

Egzamin przykładowy – odpowiedzi

Zbiór C

wersja 1.4

(wersja PL 1.0.0.2)

Sylabus ISTQB® Certyfikowany tester – poziom podstawowy

zgodny z sylabusem w wersji 4.0

International Software Testing Qualifications Board



Spis treści

Egzamin przykładowy – odpowiedzi	1
Informacja o prawach autorskich	4
Podziękowania	5
Historia zmian	6
Historia zmian polskiej wersji	6
1. Wstęp	7
Cel dokumentu	7
Instrukcje	7
2. Klucz odpowiedzi	8
1	9
2	10
3	11
4	13
5	15
6	16
7	17
8	18
9	20
10	21
11	22
12	23
13	24
14	25
15	26
16	27
17	28
18	30
19	31
20	32
21	33
22	34
23	35
24	36
25	36
26	37

27	38
28	39
29	40
30	41
31	42
32	42
33	43
34	44
35	45
36	46
37	47
38	48
39	49
40	50

Informacja o prawach autorskich

Copyright © International Software Testing Qualifications Board (zwana dalej „ISTQB®”).

ISTQB® jest zastrzeżonym znakiem towarowym International Software Testing Qualifications Board.

Wszelkie prawa zastrzeżone.

Prawa autorskie wersji polskiej zastrzeżone dla © Stowarzyszenie Jakości Systemów Informatycznych (SJSI).

Tłumaczenie z języka angielskiego wersji beta – KONTEKST A. Wolski spółka komandytowa.

Przegląd końcowy przeprowadził zespół w składzie: Adam Roman, Monika Petri-Starego, Lucjan Stapp (kierownik zespołu).

Aktualizacja do wersji 1.3 i 1.4: Monika Petri-Starego, Adam Roman.

Autorzy niniejszym przenoszą prawa autorskie na ISTQB®. Autorzy (jako obecni posiadacze praw autorskich) oraz ISTQB® (jako przyszły posiadacz praw autorskich) wyrazili zgodę na następujące warunki użytkowania:

Kopiowanie fragmentów niniejszego dokumentu w celach niekomercyjnych jest dozwolone pod warunkiem wskazania źródła.

Akredytowani dostawcy szkoleń mogą wykorzystywać niniejszy egzamin przykładowy w swoich szkoleniach pod warunkiem wskazania autorów i ISTQB® jako źródła egzaminu i właścicieli praw autorskich do niego. Zastrzega się jednak, że ewentualne materiały reklamowe dotyczące szkolenia mogą być publikowane dopiero po uzyskaniu oficjalnej akredytacji materiałów szkoleniowych ze strony uznawanej przez ISTQB® Rady Krajowej.

Osoby fizyczne i grupy osób fizycznych mogą wykorzystywać niniejszy egzamin przykładowy w artykułach i książkach pod warunkiem wskazania autorów i ISTQB® jako źródła egzaminu i właścicieli praw autorskich do niego.

Korzystanie z egzaminu przykładowego do innych celów bez wcześniejszej pisemnej zgody ISTQB® jest zabronione.

Każda uznawana przez ISTQB® Rada Krajowa może dokonywać przekładu niniejszego egzaminu przykładowego pod warunkiem powielenia powyższych uwag dotyczących praw autorskich w przetłumaczonej wersji dokumentu.

Odpowiedzialność za dokument

Odpowiedzialność za niniejszy dokument ponosi Grupa robocza ISTQB[®] ds. egzaminów.

Obsługą dokumentu zajmuje się podstawowy zespół ISTQB[®], w skład którego wchodzi przedstawiciele Grupy roboczej ds. sylabusu i Grupy roboczej ds. egzaminów.

Podziękowania

Niniejszy dokument został opracowany przez podstawowy zespół ISTQB[®]: Stuart Reid i Adam Roman.

Podstawowy zespół dziękuje zespołowi recenzentów Grupy roboczej ds. egzaminów, Grupy roboczej ds. sylabusu i Radom Krajowym za ich sugestie i wskazówki.

Historia zmian

Wersja	Data	Uwagi
1.4	29.05.2024 r.	Poprawki w odpowiedzi do pytania nr 17.
1.3	14.03.2024 r.	Poprawki w odpowiedzi do pytań: 20, 25
1.2.	20.12.2023 r.	Poprawki zgodnie z dokumentem pytań
1.1	06.11.2023 r.	Poprawka w „kompatybilny z”
1.0	16.10.2023 r.	Pierwsza wersja

Historia zmian polskiej wersji

Wersja	Data	Uwagi
1.0.0.2	17.06.2024 r.	Aktualizacja do wersji 1.4 dokumentu. Poprawka w pytaniu nr 17.
1.0.0.1	06.06.2024 r.	Aktualizacja do wersji 1.3 dokumentu. Poprawki zgodnie z dokumentem pytań. Poprawki w odpowiedzi do pytania 25
1.0.0.0	01.12.2023 r.	Publikacja wersji 1.0.0.0
0.3	15.11.2023 r.	Przegląd i wprowadzanie zmian – Zespół SJSI
0.2	12.11.2023 r.	Przegląd tłumaczenia – Zespół SJSI
0.1	31.10.2023 r.	Tłumaczenie wersji beta: KONTEKST A.Wolski spółka komandytowa
	12.10.2023 r.	Udostępnienie przez ISTQB® wersji końcowej

1. Wstęp

Cel dokumentu

Przykładowe pytania i odpowiedzi wraz z uzasadnieniami przedstawione w niniejszym egzaminie przykładowym zostały opracowane przez zespół ekspertów merytorycznych i doświadczonych autorów pytań w celu:

- udzielenia Radom Krajowym ISTQB® i komisjom egzaminacyjnym pomocy w wykonywaniu czynności związanych z opracowywaniem pytań;
- udostępnienia dostawcom szkoleń i kandydatom przykładowych pytań egzaminacyjnych.

Pytania te nie mogą być wykorzystywane w przedstawionej formie w żadnym oficjalnym egzaminie.

Rzeczywiste egzaminy mogą zawierać szeroką gamę pytań, a niniejszy egzamin przykładowy **nie ma** na celu przedstawienia wszystkich możliwych wariantów, jeśli chodzi o typ, styl czy długość pytań. Ponadto należy pamiętać, że niniejszy egzamin przykładowy może być trudniejszy lub łatwiejszy od egzaminu oficjalnego.

Instrukcje

Niniejszy dokument zawiera:

- tabelę z kluczem odpowiedzi, w tym następujące elementy związane z każdą poprawną odpowiedzią:
 - poziom wiedzy (poziom K), cel nauczania i wartość w punktach;
- tabelę z kluczem odpowiedzi dotyczącą pytań dodatkowych, w tym następujące elementy związane z każdą poprawną odpowiedzią:
 - poziom wiedzy (poziom K), cel nauczania i wartość w punktach;
- zestawy odpowiedzi, w tym następujące elementy związane z każdym pytaniem:
 - poprawna odpowiedź;
 - uzasadnienie każdej możliwej odpowiedzi;
 - poziom wiedzy (poziom K), cel nauczania i wartość w punktach;
- zestawy odpowiedzi dotyczące pytań dodatkowych, w tym następujące elementy związane z każdym pytaniem [dotyczy niektórych egzaminów przykładowych*]:
 - poprawna odpowiedź;
 - uzasadnienie każdej możliwej odpowiedzi;
 - poziom wiedzy (poziom K), cel nauczania i wartość w punktach.

