

Vzorová zkouška – odpovědi

Sada C
Verze 1.6

Učební osnovy ISTQB® Učební osnovy pro základní stupeň Kompatibilní s učebními osnovami verze 4.0

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 mohou být kopírovány pro nekomerční použití, 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 vytvářející učební osnovy (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.

Historie změn

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

Verze	Datum	Poznámky
1.6	25.3.2025	Oprava termínů dle slovníku ISTQB
1.5	11.6.2024	Oprava odpovědi 20
1.4	29.5.2024	Oprava odpovědi 17
1.3	14.3.2024	Oprava odpovědí 20 a 25
1.2	20.12.2023	Sjednocení s verzí dokumentu s otázkami.
1.1	6.11.2023	Oprava kompatibility
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	15
11	16
12	17
13	18
14	19
15	20
16	21
17	22
18	23
19	24
20	25
21	26
22	26
23	27
24	27
25	28
26	29
27	30
28	31
29	31
30	32
31	32
32	33
33	33
34	34
35	34
36	35
37	35
38	36
39	36
40	37

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 počet bodů.
- Sady odpovědí, kdy u každé otázky je uvedeno:
 - správná odpověď,
 - odůvodnění každé odpovědi,
 - K-úroveň, studijní cíl a počet bodů.
- Sady odpovědí pro dodatečné otázky, kdy u každé otázky je uvedeno (neplatí pro všechny vzorové zkoušky):
 - správná odpověď,
 - odůvodnění každé odpovědi,
 - K-úroveň, studijní cíl a počet bodů.

Pozn.: otázky jsou obsaženy v samostatném dokumentu.

Klíč odpovědí

Číslo otázky	Správná odpověď	Studijní cíl (LO)	K-Úroveň	Počet bodů
1	b	FL-1.1.1	K1	1
2	c	FL-1.1.2	K2	1
3	b	FL-1.3.1	K2	1
4	b, e	FL-1.4.1	K2	1
5	a	FL-1.4.3	K2	1
6	c	FL-1.4.5	K2	1
7	b	FL-1.5.2	K1	1
8	a	FL-1.5.3	K2	1
9	d	FL-2.1.2	K1	1
10	d	FL-2.1.3	K1	1
11	b	FL-2.1.5	K2	1
12	c	FL-2.1.6	K2	1
13	d	FL-2.2.1	K2	1
14	b	FL-2.2.3	K2	1
15	d	FL-3.1.3	K2	1
16	a	FL-3.2.1	K1	1
17	b	FL-3.2.4	K2	1
18	b	FL-3.2.5	K1	1
19	c	FL-4.1.1	K2	1
20	a	FL-4.2.1	K3	1

Číslo otázky	Správná odpověď	Studijní cíl (LO)	K-Úroveň	Počet bodů
21	d	FL-4.2.2	K3	1
22	d	FL-4.2.3	K3	1
23	a	FL-4.2.4	K3	1
24	c	FL-4.3.2	K2	1
25	a	FL-4.3.3	K2	1
26	b	FL-4.4.1	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-5.1.1	K2	1
31	c	FL-5.1.4	K3	1
32	a	FL-5.1.5	K3	1
33	b	FL-5.1.6	K1	1
34	d	FL-5.1.7	K2	1
35	c	FL-5.2.3	K2	1
36	b	FL-5.3.2	K2	1
37	d	FL-5.4.1	K2	1
38	b	FL-5.5.1	K3	1
39	d	FL-6.1.1	K2	1
40	d	FL-6.2.1	K1	1

