (Q3). b) Není správně. Funkcionální testování je testování zaměřené na byznys a podporující tým (Q2). c) Není správně. Uživatelské akceptační testování je testování zaměřené na byznys a kritiku produktu (Q3). d) Je správně. Integrační testování komponent je testování zaměřené na technologii, které podporuje tým (vede vývoj) (Q1).</p>	FL-5.1.7	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
34	c	<p>S ohledem na každé z uvedených rizik a opatření k jejich zmírnění platí, že:</p> <ol style="list-style-type: none"> 1. Dlouhé odezvy systému (1) lze testovat při testování výkonnostní efektivity (výkonnosti) (B). 2. Změny preferencí spotřebitelů (2) jsou obvykle mimo naši kontrolu, proto je toto riziko obvykle akceptováno (A). 3. Vytopení serverovny (3) může způsobit značné ztráty, proto by mělo dojít k přenosu (transferu) rizika, např. sjednáním pojistky (D). 4. To, že pacienti nad určitý věk dostávají nepřesné reporty (4), naznačuje potenciální hraniční problém, který lze účinně detekovat technikami, jako je analýza hraničních hodnot (C). <p>Tedy:</p> <ol style="list-style-type: none"> a) Není správně. b) Není správně. c) Je správně. Správné kombinace rizik a opatření k jejich zmírnění jsou: 1B, 2A, 3D a 4C. d) Není správně. 	FL-5.2.4	K2	1
35	a	<ol style="list-style-type: none"> a) Je správně. Metriky týkající se kvality produktu měří charakteristiky kvality. Střední doba do poruchy (MTTF) měří vyspělost, takže se jedná o metriku týkající se kvality produktu. b) Není správně. Toto je příklad metriky týkající se defektů, nikoli metriky týkající se kvality produktu. c) Není správně. Toto je příklad metriky pokrytí, nikoli metriky týkající se kvality produktu. d) Není správně. Toto je příklad metriky týkající se defektů, nikoli metriky týkající se kvality produktu. 	FL-5.3.1	K1	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
36	a	<p>a) Je správně. Klient se nachází na jiném místě a v jiném časovém pásmu, takže může být obtížné komunikovat osobně (tváří v tvář).</p> <p>b) Není správně. Dashboardy jsou obvykle dostupné kterémukoli uživateli kdykoli, takže rozdíl v časových pásmech není takovou překážkou v komunikaci jako verbální osobní komunikace.</p> <p>c) Není správně. Přestože je mezi Evropou a Amerikou několikahodinový časový posun, což může působit jisté komplikace, rozhodně to není tak velká překážka jako je pro osobní komunikaci.</p> <p>d) Není správně. Videokonferenční nástroje jsou vhodným komunikačním prostředkem. Ačkoli komunikace mezi Evropou a Amerikou v pracovní době obvykle vyžaduje, aby se jedna strana připojila ve velmi časných nebo velmi pozdních hodinách, není tak velkou překážkou jako je pro osobní komunikaci.</p>	FL-5.3.3	K2	1
37	a	<p>a) Je správně. V případě komplexní konfigurační položky (např. testovací prostředí) lze v rámci CM zaznamenávat dílčí položky, ze kterých se tato komplexní položka skládá, jejich vztahy a verze.</p> <p>b) Není správně. Nástroje CM neprovádějí testovací případy a nepočítají pokrytí.</p> <p>c) Není správně. Nástroj CM není nástroj pro management licencí.</p> <p>d) Není správně. CM nástroje negenerují testovací data.</p>	FL-5.4.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
38	b	<p>a) Není správně. I když je tato věta pravdivá, neposkytuje vývojářům velkou hodnotu.</p> <p>b) Je správně. Z výsledků testů se zdá, že systém ignoruje duplicity a řadí seznam bez ohledu na opakování, což je pravděpodobně příčina selhání v TC3, TC4, TC5. Tyto informace mohou vývojáři pomoci při nalezení a opravě defektu.</p> <p>c) Není správně. Systém nevykazuje selhání při řazení záporných čísel. Problém je spíše v tom, že se neberou v potaz duplicity.</p> <p>d) Není správně. Testovací případy TC3, TC4 a TC5 selžou, ale neexistuje žádná informace o tom, že by testovací případy měly nějaké defekty.</p>	FL-5.5.1	K3	1
39	c	<p>S ohledem na každou z uvedených kategorií nástrojů a jejich popisy platí, že:</p> <p>A. Nástroje pro statické testování – poskytují podporu při provádění revizí a statické analýzy (4).</p> <p>B. Nástroje podporující škálovatelnost a standardizaci nasazení, např. virtuální počítače, nástroje pro práci s kontejnery (3).</p> <p>C. Nástroje DevOps - podpora DevOps pipelines, sledování pracovních postupů, automatizovaný proces sestavování (build process), průběžná integrace/průběžné dodávání (CI/CD) (1).</p> <p>D. Nástroje pro spolupráci – usnadňují komunikaci (2).</p> <p>Tedy:</p> <p>a) Není správně.</p> <p>b) Není správně.</p> <p>c) Je správně. Správná kombinace je: 1C, 2D, 3B, 4A</p> <p>d) Není správně.</p>	FL-6.1.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Bodové hodnocení
40	a	<p>a) Je správně. Automatizace testování může poskytovat metriky, které jsou příliš komplikované pro manuální výpočet, např. míry pokrytí testů bílé skříňky (pro jakýkoliv komplexnější kód).</p> <p>b) Není správně. Při použití testovacích nástrojů odpovídá za testování tester a tato odpovědnost NENÍ sdílena s dodavatelem nástroje, protože dodavatel se na testování nepodílí. Jedinou možnou odpovědností, která by mohla být přiřazena dodavateli nástroje, je situace, kdy nástroj nefunguje podle očekávání a poskytuje nesprávné výsledky testů.</p> <p>c) Není správně. Testeři stále musí při analýze anomálií ve výsledcích testů aplikovat kritické myšlení, aby určili jejich pravděpodobnou příčinu.</p> <p>d) Není správně. Testeři ani nástroje nemohou generovat testovací případy jednoduše z analýzy programového kódu, protože kód je implementací a neposkytuje žádné informace o očekávaných výsledcích. Ty budou muset pocházet z jiné části testovací báze, jako je např. specifikace návrhu.</p>	FL-6.2.1	K1	1