Pytania znajdują się w odrębnym dokumencie.

2. Klucz odpowiedzi

Nr pytania	Poprawna odpowiedź	Cel nauczania	Poziom wiedzy	Punkty
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	c	FL-4.2.1	K3	1

Nr pytania	Poprawna odpowiedź	Cel nauczania	Poziom wiedzy	Punkty
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

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
1	b	<p>a) Odpowiedź niepoprawna. Walidacja, czy udokumentowane wymagania są spełnione, to niepoprawnie sformułowany cel, ponieważ walidacja polega na sprawdzeniu, czy zostały spełnione wymagania i oczekiwania użytkowników, natomiast sprawdzenie, czy zostały spełnione zdefiniowane (wyspecyfikowane) wymagania, jest przedmiotem weryfikacji. Odpowiedź byłaby poprawna, gdybyśmy zastąpili słowo „walidacja” słowem „weryfikacja”.</p> <p>b) Odpowiedź poprawna. Powodowanie awarii i znajdowanie defektów to prawdopodobnie najczęściej występujący cel testowania dynamicznego.</p> <p>c) Odpowiedź niepoprawna. Inicjowanie błędów i identyfikowanie podstawowych przyczyn to niepoprawnie sformułowany cel, ponieważ testerzy nie inicjują błędów, tylko próbują spowodować awarie. Błędy są zwykle popełniane przez programistów (i nie można ich w zasadzie „zainicjować”), a ich skutkami są defekty, które testerzy starają się zidentyfikować albo bezpośrednio z wykorzystaniem testowania statycznego, albo pośrednio poprzez wykrywanie awarii w testowaniu dynamicznym. Identyfikowanie podstawowych przyczyn jest przydatne, jednak stanowi część debugowania, które jest czynnością odrębną od testowania.</p> <p>d) Odpowiedź niepoprawna. Weryfikacja, czy udokumentowane wymagania są spełnione, to niepoprawnie sformułowany cel, ponieważ przedmiotem weryfikacji jest sprawdzenie, czy zostały spełnione zdefiniowane (wyspecyfikowane) wymagania, natomiast walidacja polega na sprawdzeniu, czy zostały spełnione wymagania i oczekiwania użytkowników. Odpowiedź byłaby poprawna, gdybyśmy zastąpili słowo „weryfikacja” słowem „walidacja”.</p>	FL-1.1.1	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
2	c	<p>a) Odpowiedź niepoprawna. W testowaniu dynamicznym rzeczywiście wywołuje się awarie (na których podstawie można zlokalizować i usunąć defekty). Debugowanie wiąże się jednak z lokalizowaniem defektów i ich usuwaniem. Nie polega na usuwaniu awarii.</p> <p>b) Odpowiedź niepoprawna. Zarówno testowanie, jak i debugowanie przyczyniają się do poprawy jakości przedmiotu testów, więc tak naprawdę należy uznać, że oba procesy mają charakter pozytywny. Debugowanie zwykle uważane jest za działanie pozytywne, ponieważ polega na naprawianiu czegoś. Testowanie dynamiczne wiąże się z rozmyślnym powodowaniem awarii przedmiotu testów, dlatego niektórzy uważają je za działanie negatywne, jednak jest to bardzo ograniczony punkt widzenia (który zwykle nie jest podzielany przez testerów). Możliwe jest tworzenie zarówno pozytywnych, jak i negatywnych przypadków testowych. Pozytywne przypadki testowe sprawdzają, czy przedmiot testów poprawnie realizuje wymagane funkcje, natomiast w testach negatywnych sprawdzamy, czy przedmiot testów nie realizuje tego, czego nie powinien realizować.</p> <p>c) Odpowiedź poprawna. W testach stwierdza się występowanie defektów albo bezpośrednio, po zauważeniu defektu w trakcie przeglądów (lub przez narzędzie w ramach analizy statycznej), albo pośrednio przez spowodowanie awarii w testach dynamicznych. Debugowanie jest czynnością odrębną od testowania, na ogół wykonywaną przez programistów. Wiąże się z lokalizowaniem defektów (wyłącznie w testowaniu dynamicznym) i ich usuwaniem.</p> <p>d) Odpowiedź niepoprawna. Przyczynami defektów są zwykle błędy popełnione przez ludzi. W trakcie testowania znajdowane są defekty – albo bezpośrednio w ramach testów statycznych, albo pośrednio przez spowodowanie awarii w testach dynamicznych. Debugowanie pozwala usunąć defekty. Zatem testowanie nie polega na znajdowaniu przyczyn defektów, a debugowanie – na usuwaniu przyczyn defektów.</p>	FL-1.1.2	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
3	b	<p>Zasada mówiąca, że przekonanie o braku defektów jest błędem, odwołuje się do koncepcji, według której samo zapewnienie zgodności z wymaganiami (tj. zweryfikowanie braku defektów implementacji) nie gwarantuje zadowolenia użytkowników z systemu. Należy również przeprowadzić walidację i sprawdzić, czy system zaspokaja potrzeby użytkowników i spełnia ich oczekiwania, czy pozwala zrealizować cele biznesowe i czy ma lepsze parametry od konkurencyjnych rozwiązań.</p> <p>a) Odpowiedź niepoprawna. Zasada „testowanie ujawnia defekty, ale nie może dowieść ich braku” mówi o tym, że chociaż testowanie jest w stanie wykryć obecność defektów w przedmiocie testów, nie może jednak wykazać, że jest on od nich wolny. Skoro nie można wykazać braku defektów, nie można też zagwarantować poprawności przedmiotu testów. Wyjaśnienie, że testowanie nie może wykazać braku defektów, odwołuje się zatem częściowo do tej zasady, a nie do zasady „przekonanie o braku defektów jest błędem”.</p> <p>b) Odpowiedź poprawna. Dzięki wsparciu użytkowników końcowych w wykonywaniu testowania akceptacyjnego powinna być możliwa walidacja zaspokojenia potrzeb i spełnienia oczekiwań użytkowników.</p> <p>c) Odpowiedź niepoprawna. Nie jest możliwe zapewnienie, że w dostarczonym systemie nie pozostaną żadne defekty implementacyjne, gdyż zasada „testowanie pokazuje obecność, a nie brak defektów” wyjaśnia, że choć testowanie może wykryć istnienie defektów w przedmiocie testów, nie jest możliwe wykazać, że nie ma defektów i tym samym zagwarantować poprawność przedmiotu testów.</p> <p>d) Odpowiedź niepoprawna. Zmodyfikowanie testów, które nie powodują awarii, tak aby pozostała pewna liczba defektów, jest jednym ze sposobów realizacji zasady „testy ulegają zużyciu”. Zasada ta odzwierciedla koncepcję, według której powtarzanie tych samych testów z niewielkim prawdopodobieństwem pozwoli wykryć nowe defekty, w związku z czym istotne jest modyfikowanie testów. Takie</p>	FL-1.3.1	K2	1