Odpovědi

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
1	b	<p>a) Není správně. Použití termínu validace pro kontrolu splnění zdokumentovaných požadavků je nesprávné, protože validace se zabývá naplněním potřeb a očekávání uživatelů, zatímco verifikace (ověření) se zabývá splněním specifikovaných požadavků. Správně by to tedy bylo, pokud bychom slovo „validace“ nahradili slovem „verifikace“ (ověření).</p> <p>b) Je správně. Vyvolávání selhání a identifikace defektů je nejčastějším cílem dynamického testování.</p> <p>c) Není správně. Inciování chyb a identifikace jejich kořenových příčin je nesprávně, protože testeři chyby neiniciují, snaží se vyvolat selhání. Chyby jsou obvykle způsobeny vývojáři (a opravdu je nelze iniciovat) a vedou k defektům, které se testeři snaží identifikovat buď přímo pomocí statického testování, nebo nepřímo prostřednictvím selhání při dynamickém testování. Identifikace kořenových příčin je užitečná, ale je součástí ladění, což není testovací činnost.</p> <p>d) Není správně. Verifikace (ověření), že testovaný objekt splňuje očekávání uživatelů, je nesprávné, protože ověření se zabývá kontrolou, zda jsou splněny specifikované (zdokumentované) požadavky. Splněním potřeb a očekávání uživatelů se naopak zabývá validace. Tvrzení by tedy bylo správné, pokud bychom nahradili pojem „ověření“ pojmem „validace“.</p>	FL-1.1.1	K1	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
2	c	<p>a) Není správně. Dynamické testování skutečně vyvolává selhání (na jejichž základě lze následně lokalizovat a opravit defekty). Ladění se však zabývá lokalizací defektů a jejich opravou. Ladění tedy neopravuje selhání.</p> <p>b) Není správně. Jak testování, tak ladění přispívají ke zlepšení kvality testovaného objektu, a proto by měly být obě tyto činnosti považovány za pozitivní. Ladění je obecně považováno za pozitivní činnost, protože se při ní něco opravuje. Dynamické testování zahrnuje záměrné vyvolání selhání testovaného objektu, což je důvod, proč jej někteří lidé považují za negativní aktivitu. To je ale velmi zúžený pohled (který testéři obvykle nezastávají). Jsou možné jak pozitivní, tak negativní testovací případy. Pozitivní testovací případy kontrolují, zda testovaný objekt správně provádí to, co má, zatímco negativní testování kontroluje, zda testovaný objekt nedělá to, co dělat nemá.</p> <p>c) Je správně. Testování zjišťuje existenci defektů buď přímo pozorováním defektu při revizích (nebo pomocí nástroje při statické analýze), nebo nepřímo vyvoláním selhání při dynamickém testování. Ladění je činnost odlišná od testování (obvykle prováděná vývojáři) a zabývá se vyhledáváním defektů (pouze u dynamického testování) a jejich odstraňováním.</p> <p>d) Není správně. Příčinou defektů jsou obvykle lidské chyby. Testování odhaluje defekty buď přímo prostřednictvím statického testování, nebo nepřímo tím, že vyvolává selhání při dynamickém testování. Ladění odstraňuje přímo defekty. Testování tedy nezjistí a ladění neodstraní příčinu defektů.</p>	FL-1.1.2	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
3	b	<p>Princip "nepřítomnost defektů je klam" se týká myšlenky, že zajištění správnosti v návaznosti na požadavky (tj. ověření absence defektů implementace) nezaručuje spokojenost uživatele se systémem. Je také nutno ověřit, že systém splňuje potřeby a očekávání uživatelů, plní byznysové cíle a překonává konkurenční systémy.</p> <p>a) Není správně. Princip "testování prokazuje přítomnost defektů, nikoli jejich nepřítomnost" ukazuje, že testování sice může odhalit existenci defektů v testovaném objektu, ale nedokazuje, že v něm žádné defekty neexistují, a nemůže tedy zaručit jeho správnost. Vysvětlení, že testování nemůže prokázat neexistenci defektů, by se tedy částečně odkazovalo na tento princip, nikoliv ale na princip "nepřítomnost defektů je klam".</p> <p>b) Je správně. Využitím koncových uživatelů při provádění akceptačních testů by mělo být možné ověřit, že systém splňuje potřeby a očekávání uživatelů.</p> <p>c) Není správně. Není možné zajistit, aby v dodaném systému nezůstaly žádné implementační defekty, protože princip "testování prokazuje přítomnost defektů, nikoli jejich nepřítomnost." vysvětluje, že testování sice může odhalit existenci defektů v testovaném objektu, ale není možné prokázat, že v něm žádné defekty nejsou, a tedy ani zaručit jeho správnost.</p> <p>d) Není správně. Jedním ze způsobů, jak řešit princip "testy se opotřebovávají", je úprava testů, které nezpůsobují žádné selhání tak, aby se nějaké defekty objevily. Tento princip se týká myšlenky, že opakování stejných testů na nezměněném kódu pravděpodobně neodhalí nové defekty, a proto může být důležité tyto testy upravit. Tím se ovšem nepotvrdí, že systém splňuje potřeby a očekávání uživatelů.</p>	FL-1.3.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
4	b, e	<p>Testovací analýza je popsána takto: Pro identifikaci funkcionalit, které vyžadují testování, je analyzována testovací báze, a výstupem této analýzy jsou prioritizované testovací podmínky spolu se souvisejícími riziky. Systematická identifikace testovacích podmínek jako položek pokrytí často zahrnuje použití technik testování jak během testovací analýzy, tak během činností návrhu testů. Z výše uvedeného popisu vyplývá, že techniky testování se často používají v činnostech testovací analýzy a návrhu testů. Technikami testování jsou analýza hraničních hodnot a rozdělení tříd ekvivalence.</p> <p>a) Není správně. V rámci implementace testování se pravděpodobně nebudou používat techniky testování, protože implementace testování se většinou zabývá sestavením testovacích případů do testovacích procedur, zatímco výstupem technik testování jsou testovací případy.</p> <p>b) Je správně. V rámci návrhu testů se pravděpodobně budou používat techniky testování s cílem vytvořit testovací případy z testovacích podmínek a položek pokrytí.</p> <p>c) Není správně. V rámci provedení testů se pravděpodobně nebudou používat techniky testování, protože provedení testování se většinou zabývá provedením testovacích procedur (a tím testovacích případů), zatímco výstupem technik testování jsou testovací případy.</p> <p>d) Není správně. V rámci monitorování testování se pravděpodobně nebudou používat techniky testování. Monitorování testování se většinou zabývá průběžnými kontrolami s cílem zajistit dodržování plánu, zatímco výstupem technik testování jsou testovací případy.</p> <p>e) Je správně. Testovací analýza používá testovací techniky k identifikaci testovacích podmínek.</p>	FL-1.4.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
5	a	<p>Výstupní testware testovacích činností je definován takto:</p> <ul style="list-style-type: none"> A. Testovací analýza - prioritizované testovací podmínky (4) (např. akceptační kritéria) a reporty o defektech v testovací bázi. B. Návrh testů - prioritizované testovací případy, testovací listiny, položky pokrytí (1), požadavky na testovací data a požadavky na testovací prostředí. C. Implementace testování - testovací procedury, automatizované testovací skripty, testovací sady, testovací data, harmonogram provádění testů (3) a prvky testovacího prostředí jako jsou stuby, ovladače, simulátory a virtualizace služeb. D. Dokončení testování - souhrnný report z testování, zdokumentovaná ponaučení (lessons learned), akce pro zlepšení a požadavky na změny (2) obvykle ve formě jako položek produktového backlogu. <p>Pak tedy:</p> <ul style="list-style-type: none"> a) Je správně. Správná kombinace je: 1B, 2D, 3C, 4A b) Není správně c) Není správně d) Není správně 	FL-1.4.3	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