		działanie nie umożliwi jednak walidacji zaspokojenia potrzeb i spełnienia oczekiwań użytkowników.			
--	--	---	--	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
4	b, e	<p>Zapoznajmy się z opisem analizy testów:</p> <p>Analiza testów polega na przeanalizowaniu podstawy testów w celu zidentyfikowania testowalnych cech oraz na zdefiniowaniu i określeniu priorytetów związanych z nimi warunków testowych, z uwzględnieniem występujących w danym przypadku ryzyk i poziomów ryzyka.</p> <p>Systematyczne identyfikowanie warunków testowych jako elementów pokrycia często wiąże się z zastosowaniem technik testowania, zarówno w trakcie analizy testów, jak i w ramach projektowania testów.</p> <p>Z powyższego opisu wynika, że podczas analizy i projektowania testów często stosuje się różne techniki testowania. Analiza wartości brzegowych i podział na klasy równoważności to przykłady takich technik, a zatem:</p> <p>a) Odpowiedź niepoprawna. W trakcie implementacji testów raczej nie używa się technik testowania, bo polega ona głównie na łączeniu przypadków testowych w procedury testowe, a celem technik testowania jest tworzenie przypadków testowych.</p> <p>b) Odpowiedź poprawna. W ramach projektowania testów prawdopodobnie będziemy korzystać z technik testowania, aby utworzyć przypadki testowe na podstawie warunków testowych i elementów pokrycia.</p> <p>c) Odpowiedź niepoprawna. W trakcie wykonywania testów raczej nie używa się technik testowania, bo polega ono głównie na uruchamianiu procedur testowych implementujących przypadki testowe, natomiast celem technik testowania jest tworzenie przypadków testowych.</p> <p>d) Odpowiedź niepoprawna. W ramach monitorowania testów raczej nie używa się technik testowania. Monitorowanie testów polega głównie na ciągłym sprawdzaniu czynności testowych i dbaniu o realizację planu, a celem technik testowania jest tworzenie przypadków testowych.</p>	FL-1.4.1	K2	1

		e) Odpowiedź poprawna. W trakcie analizy testów prawdopodobnie użyjemy technik testowania do zidentyfikowania warunków testowych.			
--	--	---	--	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
5	a	<p>Rozpatrzmy poszczególne czynności testowe i ich wyjściowe testalia:</p> <p>A. Analiza testów – uszeregowane według priorytetów warunki testowe (4) (np. kryteria akceptacji) i raporty o defektach zidentyfikowanych w podstawie testów.</p> <p>B. Projektowanie testów – uszeregowane według priorytetów przypadki testowe, karty opisu testu, elementy pokrycia (1), wymagania dotyczące danych testowych i środowiska testowego.</p> <p>C. Implementacja testów – procedury testowe, skrypty testów automatycznych, zestawy testowe, dane testowe, harmonogramy wykonywania testów (3) oraz elementy środowiska testowego takie jak zaślepki, sterowniki, symulatory i wirtualizacja usług.</p> <p>D. Ukończenie testów – sumaryczny raport z testów, udokumentowane wnioski, lista czynności do wykonania mających na celu wprowadzenie usprawnień oraz żądania zmiany (2) (w formie elementów backlogu produktu).</p> <p>Zatem POPRAWNE powiązanie to 1B, 2D, 3C, 4A, czyli POPRAWNĄ odpowiedzią jest odpowiedź a).</p>	FL-1.4.3	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
6	c	<p>a) Odpowiedź niepoprawna. Mimo że w przypadku zwinnego wytwarzania oprogramowania rzeczywiście niektóre z zadań związanych z zarządzaniem testami może wykonywać zespół zwinny, to za zadania związane z wykonywaniem testów nie odpowiada jedna osoba spoza zespołu. Jest bardziej prawdopodobne, że testy będą prowadzone przez różnych członków zespołu, zgodnie z podejściem „cały zespół”.</p> <p>b) Odpowiedź niepoprawna. Rola osoby odpowiedzialnej za zarządzanie testami polega głównie na wykonywaniu czynności związanych z planowaniem testów, monitorowaniem testów i nadzorem nad testami oraz ukończeniem testów. Stwierdzenie jest zatem częściowo poprawne, jednak nie można powiedzieć, że osoba występująca w roli związanej z testowaniem ponosi ogólną odpowiedzialność za monitorowanie testów i nadzór nad testami.</p> <p>c) Odpowiedź poprawna. W ramach zwinnego wytwarzania oprogramowania niektóre z zadań związanych z zarządzaniem testami może wykonywać zespół, jednak zadania obejmujące swoim zasięgiem kilka zespołów w organizacji mogą wykonywać kierownicy testów spoza zespołu tworzącego oprogramowanie.</p> <p>d) Odpowiedź niepoprawna. Rola osoby odpowiedzialnej za zarządzanie testami polega głównie na wykonywaniu czynności związanych z planowaniem testów, monitorowaniem testów i nadzorem nad testami oraz ukończeniem testów. Osoba występująca w roli związanej z testowaniem odpowiada głównie za techniczne aspekty testowania. Rola ta polega głównie na podejmowaniu działań związanych z analizą, projektowaniem, implementacją i wykonywaniem testów. Osoba występująca w roli związanej z zarządzaniem testami zwykle nie ponosi odpowiedzialności za analizę i projektowanie testów, chociaż rzeczywiście osoba występująca w roli związanej z testowaniem ponosi ogólną odpowiedzialność za implementację i wykonywanie testów.</p>	FL-1.4.5	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
7	b	<p>a) Odpowiedź niepoprawna. W podejściu „cały zespół” testerzy odgrywają kluczową rolę, przekazując swoją wiedzę w dziedzinie testowania innym członkom zespołu i wpływając na proces wytwarzania produktu. Współpracują z innymi członkami zespołu, aby zagwarantować osiągnięcie wymaganych poziomów jakości. Obejmuje to również współpracę z przedstawicielami jednostek biznesowych w celu stworzenia odpowiednich testów akceptacyjnych. Testerzy współpracują również z programistami w celu uzgodnienia optymalnej strategii testów i podjęcia decyzji co do sposobu automatyzacji testów.</p> <p>b) Odpowiedź poprawna. Dzięki efektywnemu wykorzystaniu różnych zestawów kompetencji poszczególnych członków zespołu podejście „cały zespół” zwiększa dynamikę pracy zespołowej, usprawnia wymianę informacji i współpracę oraz generuje efekt synergii przynoszący korzyści dla całego projektu.</p> <p>c) Odpowiedź niepoprawna. W ramach podejścia „cały zespół” każdy członek zespołu, który dysponuje niezbędną wiedzą i umiejętnościami, może wykonywać dowolne zadania, więc wśród korzyści nie można wskazać specjalizacji członków zespołu.</p> <p>d) Odpowiedź niepoprawna. Nie ma konkretnych zaleceń dotyczących optymalnej wielkości zespołów korzystających z tego podejścia, w szczególności sugerujących, że większe zespoły są lepsze.</p>	FL-1.5.2	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
8	a	<p>a) Odpowiedź poprawna. Główną korzyścią wynikającą z niezależności testowania jest większe prawdopodobieństwo wykrycia przez niezależnych testerów innego rodzaju defektów niż te wykryte przez programistów ze względu na różne doświadczenia, techniczne punkty widzenia i błędy poznawcze. Główną wadą niezależności testowania jest potencjalne odizolowanie niezależnych testerów od zespołu tworzącego oprogramowanie, co może prowadzić do braku współpracy, problemów z wymianą informacji, a nawet konfliktu z tym zespołem. Istnieje ryzyko, że niezależni testerzy zostaną potraktowani jako wąskie gardło i obarczeni winą za opóźnienia w przekazaniu produktu do eksploatacji.</p> <p>b) Odpowiedź niepoprawna. Znajomość kodu przez programistów nie oznacza, że rzadziej znajdują oni defekty – przeciwnie, dzięki temu są w stanie efektywnie wykrywać wiele defektów w tworzonym przez siebie kodzie. Poza tym zwykle uznaje się, że to różne, a nie podobne doświadczenia programistów i testerów sprawiają, że obie grupy wykrywają różne rodzaje defektów.</p> <p>c) Odpowiedź niepoprawna. Testy można prowadzić na różnych poziomach niezależności – od sytuacji, gdy produkty pracy mogą być testowane przez autora (brak niezależności) do sytuacji, gdy są testowane przez testerów spoza organizacji (bardzo wysoki poziom niezależności). W większości projektów testy są prowadzone na wielu poziomach niezależności, np. programiści wykonują testowanie modułowe i testowanie integracji modułów, zespół testowy wykonuje testowanie systemowe i testowanie integracji systemów, a przedstawiciele jednostek biznesowych wykonują testowanie akceptacyjne. Testerzy mogą być zatem członkami zespołu programistów i nie muszą należeć do innej organizacji. Znajomość dziedziny zastosowania może być różna w różnych sytuacjach i nie wynika z poziomu niezależności.</p> <p>d) Odpowiedź niepoprawna. Testy można prowadzić na różnych poziomach niezależności – od sytuacji, gdy produkty pracy mogą być testowane przez autora</p>	FL-1.5.3	K2	1