6	c	<p>a) Není správně. Ačkoli je správné říci, že v agilním vývoji softwaru může některé úkoly spojené s rolí manažera testování vykonávat samotný agilní tým, není role testera primárně odpovědností jednoho člověka mimo tým. Místo toho je pravděpodobnější, že budou tuto roli zastávat různí členové týmu dle týmového přístupu.</p> <p>b) Není správně. Role manažera testování se primárně zaměřuje na činnosti související s plánováním testování, monitoringem testování, řízením testování a dokončením testování. Ačkoli je tedy toto tvrzení částečně správné, je nesprávné tvrdit, že role testera je primárně zodpovědná za monitorování a řízení testování.</p> <p>c) Je správně. V agilním vývoji jsou některé úkoly manažera testování vykonávány celým agilním týmem. U testovacích činností, které zasahují do více týmů v rámci organizace, mohou tyto úkoly vykonávat manažeři testování mimo vývojový tým.</p> <p>d) Není správně. Role manažera testování se primárně zaměřuje na činnosti související s plánováním testování, monitorováním testování, řízením testování a dokončením testování, zatímco role testera je primárně zodpovědná za technické a inženýrské aspekty testování, jako je testovací analýza, návrh testů, implementace testování a provedení testů. Role manažera testování tedy obvykle neodpovídá za testovací analýzu a návrh testů (ačkoli tvrzení, že role testera odpovídá především za implementaci a provedení testů je správné).</p>	FL-1.4.5	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
7	b	<p>a) Není správně. V týmovém přístupu hrají testéři důležitou roli tím, že sdílejí své odborné znalosti v oblasti testování s týmem a usměrňují vývoj produktu. Spolupracují s ostatními členy týmu na dosažení požadované úrovně kvality a také se zástupci byznysu na vytváření akceptačních testů. Testéři také spolupracují s vývojáři při určování optimální testovací strategie a přístupů k automatizaci testů.</p> <p>b) Je správně. Tím, že co nejefektivněji využívá rozmanité dovednosti každého člena týmu, týmový přístup podporuje týmovou dynamiku, podporuje komunikaci a spolupráci, a vytváří týmovou synergii, která je přínosem pro celý projekt.</p> <p>c) Není správně. Týmový přístup umožňuje každému členovi týmu s potřebnými dovednostmi a znalostmi převzít jakýkoli úkol, takže specializovaní členové týmu nejsou v tomto přístupu výhodou.</p> <p>d) Není správně. Neexistuje žádné konkrétní doporučení ohledně optimální velikosti týmů využívajících týmový přístup a nic nenasvědčuje tomu, že větší týmy jsou lepší.</p>	FL-1.5.2	K1	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
8	a	<p>a) Je správně. Hlavní výhodou nezávislosti testování je to, že testéři odhalí různé typy selhání a defektů mnohem pravděpodobněji než vývojáři, kteří mají jiné odborné zázemí, technický pohled a možné předsudky včetně kognitivního zkreslení. Hlavní nevýhodou nezávislosti testování je však to, že testéři mohou být izolováni od vývojového týmu, což vede k problémům v komunikaci, nízké úrovni spolupráce a potenciálně i k nepřátelskému vztahu, kdy jsou testéři obviňováni ze zpoždění a překážek při vydávání (release).</p> <p>b) Není správně. To, že vývojář zná kód, neznamená, že v něm zřídka najde defekty, naopak, tato znalost znamená, že může efektivně najít mnoho defektů ve svém vlastním kódu. Jako důvod, proč testéři a vývojáři nacházejí různé druhy defektů, se obvykle uvádí to, že vývojáři a testéři nemají stejné odborné zázemí.</p> <p>c) Není správně. Testování lze provádět v různých úrovních nezávislosti, od nulové nezávislosti autora až po velmi vysokou nezávislost testerů mimo organizaci. Ve většině projektů se využívá více úrovní nezávislosti, kdy vývojáři provádějí testování komponent a integrační testování komponent, testovací tým provádí systémové testování a systémové integrační testování a zástupci byznysu provádějí akceptační testování. Testéři tedy mohou být součástí vývojového týmu a nemusí pocházet z prostředí mimo své organizace. Znalost byznysového kontextu aplikace se bude lišit případ od případu a není navázaná na úroveň nezávislosti.</p> <p>d) Není správně. Testování lze provádět v různých úrovních nezávislosti, od nulové nezávislosti autora až po velmi vysokou nezávislost testerů mimo organizaci, přičemž testéři mimo tým vývojáře jsou obecně nezávislejší než testéři uvnitř týmu. Existuje ale předpoklad, že testéři mimo vývojový tým budou pravděpodobně více izolováni od vývojářů, a tak je pravděpodobnější, že budou více obviňováni ze zpoždění při vydávání produktu.</p>	FL-1.5.3	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
9	d	<p>a) Není správně. Ke každé vývojové činnosti existuje odpovídající testovací činnost, takže všechny vývojové činnosti podléhají řízení kvality. Nicméně zde je pokus o přiřazení úrovní testování k úrovním vývoje, které nemají (na rozdíl od přesné definice úrovní testování) obecně přijímanou definici.</p> <p>b) Není správně. Každá činnost při vývoji softwaru má odpovídající činnost testování, avšak cíle testování jsou zcela odlišné. Cílem testování může být například zajistit, aby testovaný objekt splňoval smluvní požadavek, že před dodáním musí být proveden určitý typ testování. V tomto případě není důvod, aby existoval odpovídající cíl vývoje.</p> <p>c) Není správně. Ke každé vývojové činnosti existuje odpovídající testovací činnost, takže všechny vývojové činnosti podléhají řízení kvality. Stejná symetrie však neplatí pro testovací činnosti a uživatelské činnosti. U některých systémů je například obtížné identifikovat koncové uživatele. Některé činnosti testování jsou také zaměřeny na vývojáře (např. testování vysoké míry udržitelnosti), k čemuž neexistuje žádný uživatelský aspekt.</p> <p>d) Je správně. Ke každé vývojové činnosti existuje odpovídající testovací činnost, takže všechny vývojové činnosti podléhají řízení kvality.</p>	FL-2.1.2	K1	1
10	d	<p>a) Není správně. Vývoj řízený testy komponent není správným příkladem přístupu k vývoji iniciovaného testy (test-first).</p> <p>b) Není správně. Vývoj řízený integračním testováním není správným příkladem přístupu k vývoji iniciovaného testy.</p> <p>c) Není správně. Vývoj řízený systémovými testy není správným příkladem přístupu k vývoji iniciovaného testy.</p> <p>d) Je správně. Vývoj řízený akceptačními testy (ATDD) je známým příkladem přístupu k vývoji iniciovaného testy.</p>	FL-2.1.3	K1	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