		(brak niezależności), do sytuacji, gdy są testowane przez testerów spoza organizacji (bardzo wysoki poziom niezależności). Ogólnie rzecz biorąc, testerzy spoza zespołu programistów są bardziej niezależni od testerów z zespołu. Istnieje jednak większe prawdopodobieństwo, że testerzy spoza zespołu programistycznego będą odizolowani od programistów i w rezultacie mogą zostać obarczeni winą za ewentualne opóźnienia w przekazaniu produktu do eksploatacji.			
--	--	--	--	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
9	d	<p>a) Odpowiedź niepoprawna. Kontroli jakości podlegają wszystkie czynności związane z wytwarzaniem, więc z każdą z nich wiąże się odpowiednia czynność testowa. W odpowiedzi jednak zestawiono poziomy testów z poziomami wytwarzania. Pojęcie poziomu testowania jest dobrze określone, nie ma jednak ogólnie przyjętej definicji „poziomu wytwarzania”.</p> <p>b) Odpowiedź niepoprawna. Każdej czynności związanej z wytwarzaniem oprogramowania odpowiada pewna czynność testowa, jednak cele testów mogą się znacznie różnić. Celem testów może być na przykład realizacja wynikającego z umowy wymagania dotyczącego przedmiotu testów i mówiącego o tym, że przed dostarczeniem produktu należy przeprowadzić pewien rodzaj testowania. W takiej sytuacji nie ma powodu definiowania odpowiedniego celu wytwarzania.</p> <p>c) Odpowiedź niepoprawna. Kontroli jakości podlegają wszystkie czynności związane z wytwarzaniem, więc z każdą z nich wiąże się odpowiednia czynność testowa. Nie ma jednak podobnej odpowiedniości między czynnościami testowymi i czynnościami wykonywanymi przez użytkowników. W przypadku niektórych systemów nie da się nawet określić użytkowników końcowych. Poza tym niektóre czynności testowe dotyczą programistów (np. testowanie utrzymywaności) i nie obejmują aspektów związanych z użytkownikami.</p> <p>d) Odpowiedź poprawna. Kontroli jakości podlegają wszystkie czynności związane z wytwarzaniem, więc z każdą z nich wiąże się odpowiednia czynność testowa.</p>	FL-2.1.2	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
10	d	<p>a) Odpowiedź niepoprawna. Wytwarzanie sterowane testami modułowymi nie jest poprawnym przykładem podejścia typu „najpierw test” do wytwarzania oprogramowania.</p> <p>b) Odpowiedź niepoprawna. Wytwarzanie sterowane testami integracyjnymi nie jest poprawnym przykładem podejścia typu „najpierw test” do wytwarzania oprogramowania.</p> <p>c) Odpowiedź niepoprawna. Wytwarzanie sterowane testami systemowymi nie jest poprawnym przykładem podejścia typu „najpierw test” do wytwarzania oprogramowania.</p> <p>d) Odpowiedź poprawna. Wytwarzanie sterowane testami akceptacyjnymi (ATDD) to ogólnie znany przykład podejścia typu „najpierw test” do wytwarzania oprogramowania.</p>	FL-2.1.3	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
11	b	<p>a) Odpowiedź niepoprawna. Celem praktyk stosowanych w ramach „przesunięcia w lewo” jest realizacja większej liczby czynności testowych na wcześniejszych etapach cyklu wytwarzania oprogramowania – przyjmujemy, że cykl ten przedstawiamy jako proces przebiegający na osi czasu od lewej do prawej. Nie ma czegoś takiego jak „lewa strona procesu testowego”.</p> <p>b) Odpowiedź poprawna. Zasada „przesunięcie w lewo” podkreśla znaczenie rozpoczęcia testów na wczesnych etapach cyklu wytwarzania oprogramowania. Przesunięcie w lewo wymaga dodatkowych szkoleń, nakładów pracy i kosztów na wcześniejszym etapie procesu, ale powinno to zostać zrekompensowane dzięki oszczędnościom uzyskanych w ramach całego cyklu.</p> <p>c) Odpowiedź niepoprawna. Chociaż automatyczne testy modułowe i testy integracji modułów są w ogólności przydatne w testowaniu regresji, za tworzenie takich testów zwykle odpowiadają programiści. Jeśli stosowane jest podejście oparte na ciągłej integracji i ciągłym dostarczaniu, to takie testy dostarczane są razem z kodem. W pewnych sytuacjach testerzy mogą automatyzować testy na potrzeby testów regresji i czasami może to dotyczyć również testów modułowych i testów integracji modułów, jednak nie jest to elementem podejścia „przesunięcie w lewo”, w którym testowanie odbywa się na wczesnych etapach cyklu wytwarzania.</p> <p>d) Odpowiedź niepoprawna. Przeszkolenie testerów i podkreślenie znaczenia wykonywania czynności testowych na wczesnych etapach cyklu wytwarzania oprogramowania jest zgodne z podejściem „przesunięcie w lewo”. Automatyzacja dodatkowych czynności testowych wykonywanych na późniejszych etapach nie jest jednak elementem tego podejścia.</p>	FL-2.1.5	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
12	c	<p>a) Odpowiedź niepoprawna. Jednym z celów retrospektywy jest zidentyfikowanie potencjalnych udoskonaleń w procesie, które po zastosowaniu w praktyce powinny doprowadzić do zwiększenia jakości kolejnych produktów procesu wytwarzania (przedmiotów testów). Po przeprowadzeniu retrospektywy możemy zatem osiągnąć wskazany w odpowiedzi rezultat.</p> <p>b) Odpowiedź niepoprawna. Jedną z korzyści wynikających z retrospektywy jest wzrost efektywności testowania dzięki udoskonaleniom w procesie. Po przeprowadzeniu retrospektywy możemy zatem osiągnąć wskazany w odpowiedzi rezultat.</p> <p>c) Odpowiedź poprawna. W retrospektywach uczestniczą zwykle testerzy, programiści, architekci, właściciele produktu i analitycy biznesowi, jednak rzadko są obecni (i rzadko zapraszani) użytkownicy końcowi – raczej nie otrzymują oni również raportów z takich spotkań. Dlatego istnieje bardzo niewielkie prawdopodobieństwo, że dzięki retrospektywom użytkownicy lepiej zrozumieją procesy wytwarzania i procesy testowania.</p> <p>d) Odpowiedź niepoprawna. Jedną z korzyści wynikających z retrospektywy jest podniesienie jakości testaliów (w tym skryptów testów automatycznych) w wyniku wspólnego przeglądu z udziałem programistów. Po przeprowadzeniu retrospektywy możemy zatem osiągnąć wskazany w odpowiedzi rezultat.</p>	FL-2.1.6	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
13	d	<p>a) Odpowiedź niepoprawna. Testowanie modułowe (nazywane także testowaniem jednostkowym) skupia się na oddzielnym testowaniu poszczególnych modułów i polega głównie na weryfikacji zgodności ze specyfikacją, a nie na walidacji zaspokojenia potrzeb użytkowników. Zwykle jednak testy tego rodzaju nie są wykonywane przez testerów, ponieważ zajmują się tym programiści, którzy wykonują te testy w swoim środowisku wytwórczym.</p> <p>b) Odpowiedź niepoprawna. Testowanie integracji modułów skupia się na interfejsach i interakcjach między modułami. Polega głównie na weryfikacji zgodności ze specyfikacją, a nie na walidacji zaspokojenia potrzeb użytkowników. Zwykle jednak testy tego rodzaju nie są wykonywane przez testerów, ponieważ zajmują się tym programiści.</p> <p>c) Odpowiedź niepoprawna. Testowanie integracji systemów skupia się na interfejsach z innymi systemami oraz usługami zewnętrznymi. Polega głównie na weryfikacji zgodności ze specyfikacją, a nie na walidacji zaspokojenia potrzeb użytkowników. Ponadto testy tego rodzaju są najczęściej wykonywane przez testerów.</p> <p>d) Odpowiedź poprawna. Testowanie akceptacyjne skupia się na przeprowadzeniu walidacji i wykazaniu, że system jest gotowy do wdrożenia (tzn. zaspokaja potrzeby biznesowe użytkownika). W idealnym przypadku testy tego rodzaju są wykonywane przez użytkowników końcowych.</p>	FL-2.2.1	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
14	b	<p>a) Odpowiedź niepoprawna. Testowanie potwierdzające pozwala ustalić, czy w efekcie aktualizacji powstała poprawna implementacja, więc jest ono niezbędne. Celowe byłoby również jednak przeprowadzenie testów regresji i potwierdzenie, że nie wprowadzono ani nie ujawniono defektów w obszarach systemu, które nie podlegały zmianom.</p> <p>b) Odpowiedź poprawna. Testowanie potwierdzające pozwala ustalić, czy w efekcie aktualizacji powstała poprawna implementacja. Następnie należy przeprowadzić testy regresji i potwierdzić, że nie wprowadzono ani nie ujawniono defektów w obszarach systemu, które nie podlegały zmianom.</p> <p>c) Odpowiedź niepoprawna. Należy przeprowadzić testy regresji i potwierdzić, że po aktualizacji nie wprowadzono ani nie ujawniono defektów w obszarach systemu, które nie podlegały zmianom. Niezbędne jest jednak również wykonanie testów potwierdzających, że w efekcie aktualizacji powstała poprawna implementacja.</p> <p>d) Odpowiedź niepoprawna. Testowanie potwierdzające pozwala ustalić, czy w efekcie aktualizacji powstała poprawna implementacja. Następnie należy przeprowadzić testy regresji i potwierdzić, że nie wprowadzono ani nie ujawniono defektów w obszarach systemu, które nie podlegały zmianom. Jednak w przypadku konieczności przetestowania aktualizacji testowanie potwierdzające należy wykonać przed testowaniem regresji.</p>	FL-2.2.3	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
15	d	<p>Rozważmy poszczególne przykładowe defekty umieszczone na liście:</p> <ol style="list-style-type: none"> 1. W związku z dużą złożonością projektu występuje niezgodność dwóch różnych elementów specyfikacji projektowej – to przykład defektu w specyfikacji. Defekty tego typu to m.in. niespójności, niejednoznaczności, sprzeczności, przeoczenia, nieścisłości i powtórzenia, a najłatwiej je wykryć za pomocą testowania statycznego. 2. Czas odpowiedzi jest zbyt długi i użytkownicy tracą cierpliwość – to przykład defektu dotyczącego czasu odpowiedzi. Defekty tego typu można wykryć tylko po uruchomieniu programu i zmierzeniu czasu, a zrobić to można tylko w ramach testowania dynamicznego. 3. Ścieżka w kodzie nie może zostać osiągnięta w trakcie wykonywania – to przykład defektu w kodzie. Defekty tego typu to m.in. zmienne z niezdefiniowanymi wartościami, niezadeklarowane zmienne, wielokrotnie powtórzony lub nieosiągalny kod oraz kod o nadmiernej złożoności, a najłatwiej je wykryć za pomocą testowania statycznego. 4. Zmienna została zadeklarowana, ale nie jest używana w programie – to przykład defektu w kodzie. Defekty tego typu to m.in. zmienne z niezdefiniowanymi wartościami, niezadeklarowane zmienne, wielokrotnie powtórzony lub nieosiągalny kod oraz kod o nadmiernej złożoności, a najłatwiej je wykryć za pomocą testowania statycznego. 5. Ilość pamięci, której wymaga program do wygenerowania raportu, jest zbyt duża – to przykład defektu związanego z wydajnością. Defekty tego typu można wykryć tylko w działaniu, po uruchomieniu programu i zmierzeniu wykorzystania pamięci, a najłatwiej to zrobić w ramach testowania dynamicznego. <p>Z testowaniem statycznym możemy powiązać zestaw defektów (1, 3, 4), zatem odpowiedzią POPRAWNĄ jest odpowiedź d).</p>	FL-3.1.3	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
16	a	<p>a) Odpowiedź poprawna. Częste otrzymywanie informacji zwrotnych od interesariuszy, również we wczesnych etapach procesu wytwarzania oprogramowania, może przynieść wiele korzyści. Umożliwia wczesne rozpoznawanie i sygnalizowanie potencjalnych problemów z jakością, pozwala uniknąć ewentualnych nieporozumień w kwestii wymagań, a także umożliwia lepsze zrozumienie i szybsze wprowadzanie ewentualnych zmian w wymaganiach interesariuszy.</p> <p>b) Odpowiedź niepoprawna. Przekazywanie informacji zwrotnych przez interesariuszy i same informacje zwrotne raczej nie zwiększą u interesariuszy stopnia zrozumienia własnych wymagań przez użytkowników.</p> <p>c) Odpowiedź niepoprawna. Częste otrzymywanie informacji zwrotnych od interesariuszy, również we wczesnych etapach procesu wytwarzania oprogramowania, może przynieść wiele korzyści. Umożliwia wczesne rozpoznawanie i sygnalizowanie potencjalnych problemów z jakością, pozwala uniknąć ewentualnych nieporozumień w kwestii wymagań, a także umożliwia lepsze zrozumienie i szybsze wprowadzanie ewentualnych zmian w wymaganiach interesariuszy. Jednak chociaż zmiany w wymaganiach mogą być lepiej analizowane i szybciej wprowadzane, nie oznacza to, że nieograniczona możliwość dokonywania zmian jest zalecanym podejściem.</p> <p>d) Odpowiedź niepoprawna. Otrzymujemy informacje zwrotne od interesariuszy, a nie przekazujemy im informacje. Moglibyśmy powiadomić użytkowników końcowych przed wydaniem produktu między innymi o wymaganiach, które nie zostaną zaimplementowane, jednak najlepiej w ogóle pominąć taką komunikację.</p>	FL-3.2.1	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
17	b	<p>W odniesieniu do poszczególnych typów przeglądów z podanej listy:</p> <ol style="list-style-type: none"> Przeгляд techniczny – wykonują go przeglądający, którzy dysponują odpowiednimi kwalifikacjami technicznymi i jest prowadzony przez moderatora. Celami przeglądu są nie tylko osiągnięcie konsensusu i podjęcie decyzji dotyczących problemów technicznych, ale także dokonanie oceny jakości produktu pracy i zwiększenie do niego zaufania, wygenerowanie nowych pomysłów, zmotywowanie autorów do wprowadzania udoskonaleń i stworzenie im do tego warunków oraz wykrycie anomalii. Przeгляд nieformalny – głównym celem jest wykrycie anomalii. Przeglądy nieformalne nie przebiegają zgodnie ze zdefiniowanym procesem, a uzyskanych dzięki nim informacji nie trzeba formalnie dokumentować. Inspekcja – jest najbardziej formalnym typem przeglądu, w związku z czym odbywa się zgodnie z pełnym ogólnym procesem przeglądu. Jej głównym celem jest wykrycie jak największej liczby anomalii, a pozostałe cele to: dokonanie oceny jakości produktu pracy i zwiększenie do niego zaufania oraz zmotywowanie autorów do wprowadzania udoskonaleń i stworzenie im do tego warunków, a także zebranie metryk, które można wykorzystać następnie do udoskonalenia cyklu wytwarzania oprogramowania (w tym procesu inspekcji). W ramach inspekcji autor nie może być liderem przeglądu ani protokolantem. Przejrzenie – prowadzi je autor i może służyć wielu celom, takim jak: dokonanie oceny jakości produktu pracy i zwiększenie do niego zaufania, edukowanie przeglądających, osiągnięcie konsensusu, wygenerowanie nowych pomysłów, zmotywowanie autorów do udoskonalania przyszłych produktów pracy i stworzenie im do tego warunków oraz wykrycie anomalii. Przed rozpoczęciem przejrzania przeglądający mogą przeprowadzić przegląd indywidualny, ale nie jest to konieczne. <p>A. Wśród celów są osiągnięcie konsensusu, wygenerowanie nowych pomysłów i zmotywowanie autorów do udoskonalania przyszłych produktów pracy.</p>	FL-3.2.4	K2	1