11	b	<p>a) Není správně. Praktiky spojené s principem shift left jsou zaměřeny na zavedení většího počtu testovacích činností v raných fázích životního cyklu vývoje softwaru (SDLC), přičemž tento cyklus je zobrazován jako plynoucí zleva doprava. Neexistuje ale nic jako levá strana testovacího procesu.</p> <p>b) Je správně. Princip shift left zdůrazňuje důležitost zahájení testování na začátku SDLC. Jeho zavedení vyžaduje dodatečné školení a navýšenou pracnost a náklady v počátečních fázích SDLC, nicméně v souhrnu by mělo dojít k úspoře nákladů.</p> <p>c) Není správně. Ačkoli jsou automatizované testy komponent a integrační testy komponent pro regresní testování obecně cenné, za jejich vytvoření jsou obvykle odpovědní vývojáři. Pokud je využívána průběžná integrace a průběžné dodávání (CI/CD), pak jsou tyto testy odevzdávány spolu s kódem. V některých situacích může tester automatizovat regresní testy a někdy i v úrovni testů komponent nebo v úrovni integračního testování komponent, což však není součástí principu shift left (jeho podstatou je přesun testování do dřívějších fází SDLC).</p> <p>d) Není správně. Školení testerů k vykonávání úkolů v raných fázích SDLC by podpořilo princip shift left, protože zdůrazňuje důležitost zahájení testování již v počátečních fázích SDLC. Automatizace dalších testovacích činností, které se mají provádět později v SDLC, však není součástí principu shift left.</p>	FL-2.1.5	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
12	c	<p>a) Není správně. Jedním z cílů retrospektiv je identifikovat potenciální zlepšení procesů, která – pokud jsou uvedena do praxe – by měla vést ke zvýšení kvality budoucích výstupů procesu vývoje. Je tedy pravděpodobné, že k tomu dojde v důsledku retrospektivy.</p> <p>b) Není správně. Přínosem retrospektivy pro testování je zvýšení efektivity testování prostřednictvím zlepšení procesů. Je tedy pravděpodobné, že k tomu dojde v důsledku retrospektivy.</p> <p>c) Je správně. Retrospektivy se obvykle účastní testéři, vývojáři, architekti, vlastníci produktů a byznysové analytici. Koncoví uživatelé jsou ale na tyto schůzky zváni (nebo se jich účastní) jen zřídka. Je tedy nepravděpodobné, že by z těchto schůzek dostali nějaké reporty. Stejně tak je velmi nepravděpodobné, že by se prostřednictvím retrospektivy dozvěděli a pochopili více o procesech vývoje a testování.</p> <p>d) Není správně. Přínosem retrospektivy pro testování je zlepšení kvality testwaru (včetně automatizovaných testovacích skriptů) díky společným revizím s vývojáři. Je tedy pravděpodobné, že k tomu dojde v důsledku retrospektivy.</p>	FL-2.1.6	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
13	d	<p>a) Není správně. Testování komponent (nazývané také testování jednotek) testuje jednotlivé komponenty izolovaně a většinou se jedná o ověřování vůči specifikaci, nikoli o validaci vůči potřebám uživatele. Toto testování však obvykle neprovádějí testeři, ale vývojáři, a to ve svém vývojovém prostředí.</p> <p>b) Není správně. Integrační testování komponent spočívá v testování rozhraní a interakcí mezi komponentami a většinou se jedná o ověřování vůči specifikaci, nikoli o validaci vůči potřebám uživatele. Toto testování také obvykle neprovádějí testeři, ale vývojáři.</p> <p>c) Není správně. Systémové integrační testování zkoumá rozhraní s jinými systémy a externími službami a jedná se převážně o ověřování vůči specifikaci, nikoliv o validaci vůči potřebám uživatele. Tento typ testování obvykle provádějí testeři.</p> <p>d) Je správně. Akceptační testování se zaměřuje na validaci, zda systém splňuje byznysové potřeby uživatele a je připraven k nasazení. V ideálním případě toto testování provádějí koncoví uživatelé.</p>	FL-2.2.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
14	b	<p>a) Není správně. Konfirmační testování s cílem ověřit, že provedené aktualizace vedly ke správné implementaci, je nezbytné. Poté však dává smysl provést také regresní testování, aby se zajistilo, že v nezměněných částech systému nevznikly nebo se neprojevily žádné další defekty.</p> <p>b) Je správně. Konfirmační testování prověří, že provedené aktualizace vedly ke správné implementaci. Regresní testování je použito k zajištění toho, že v nezměněných částech systému nevznikly nebo se neprojevily žádné další defekty.</p> <p>c) Není správně. Regresní testování by mělo být použito k zajištění toho, že při aktualizaci nevznikly nebo se neprojevily žádné defekty v nezměněných oblastech systému. Je však také nutné provést konfirmační testování, které ověří, zda aktualizace vedly ke správné implementaci.</p> <p>d) Není správně. Konfirmační testování prověří, že provedené aktualizace vedly ke správné implementaci. Regresní testování je použito k zajištění toho, že v nezměněných částech systému nevznikly nebo se neprojevily žádné další defekty. Pokud se však provádí (tj. pokud je potřeba otestovat aktualizaci), konfirmační testování předchází regresnímu testování.</p>	FL-2.2.3	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
15	d	<p>Pro uvedené příklady defektů platí, že:</p> <ul style="list-style-type: none"> i. Kvůli složitosti návrhu si dvě různé části specifikace návrhu protiřečí. Jedná se o příklad defektu ve specifikaci, která může obsahovat nesrovnalosti, nejednoznačnosti, rozpory, opomenutí, nepřesnosti a duplicity. Tyto defekty lze nejlépe odhalit pomocí statického testování. ii. Doba odezvy je příliš dlouhá, takže uživatelé ztrácejí trpělivost. To je příklad defektu v době odezvy, kterou lze v praxi zjistit pouze spuštěním programu a změřením doby odezvy, což lze nejlépe zjistit dynamickým testováním. iii. Určitá cesta v kódu je během provádění nedosažitelná (nelze ji spustit). Jedná se o příklad defektu programování (například proměnná s nedefinovanými hodnotami, nedeklarované proměnné, duplicitní nebo nedosažitelný kód a přílišná složitost kódu), který lze nejlépe zjistit statickým testováním. iv. Proměnná je deklarována, ale v programu není nikdy použita. Jedná se o příklad defektu programování (například proměnná s nedefinovanými hodnotami, nedeklarované proměnné, duplicitní nebo nedosažitelný kód a přílišná složitost kódu), který lze nejlépe zjistit statickým testováním. v. Program potřebuje k vygenerování reportu příliš velké množství paměti. To je příklad defektu výkonnosti, kterou lze v praxi zjistit pouze spuštěním programu a měřením spotřebované paměti, což lze nejlépe zjistit dynamickým testováním. <p>Pak tedy:</p> <ul style="list-style-type: none"> a) Není správně b) Není správně c) Není správně d) Je správně. Správná kombinace je tedy i, iii a iv. 	FL-3.1.3	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
16	a	<p>a) Je správně. Získávání zpětné vazby od zainteresovaných stran v rané fázi procesu vývoje softwaru může být velmi přínosné. Usnadňuje včasnou komunikaci o možných problémech s kvalitou, může zabránit nedorozuměním ohledně požadavků a zajišťuje, že případné změny v požadavcích zainteresovaných stran budou pochopeny a implementovány rychleji (dříve).</p> <p>b) Není správně. Zpětná vazba pochází od zainteresovaných stran a je tedy nepravděpodobné, že by poskytování zpětné vazby zlepšilo to, jak rozumí svým vlastním uživatelským požadavkům.</p> <p>c) Není správně. Získávání zpětné vazby od zainteresovaných stran v rané fázi procesu vývoje softwaru může být velmi přínosné. Usnadňuje včasnou komunikaci o možných problémech s kvalitou, může zabránit nedorozuměním ohledně požadavků a zajišťuje, že případné změny v požadavcích zainteresovaných stran budou pochopeny a implementovány rychleji (dříve). To, že změny v požadavcích lze pochopit a implementovat dříve, však neznamena, že jsou změny požadavků umožněny neomezeně.</p> <p>d) Není správně. Zpětná vazba je od zainteresovaných stran a nezahrnuje komunikaci s nimi. Komunikace směrem ke koncovým uživatelům může znamenat také informování o tom, které požadavky nebudou před vydáním implementovány (ideálně by ovšem k takové situaci vůbec nemělo dojít).</p>	FL-3.2.1	K1	1