		<p>B. Służy takim celom jak edukowanie przeglądających, osiągnięcie konsensusu, wygenerowanie nowych pomysłów i wykrycie potencjalnych defektów.</p> <p>C. Głównym celem jest wykrycie potencjalnych defektów. Obejmuje zbieranie metryk wykorzystywanych do doskonalenia procesu.</p> <p>D. Głównym celem jest wykrycie potencjalnych defektów. Uzyskanych informacji nie trzeba formalnie dokumentować.</p> <p>Zatem POPRAWNE powiązanie to 1A, 2D, 3C, 4B, czyli POPRAWNĄ odpowiedzią jest odpowiedź b).</p>			
--	--	---	--	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
18	b	<p>a) Odpowiedź niepoprawna. Warunkiem powodzenia każdego przeglądu jest uzyskanie wsparcia kierownictwa dla procesu przeglądu, jednak nie oznacza to, że przedstawiciele kierownictwa powinni występować w roli przeglądających.</p> <p>b) Odpowiedź poprawna. Jednym z czynników powodzenia przeglądu jest podział produktu pracy na mniejsze partie materiału, które można poddać przeglądowi w rozsądnych ramach czasowych i na których uczestnicy będą w stanie się skoncentrować podczas przeglądu indywidualnego lub spotkania związanego z przeglądem.</p> <p>c) Odpowiedź niepoprawna. Ważnym czynnikiem powodzenia przeglądu jest określenie jednoznacznych celów i mierzalnych kryteriów wyjścia, jednak celem w żadnym razie nie powinna być ocena uczestników.</p> <p>d) Odpowiedź niepoprawna. Jednym z warunków powodzenia przeglądu jest jego przeprowadzenie w odniesieniu do mniejszych partii materiału, na których uczestnicy będą w stanie się skoncentrować podczas przeglądu indywidualnego lub spotkania związanego z przeglądem. Nie należy więc planować przeglądów w taki sposób, aby obejmowały tylko jeden dokument.</p>	FL-3.2.5	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
19	c	<p>a) Odpowiedź niepoprawna. W większości przypadków czarnoskrzynkowe techniki testowania i techniki testowania oparte na doświadczeniu mogą być stosowane dla tych samych przedmiotów testów.</p> <p>b) Odpowiedź niepoprawna. Zarówno czarnoskrzynkowe techniki testowania, jak i techniki testowania oparte na doświadczeniu mogą być stosowane na wszystkich poziomach testów.</p> <p>c) Odpowiedź poprawna. Czarnoskrzynkowe techniki testowania (zwane także technikami opartymi na specyfikacji) bazują na analizie wyspecyfikowanego zachowania przedmiotu testów, bez odwoływania się do jego struktury wewnętrznej. Podstawą testów jest więc zwykle specyfikacja. Techniki testowania oparte na doświadczeniu pozwalają w efektywny sposób wykorzystać wiedzę i doświadczenie testerów do projektowania i implementowania przypadków testowych. Oznacza to, że testerzy podczas projektowania testów mogą w ogóle nie korzystać ze specyfikacji.</p> <p>d) Odpowiedź niepoprawna. Techniki oparte na doświadczeniu pozwalają wykrywać defekty, które łatwo jest przeoczyć w przypadku stosowania technik czarnoskrzynkowych i białoskrzynkowych, dlatego stanowią one uzupełnienie czarnoskrzynkowych i białoskrzynkowych technik testowania. Ponadto w każdym cyklu wytwarzania oprogramowania można zastosować czarnoskrzynkowe techniki testowania i techniki testowania oparte na doświadczeniu.</p>	FL-4.1.1	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
20	c	<p>a) Odpowiedź niepoprawna. Trzy podane wartości zapewniają pełne pokrycie klas równoważności, jednak do uzyskania pełnego pokrycia wystarczą dwie wartości.</p> <p>b) Odpowiedź niepoprawna. Trzy podane wartości zapewniają pełne pokrycie klas równoważności, jednak do uzyskania pełnego pokrycia wystarczą dwie wartości.</p> <p>c) Odpowiedź poprawna. Wartość „1” pokrywa klasy „niepoprawna długość” i „niepoprawna liczba różnych cyfr”. Wartość „1234” pokrywa klasy „poprawna długość” i „poprawna liczba różnych cyfr”. Dwie podane wartości pokrywają wszystkie cztery zidentyfikowane klasy równoważności.</p> <p>d) Odpowiedź niepoprawna. Ten zestaw nie pokrywa klasy równoważności „niepoprawna liczba różnych cyfr”.</p>	FL-4.2.1	K3	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
21	d	<p>Istnieją następujące klasy równoważności: {..., 99, 100}, {101, 102, ..., 198, 199}, {200, 201, ...}.</p> <p>Dlatego są 4 wartości brzegowe: 100, 101, 199 i 200.</p> <p>W przypadku dwupunktowej analizy wartości brzegowych na każdą wartość brzegową przypadają dwa elementy pokrycia (sama wartość brzegowa oraz najbliższa jej wartość należąca do sąsiedniej klasy równoważności). Ponieważ najbliższe wartości są jednocześnie wartościami brzegowymi w sąsiednich klasach równoważności, występują jedynie cztery elementy pokrycia.</p> <p>Tym samym:</p> <p>a) Odpowiedź niepoprawna. Jedynie 100 i 200 są poprawnymi elementami pokrycia w dwupunktowej analizie wartości brzegowych, uzyskujemy zatem tylko 50% pokrycia.</p> <p>b) Odpowiedź niepoprawna. Jedynie 100 i 200 są poprawnymi elementami pokrycia w dwupunktowej analizie wartości brzegowych, uzyskujemy zatem tylko 50% pokrycia.</p> <p>c) Odpowiedź niepoprawna. Jedynie 100 i 101 są poprawnymi elementami pokrycia w dwupunktowej analizie wartości brzegowych, uzyskujemy zatem tylko 50% pokrycia.</p> <p>d) Odpowiedź poprawna. 101, 199 i 200 są poprawnymi elementami pokrycia w dwupunktowej analizie wartości brzegowych, uzyskujemy zatem 75% pokrycia.</p>	FL-4.2.2	K3	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
22	d	<p>a) Odpowiedź niepoprawna. Kombinacja (P, P, F) nie jest zgodna z żadną regułą. To przykład pominięcia, a nie sprzeczności.</p> <p>b) Odpowiedź niepoprawna. Kombinacja (P, F, P) odpowiada tylko jednej kolumnie (R2), więc sprzeczność nie występuje.</p> <p>c) Odpowiedź niepoprawna. Obie kombinacje – (P, P, P) i (F, P, P) – odpowiadają tylko jednej kolumnie (R1), więc sprzeczność nie występuje.</p> <p>d) Odpowiedź poprawna. Kombinacja (F, F, F) odpowiada zarówno kolumnie R2, jak i R3, ale zdefiniowano w nich różne akcje. Występuje zatem sprzeczność między R2 i R3.</p>	FL-4.2.3	K3	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
23	a	<p>Trzy poniższe przejścia:</p> <p>„WNIOSEK PRZYJĘTY → POTWIERDZENIE”</p> <p>„LISTA OCZEKUJĄCYCH → POTWIERDZENIE”</p> <p>„LISTA OCZEKUJĄCYCH → KONIEC”</p> <p>nie mogą wystąpić w tym samym przypadku testowym, co oznacza, że będą potrzebne co najmniej trzy przypadki testowe. Wszystkie pozostałe przejścia mogą wystąpić w połączeniu z jednym z powyższych trzech przejść, więc potrzebujemy co najmniej trzech przypadków testowych. Możliwe są zresztą jedynie trzy sekwencje:</p> <p>PT1: START (Wniosek o rezerwację sali) → WNIOSEK PRZYJĘTY (Dostępna) → POTWIERDZENIE (Zapłać) → KONIEC</p> <p>PT2: START (Wniosek o rezerwację sali) → WNIOSEK PRZYJĘTY (Niedostępna) → LISTA OCZEKUJĄCYCH (Dostępna) → POTWIERDZENIE (Zapłać) → KONIEC</p> <p>PT3: START (Wniosek o rezerwację sali) → WNIOSEK PRZYJĘTY (Niedostępna) → LISTA OCZEKUJĄCYCH (Anuluj) → KONIEC</p> <p>Tym samym:</p> <ul style="list-style-type: none"> a) Odpowiedź poprawna b) Odpowiedź niepoprawna c) Odpowiedź niepoprawna d) Odpowiedź niepoprawna 	FL-4.2.4	K3	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