17	b	<p>Platí, že:</p> <ol style="list-style-type: none"> 1. Technická revize - tento typ revize provádějí technicky kvalifikovaní revidující a vede ji moderátor. Cílem je dosáhnout dohody a rozhodnout o technických problémech a zároveň hodnotit kvalitu a budovat důvěru v pracovní produkt, generovat nové nápady, odhalovat anomálie a motivovat autory a umožnit jim zlepšovat se. 2. Neformální revize - hlavním cílem je odhalit anomálie. Proces není definován a nevyžaduje se formální dokumentovaný výstup. 3. Inspekce - jedná se o nejformálnější typ revize, který se řídí celým obecným procesem revize. Primárním cílem je najít co nejvíce anomálií, nicméně mezi další cíle patří hodnotit kvalitu a budovat důvěru v pracovní produkt, generovat nové nápady, odhalovat anomálie a motivovat autory a umožnit jim zlepšovat se včetně shromažďování metrik, které lze využít ke zlepšení životního cyklu vývoje softwaru (SDLC), včetně samotného procesu inspekce. Při inspekcích nemůže autor vystupovat jako vedoucí revize nebo zapisovatel. 4. Předvedení (walkthrough), které je vedené autorem, může sloužit mnoha cílům jako je hodnocení kvality a budování důvěry v pracovní produkt, vzdělávání revidujících, dosažení dohody, generování nových nápadů, motivace a podpora pro autory s cílem zlepšovat pracovní produkt a odhalovat anomálie. Revidující mohou (ale nemusí) před předvedením provést individuální revizi. <ol style="list-style-type: none"> A. Zahrnuje cíle jako je zajištění dohody, generování nových nápadů a motivování autorů ke zlepšení. B. Zahrnuje cíle jako je vzdělávání revidujících, dosažení dohody, generování nových nápadů a odhalování potenciálních anomálií. C. Hlavním cílem je odhalování anomálií, mimo jiné je vyžadováno zavedení sady metrik s cílem podpořit zlepšování procesů. 	FL-3.2.4	K2	1
----	---	---	----------	----	---

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
		<p>D. Hlavním cílem je odhalování anomálií a není vytvářen žádný formální dokumentovaný výstup.</p> <p>Pak tedy:</p> <ul style="list-style-type: none"> a) Není správně b) Je správně. c) Není správně d) Není správně 			
18	b	<ul style="list-style-type: none"> a) Není správně. Pro zajištění úspěšné revize je důležité si zajistit podporu managementu, což však neznamená, že by se management měl účastnit procesu revize jako revidující. b) Je správně. Pro zajištění úspěšné revize je důležité rozdělit pracovní produkt na části, které jsou dostatečně malé na to, aby mohly být revidovány v přiměřeném časovém horizontu. Zamezí se tak ztrátě pozornosti revidujících během jednotlivých revizí nebo revizních schůzek. c) Není správně. Pro zajištění úspěšné revize je důležité jasně definovat cíle a měřitelná výstupní kritéria a rozhodně nehodnotit její účastníky. d) Není správně. Pro zajištění úspěšné revize je důležité rozdělit revizi na menší části tak, aby nedošlo ke ztrátě pozornosti revidujících během jednotlivých revizí nebo revizních schůzek. Není tedy vhodné plánovat pokrytí jednoho dokumentu v rámci jedné revize. 	FL-3.2.5	K1	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
19	c	<p>a) Není správně. Ve většině případů lze pro stejné testované objekty použít jak techniky testování černé skříňky, tak techniky testování založené na zkušenostech.</p> <p>b) Není správně. Techniky testování černé skříňky i techniky testování založené na zkušenostech lze použít ve všech úrovních testování.</p> <p>c) Je správně. Techniky testování černé skříňky (známé též jako techniky založené na specifikaci) jsou založeny na analýze specifikovaného chování testovaného objektu bez informací o jeho vnitřní struktuře. Testovací bází je tedy obvykle specifikace. Techniky testování založené na zkušenostech využívají znalosti a zkušenosti testerů pro návrh a implementaci testovacích případů. To znamená, že tester při navrhování testů nemusí specifikaci vůbec použít.</p> <p>d) Není správně. Techniky testování založené na zkušenostech mohou odhalit defekty, které techniky testování černé (i bílé) skříňky neodhalí. Proto jsou techniky testování založené na zkušenostech doplňkem k technikám testování černé skříňky a k technikám testování bílé skříňky. Techniky testování černé skříňky i techniky testování založené na zkušenostech lze použít ve všech SDLC.</p>	FL-4.1.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
20	a	<p>a) Je správně.</p> <ul style="list-style-type: none"> Hodnota "12" pokrývá "nesprávná délka - málo číslic". Hodnota "1111" pokrývá "správná délka" a "nesprávný počet odlišných číslic". Hodnota "1234" pokrývá "správná délka" a "správný počet odlišných číslic". Hodnota "12345" pokrývá "nesprávná délka - příliš mnoho číslic". <p>b) Není správně. Všechny třídy jsou pokryty, ale u třídy "nesprávná délka" je pokryta pouze dolní část (tedy málo číslic).</p> <p>c) Není správně. Neexistují žádné hodnoty pokrývající "správný PIN".</p> <p>d) Není správně. Neexistuje žádná hodnota pokrývající "počet odlišných číslic".</p>	FL-4.2.1	K3	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
21	d	<p>Třídy ekvivalence jsou: {..., 99, 100}, {101, 102, ..., 198, 199}, {200, 201, ...}.</p> <p>Existují tedy 4 hraniční hodnoty, kterými jsou: 100, 101, 199 a 200.</p> <p>V případě 2-hodnotové varianty BVA existují pro každou hraniční hodnotu dvě položky pokrytí (daná hraniční hodnota a její nejbližší soused patřící do vedlejší třídy). Protože nejbližší sousedé jsou zároveň hraničními hodnotami v sousední třídě, pak existují pouze čtyři položky pokrytí.</p> <p>Pak tedy:</p> <ul style="list-style-type: none"> a) Není správně. Pouze 100 a 200 jsou platné položky pokrytí pro 2-hodnotové BVA, takže je dosaženo 50 % pokrytí. b) Není správně. Pouze 100 a 200 jsou platné položky pokrytí pro 2-hodnotové BVA, takže je dosaženo 50 % pokrytí. c) Není správně. Pouze 100 a 101 jsou platné položky pokrytí pro 2-hodnotové BVA, takže je dosaženo 50 % pokrytí. d) Je správně. Hodnoty 101, 199 a 200 jsou platné položky pokrytí pro 2-hodnotové BVA, takže je dosaženo 75 % pokrytí. 	FL-4.2.2	K3	1
22	d	<ul style="list-style-type: none"> a) Není správně. Kombinace (T, T, F) neodpovídá žádnému pravidlu. Jedná se o příklad opomenutí, nikoliv rozporu. b) Není správně. Kombinace (T, F, T) odpovídá pouze jednomu sloupci R2, takže zde neexistuje žádný rozpor. c) Není správně. Obě kombinace (T, T, T) a (F, T, T) odpovídají pouze jednomu sloupci R1, takže zde neexistuje žádný rozpor. d) Je správně. Kombinace (F, F, F) odpovídá R2 i R3, ale R2 a R3 mají různé akce. Tím existuje mezi R2 a R3 rozpor. 	FL-4.2.3	K3	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
23	a	<p>Následující tři přechody: "ŽÁDOST -> POTVRZENO" "VE FRONTĚ -> POTVRZENO" "VE FRONTĚ -> KONEC"</p> <p>se nemohou vyskytovat ve stejném testovacím případě, což naznačuje, že jsou zapotřebí alespoň tři testovací případy. Všechny ostatní přechody se mohou objevit v kombinaci s jedním nebo více z těchto tří přechodů, takže potřebujeme minimálně tři testovací případy. Ve skutečnosti jsou možné pouze tři sekvence:</p> <p>TC1: START (Žádost o pokoj) → ŽÁDOST (Je k dispozici) → POTVRZENO (Zaplatit) → KONEC TC2: START (Žádost o pokoj) → ŽÁDOST (Není k dispozici) → VE FRONTĚ (Je k dispozici) → POTVRZENO (Zaplatit) → KONEC TC3: START (Žádost o pokoj) → ŽÁDOST (Není k dispozici) → VE FRONTĚ (Zrušit) → KONEC</p> <p>Pak tedy:</p> <ul style="list-style-type: none"> a) Je správně. b) Není správně. c) Není správně. d) Není správně. 	FL-4.2.4	K3	1
24	c	<p>Při testování větví jsou položkami pokrytí větve, které jsou reprezentovány hranami (spojnicemi) v grafu řídicího toku. V něm je 8 hran.</p> <p>Pak tedy:</p> <ul style="list-style-type: none"> a) Není správně. b) Není správně. c) Je správně. d) Není správně. 	FL-4.3.2	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
25	a	<p>a) Je správně. Pouhé testování pomocí technik černé skříňky neposkytuje informace o míře skutečného pokrytí kódu. Naopak výsledky testování s využitím technik bílé skříňky poskytují objektivní měření pokrytí, poskytují také nezbytné informace umožňující vytváření dalších testů s cílem zvýšení tohoto pokrytí a následného zvýšení důvěry v kód.</p> <p>b) Není správně. Toto tvrzení je správné, ale nemá nic společného s testováním černé skříňky.</p> <p>c) Není správně. Mezi technikami testování bílé skříňky a technikami černé skříňky obecně neexistuje žádný vztah.</p> <p>d) Není správně. Testovací techniky bílé skříňky se používají k návrhu testů na základě samotného testovaného objektu, zatímco testovací techniky černé skříňky se používají k návrhu testů na základě specifikace. Proto neexistuje žádný vztah mezi položkami pokrytí odvozenými z těchto dvou typů testovacích technik.</p>	FL-4.3.3	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
26	b	<p>a) Není správně. Průzkumné testování používá testovací listiny, nikoli seznam možných defektů/selhání. Ačkoli může průzkumné testování využívat i jiné testovací techniky, v tomto případě je nejpravděpodobnější možností technika útoku na chyby.</p> <p>b) Je správně. Toto je seznam možných selhání. Útoky na chyby jsou metodickým přístupem jak provést odhadování chyb a vyžadují, aby tester vytvořil nebo získal seznam možných chyb, defektů a selhání. Na základě tohoto seznamu navrhne testy, které identifikují defekty spojené s těmito chybami, odhalí je nebo způsobí selhání.</p> <p>c) Není správně. Tester používá kontrolní seznam položek, který mu pomáhá při testování. Jak odhadování chyb, tak testování na základě kontrolních seznamů takové seznamy používají, nicméně zde se jedná o seznam možných selhání, nikoliv o testovací podmínky, a proto je NEJPRAVDĚPODOBNĚJŠÍ testovací technikou útok na chyby, protože se zaměřuje na chyby, defekty a selhání.</p> <p>d) Není správně. Technika analýzy hraničních hodnot (BVA) je založena na analýze hraničních hodnot tříd ekvivalence. Výše uvedený seznam nezmiňuje třídy ekvivalence ani jejich hranice.</p>	FL-4.4.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
27	d	<p>a) Není správně. Je sice pravda, že tester může na základě kontrolního seznamu vytvářet a provádět detailní (tedy ne obecné) testovací případy, nevysvětluje to ale, jak by to mělo vést ke zvýšení pokrytí.</p> <p>b) Není správně. Položky kontrolního seznamu by neměly být automatizovány. I kdyby ale byly, automatizované testovací skripty provádějí testy vždy stejným způsobem, což obvykle nevede ke zvýšení pokrytí.</p> <p>c) Není správně. Je pravda, že každá položka kontrolního seznamu by měla být testována samostatně a nezávisle. To však ovlivňuje pořadí provedení testů bez vlivu na dosažené pokrytí, takže nemůže vést ke zvýšení pokrytí.</p> <p>d) Je správně. Jelikož se jedná o obecné kontrolní seznamy (high-level), je pravděpodobné, že se v daném testování může objevit určitá variabilita, což může přispět k většímu pokrytí, ale i k menší míře opakovatelnosti testování. Pokud dva testéři postupují podle kontrolního seznamu na obecné úrovni, každý z nich může použít jiná testovací data, jiné testovací kroky apod. Tímto způsobem první tester pravděpodobně otestuje oblasti, které druhý tester ne, což povede k většímu pokrytí.</p>	FL-4.4.3	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
28	b	<p>a) Není správně. Toto akceptační kritérium popisuje, jaká pravidla nebo předpisy musí systém dodržovat (v tomto případě právo být zapomenut). Jedná se tedy o příklad kritéria orientovaného na pravidla.</p> <p>b) Je správně. Toto akceptační kritérium popisuje příklad scénáře, který musí být v systému realizován. Jedná se tedy o je příklad kritéria orientovaného na scénář.</p> <p>c) Není správně. Tato věta vypadá spíše jako řádek kódu, který implementuje nějaké byznysové pravidlo. Akceptační kritéria by měla být sepsána ve spolupráci se zástupci byznysu, a proto by měla být napsána v jazyce, kterému rozumí. Tato věta bude pro ně pravděpodobně nesrozumitelná.</p> <p>d) Není správně. Toto akceptační kritérium popisuje, s jakými pravidly nebo předpisy musí být systém v souladu a jak bude tento soulad zajištěn. Jedná se tedy o příklad kritéria orientovaného na pravidla, nikoliv kritéria založeného na scénáři.</p>	FL-4.5.2	K2	1
29	d	<p>a) Není správně. Chceme zkontrolovat, zda mají speciální uživatelé práva běžných uživatelů, takže potřebujeme otestovat přístupová práva pro speciálního uživatele, nikoli pro běžného uživatele.</p> <p>b) Není správně. Chceme zkontrolovat, zda mají speciální uživatelé práva běžných uživatelů, takže potřebujeme otestovat přístupová práva pro speciálního uživatele, nikoli pro běžného uživatele.</p> <p>c) Není správně. V akceptačních kritériích není popsáno žádné podlaží 5. Testovací případy by neměly rozšiřovat rozsah uživatelského scénáře. I kdybychom chtěli provést negativní testování, nebude tento test přímo souviset s AC3.</p> <p>d) Je správně. Tímto způsobem můžeme zkontrolovat, zda má speciální uživatel přístup k podlažím, která jsou přístupná běžnému uživateli.</p>	FL-4.5.3	K3	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
30	a	<p>a) Je správně. Plán testování může obsahovat <i>požadavky na testovací data</i> (jako součást přístupu k testování), ale nikoliv konkrétní testovací data využitelná v testovacích případech. Testovací data jsou součástí testovacích případů, nikoliv plánu testů. Při tvorbě plánu testů také obvykle není možné tato data definovat, protože není přesně známo, jak budou komponenty vypadat.</p> <p>b) Není správně. Jedním z účelů plánu testování je pomoci zajistit, aby testovací činnosti splňovaly stanovená kritéria, a to zahrnutím vstupních a výstupních kritérií. Kritéria pokrytí kódu jsou příkladem takových kritérií v úrovni testování komponent.</p> <p>c) Není správně. Typickým obsahem plánu testů jsou šablony dokumentace. To pomáhá usnadnit komunikaci mezi zainteresovanými stranami definováním standardního způsobu komunikace nebo reportování.</p> <p>d) Není správně. Jedním z účelů plánu testování je prokázat, že testování je v souladu se stávající politikou a strategií testování, nebo vysvětlit, proč se od nich testování odchyluje. Tato možnost je příkladem vysvětlení takové odchylky a týká se úrovně testování, které budou (nebo nebudou) použity.</p>	FL-5.1.1	K2	1
31	c	<p>Z grafu vyplývá: $A(4)=6$ a $A(3)=8$ (poslední dvě šedá pole). Ze vzorce získáme: $E(5) = (3 \cdot A(4) + A(3)) / 4 = (3 \cdot 6 + 8) / 4 = 26 / 4 = 6,5$ člověkodní.</p> <p>Pak tedy:</p> <p>a) Není správně. b) Není správně. c) Je správně. d) Není správně.</p>	FL-5.1.4	K3	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
32	a	<p>Chceme sice provést testovací případy podle jejich priorit, ale musíme také zohlednit závislosti.</p> <p>Pokud bereme v úvahu pouze priority, chceme nejprve spustit TC 5 a TC 7 (nejvyšší priorita), poté TC 1, TC 3 a TC 4 a nakonec TC 2 a TC 6 (nejnižší priorita).</p> <p>Abychom však mohli spustit TC 7, musíme nejprve spustit TC 4.</p> <p>Abychom mohli spustit TC 5, musíme spustit TC 4 a TC 2, ale TC 2 je blokován TC 1, který by měl být spuštěn před TC 2.</p> <p>Aby bylo možné spustit testovací případy s prioritou 1 co nejdříve, mělo by být pořadí prvních pět testovacích případů následující: TC 4 - TC 7 - TC 1 - TC 2 - TC 5.</p> <p>Dále musíme spustit TC 3, protože má vyšší prioritu než TC 6.</p> <p>Úplný harmonogram tedy bude vypadat následovně: TC 4 - TC 7 - TC 1 - TC 2 - TC 5 - TC 3 - TC 6.</p> <p>Šestým testovacím případem tedy bude TC 3.</p> <p>Pak tedy:</p> <ul style="list-style-type: none"> a) Je správně. b) Není správně. c) Není správně. d) Není správně. 	FL-5.1.5	K3	1
33	b	<ul style="list-style-type: none"> a) Není správně. Testovací pyramida neobsahuje informace o prioritách testů. b) Je správně. Testovací pyramida ukazuje, že různé testy mají různé úrovně granularity. c) Není správně. Testovací pyramida je nezávislá na kritériích pokrytí. d) Není správně. Testovací pyramida neznázorňuje žádné vztahy mezi jednotlivými testy. 	FL-5.1.6	K1	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
34	d	<p>a) Není správně. Testovací kvadranty seskupují úrovně testování a typy testů odděleně podle několika kritérií. Nepředstavují žádné kombinace úrovní testování a typů testů a nesouvisí s žádným umístěním v rámci životního cyklu vývoje softwaru. Jak typy testů, tak testovací úrovně jsou v testovacích kvadrantech řešeny odděleně.</p> <p>b) Není správně. Testovací kvadranty seskupují testovací úrovně a typy testů podle několika kritérií. Nepopisují míru granularity jednotlivých typů testů prováděných v jednotlivých úrovních testování (tuto informaci obsahuje testovací pyramida).</p> <p>c) Není správně. Toto tvrzení je chybné, protože obecně lze provést jakýkoli typ testu v jakékoli úrovni testování.</p> <p>d) Je správně. Kvadranty testování (používané zejména v agilním vývoji softwaru) seskupují úrovně testování, typy testů, činnosti, techniky testování a pracovní produkty. V tomto modelu mohou být testy zaměřené na byznys nebo na technologii a mohou také podporovat tým (tj. usměrňovat vývojové aktivity) nebo kritizovat produkt (tj. měřit jeho chování proti očekáváním). Kombinací těchto dvou úhlů pohledu je určen každý ze čtyř kvadrantů.</p>	FL-5.1.7	K2	1
35	c	<p>a) Není správně. Monitoring rizik je součástí řízení rizik, nikoli analýzy rizik.</p> <p>b) Není správně. Samotná identifikace rizik nám neumožňuje provádět aktivity na jejich zmírnění. Zmírňující opatření jsou definována ve fázi řízení rizik.</p> <p>c) Je správně. Toto je příklad toho, jak analýza rizik ovlivňuje důkladnost a rozsah testování.</p> <p>d) Není správně. Položky pokrytí jsou odvozeny pomocí testovacích technik, nikoli pomocí analýzy rizik.</p>	FL-5.2.3	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
36	b	<p>a) Není správně. Reporty o postupu prací při testování se většinou používají při monitoringu, řízení a dokončení testování, nikoliv při návrhu testů.</p> <p>b) Je správně. Souhrnný report z testování se připravuje v průběhu fáze dokončení testování, kdy je projekt, úroveň testování nebo typ testů ukončen, a kdy jsou (v ideálním případě) splněna stanovená výstupní kritéria. Tento report využívá informace z dílčích reportů o postupu prací při testování, ale i jiné zdroje dat.</p> <p>c) Není správně. Reporty o průběhu testování se většinou používají při monitorování a kontrole testů a při dokončení testování, nikoli při testovací analýze.</p> <p>d) Není správně. Zprávy o průběhu testování se většinou používají při monitorování testování, řízení testování a při dokončení testování, nikoli při plánování testování.</p>	FL-5.3.2	K2	1
37	d	<p>a) Není správně. Když uživatel nahlásí selhání v softwaru, díky jedinečné identifikaci commitů je možné znovu sestavit soubory z verze softwaru, kterou uživatel používal (stejně jako související verze testovacích skriptů), a tím reprodukovat selhání a rychleji nalézt defekt.</p> <p>b) Není správně. Pokud změna v testovacím prostředí způsobí během testování neočekávané problémy, konfigurační management umožňuje testerům vrátit se k předchozí verzi prostředí. Tím je zajištěno, že testování může pokračovat, aniž by bylo ovlivněno touto změnou.</p> <p>c) Není správně. Konfigurační management zajišťuje, že veškerá identifikovaná dokumentace (např. specifikace požadavků) a softwarové položky jsou jednoznačně uvedeny v testovací dokumentaci (např. v testovacích plánech).</p> <p>d) Je správně. Toto je zajištěno v rámci procesu managementu defektů, nikoliv konfiguračního managementu.</p>	FL-5.4.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
38	b	<p>a) Není správně. Tyto údaje jsou důležité, ale ne tak důležité jako prvky testovacího prostředí.</p> <p>b) Je správně. Důležité je, že chybí identifikace prohlížeče a zařízení použitého při testování. Tyto informace jsou důležité, protože defekt může být specifický pro určitý prohlížeč nebo určité zařízení. Například přihlašovací tlačítko může v jednom prohlížeči (nebo v jedné verzi určitého prohlížeče) fungovat správně, ale v jiném ne. Informace o prohlížeči a zařízení proto mohou vývojářům pomoci reprodukovat problém a rychleji najít jeho kořenovou příčinu.</p> <p>c) Není správně. Testovaný objekt je identifikován (WebShop v0.99)</p> <p>d) Není správně. Dopad je v reportu zahrnut ve formě závažnosti (vysoká).</p>	FL-5.5.1	K3	1
39	d	<p>a) Není správně. Nástroje pro provedení testů a měření pokrytí usnadňují automatizované provedení testovacích případů a měření pokrytí dosaženého spuštěním těchto testovacích případů. Tyto nástroje však nepomáhají se správou defektů a konfiguračním managementem.</p> <p>b) Není správně. Nástroje pro návrh a implementaci testů usnadňují vytváření testovacích případů, testovacích dat a testovacích procedur, ale nepomáhají se správou defektů a konfiguračním managementem.</p> <p>c) Není správně. Nástroje pro management defektů se používají k managementu defektů, ale nejsou to nástroje pro testování a nepoužívají se ke správě testovacích případů nebo ke konfiguračnímu managementu.</p> <p>d) Je správně. Nástroje pro management testování zvyšují efektivitu procesu testování tím, že usnadňují management životního cyklu vývoje softwaru (SDLC), požadavků, testů, defektů a konfigurační management.</p>	FL-6.1.1	K2	1

Číslo otázky	Správná odpověď	Vysvětlení / zdůvodnění	Studijní cíl (LO)	K-Úroveň	Počet bodů
40	d	<p>a) Není správně. Generování testovacích případů bez přístupu k testovací bázi není možné. Generování testovacích případů testery nebo nástroji vyžaduje přístup k testovací bázi.</p> <p>b) Není správně. Dosažení většího pokrytí díky objektivnějšímu hodnocení není přímým přínosem automatizace testování. Automatizace testování umožní objektivnější posouzení pokrytí, avšak toto objektivní posouzení nezvýší pokrytí (pouze použitím výsledků pokrytí k psaní dalších testovacích případů lze zvýšit pokrytí).</p> <p>c) Není správně. "Prodloužení doby provádění testů, které je k dispozici s vyšším výpočetním výkonem" je protichůdné tvrzení, protože vyšší výpočetní výkon by za normálních okolností zkrátil dobu provádění testů, a prodloužení doby provádění testů není přínosem, protože testování by trvalo déle.</p> <p>d) Je správně. Výhodou automatizace testování je prevence lidských chyb díky větší konzistenci a opakovatelnosti, protože automatizace testování není náchylná k lidským chybám. Znamená to například, že testy jsou důsledně odvozovány z požadavků, testovací data jsou vytvářena systematicky a testy jsou nástrojem prováděny ve stejném pořadí a se stejnou frekvencí.</p>	FL-6.2.1	K1	1