24	c	W testowaniu gałęzi elementami pokrycia są gałęzie kodu reprezentowane przez krawędzie grafu przepływu sterowania. W grafie mamy 8 krawędzi. Poprawną odpowiedzią jest zatem odpowiedź c).	FL-4.3.2	K2	1
25	a	<p>a) Odpowiedź poprawna. Wykonanie jedynie testowania czarnoskrzynkowego nie pozwala zmierzyć faktycznego pokrycia kodu, natomiast miary pokrycia stosowane w technikach białoskrzynkowych zapewniają obiektywny pomiar pokrycia i dostarczają niezbędnych informacji umożliwiających wygenerowanie dodatkowych testów w celu jego zwiększenia, co ostatecznie przekłada się na wzrost zaufania do kodu.</p> <p>b) Odpowiedź niepoprawna. Stwierdzenie jest prawdziwe, ale nie stanowi odpowiedzi na pytanie (nie ma związku z testowaniem czarnoskrzynkowym).</p> <p>c) Odpowiedź niepoprawna. Ogólnie rzecz biorąc, białoskrzynkowe techniki testowania nie obejmują zakresu czarnoskrzynkowych technik testowania.</p> <p>d) Odpowiedź niepoprawna. Białoskrzynkowe techniki testowania służą do projektowania testów opartych na przedmiocie testów, natomiast czarnoskrzynkowe techniki testowania służą do projektowania testów opartych na specyfikacji. Nie ma zatem związków między elementami pokrycia wyprowadzanymi w oparciu o oba rodzaje technik.</p>	FL-4.3.3	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
26	b	<p>a) Odpowiedź niepoprawna. W testowaniu eksploracyjnym używa się kart opisu testu, a nie listy możliwych defektów/awarii. Mimo że w testowaniu eksploracyjnym możemy zastosować inne techniki testowania, w tym przypadku najbardziej prawdopodobną opcją jest użycie ataku usterek.</p> <p>b) Odpowiedź poprawna. To lista możliwych awarii. Atak usterek to metodyczne podejście do implementacji zgadywania błędów. Technika ta wymaga od testera stworzenia lub uzyskania listy potencjalnych błędów, defektów i awarii, a następnie zaprojektowania testów pozwalających zidentyfikować defekty związane z błędami i uwidocznienie te defekty bądź spowodowanie awarie.</p> <p>c) Odpowiedź niepoprawna. Tester używa elementów listy kontrolnej jako warunków testowych. Zarówno w zgadywaniu błędów, jak i w testowaniu w oparciu o listę kontrolną występują takie listy, jednak w tym wypadku mamy do czynienia z listą możliwych awarii, a nie warunków testowych. Zatem NAJBARDZIEJ prawdopodobne jest tu zastosowanie techniki ataku usterek, która skupia się na błędach, defektach i awariach.</p> <p>d) Odpowiedź niepoprawna. Analiza wartości brzegowych opiera się na wartościach brzegowych klas równoważności. Na powyższej liście nie wspomniano o klasach równoważności i ich brzegach.</p>	FL-4.4.1	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
27	d	<p>a) Odpowiedź niepoprawna. Na podstawie listy kontrolnej testerzy rzeczywiście mogą zaimplementować i wykonać szczegółowe przypadki testowe, jednak odpowiedź nie wyjaśnia, w jaki sposób mogłoby to się przyczynić do zwiększenia pokrycia.</p> <p>b) Odpowiedź niepoprawna. Elementów listy kontrolnej nie należy automatyzować. Jednak nawet jeśli tak się stanie, skrypty testów automatycznych zawsze wykonują testowanie w ten sam sposób, co na ogół nie powoduje zwiększenia pokrycia.</p> <p>c) Odpowiedź niepoprawna. Każdy element listy kontrolnej rzeczywiście powinien być testowany odrębnie i niezależnie od innych. Ma to jednak wpływ na kolejność wykonywania testów, a nie na osiągnięte pokrycie, w szczególności więc nie powoduje zwiększenia pokrycia.</p> <p>d) Odpowiedź poprawna. Jeśli listy kontrolne mają charakter ogólny, prawdopodobnie wystąpią pewne różnice w testowaniu, a to może oznaczać potencjalne zwiększenie pokrycia, ale także mniejszą powtarzalność. Jeśli dwaj testerzy posługują się listą kontrolną zawierającą ogólne elementy, każdy z nich może użyć innych danych testowych, innych kroków testów itd. Dzięki temu jeden z testerów prawdopodobnie pokryje pewne obszary, którymi nie zajął się drugi tester, co pozwoli zwiększyć pokrycie.</p>	FL-4.4.3	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
28	b	<p>a) Odpowiedź niepoprawna. To kryterium akceptacji wskazuje przepisy, z którymi system powinien być zgodny (w tym przypadku prawo do usunięcia danych). To przykład kryterium ukierunkowanego na reguły.</p> <p>b) Odpowiedź poprawna. To kryterium akceptacji opisuje przykładowy scenariusz, który musi być możliwy do zrealizowania przez system. To przykład kryterium ukierunkowanego na scenariusze.</p> <p>c) Odpowiedź niepoprawna. Ta odpowiedź przypomina linię kodu implementującego pewną regułę biznesową. Kryteria akceptacji powinny być zapisywane we współpracy z przedstawicielami jednostek biznesowych, a więc w języku zrozumiałym dla takich osób. Kryterium w tej formie prawdopodobnie będzie dla tych interesariuszy nieczytelne.</p> <p>d) Odpowiedź niepoprawna. To kryterium akceptacji opisuje przepisy, z którymi system powinien być zgodny oraz sposób zapewnienia tej zgodności. Jest to zatem przykład kryterium ukierunkowanego na reguły, a nie na scenariusze.</p>	FL-4.5.2	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
29	d	<p>a) Odpowiedź niepoprawna. Chcemy sprawdzić, czy specjaliści użytkownicy mają prawa zwykłych użytkowników, więc musimy przetestować prawa dostępu specjalnego użytkownika, a nie zwykłego użytkownika.</p> <p>b) Odpowiedź niepoprawna. Chcemy sprawdzić, czy specjaliści użytkownicy mają prawa zwykłych użytkowników, więc musimy przetestować prawa dostępu specjalnego użytkownika, a nie zwykłego użytkownika.</p> <p>c) Odpowiedź niepoprawna. Kryteria akceptacji nie wspominają o piętrze 5. Przypadki testowe nie powinny poszerzać zakresu historii użytkownika. Jednak nawet gdybyśmy chcieli prowadzić testowanie negatywne, ten test nie jest bezpośrednio powiązany z kryterium KA3.</p> <p>d) Odpowiedź poprawna. W ten sposób jesteśmy w stanie sprawdzić, czy specjalny użytkownik ma dostęp do pięter, które są dostępne dla zwykłych użytkowników.</p>	FL-4.5.3	K3	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
30	a	<p>a) Odpowiedź poprawna. Plan testów może zawierać <i>wymagania</i> dotyczące danych testowych (w ramach podejścia do testowania), jednak nie szczegółowe dane testowe dla przypadków testowych. Dane testowe wchodzi w skład przypadków testowych, a nie planu testów. Ponadto w momencie tworzenia planu zwykle nie jest możliwe zdefiniowanie takich danych, ponieważ nie wiadomo jeszcze, jak będą wyglądać poszczególne moduły.</p> <p>b) Odpowiedź niepoprawna. Jednym z celów planu testów jest zagwarantowanie, że wykonywane czynności testowe spełnią ustalone kryteria, dlatego uwzględnia się w nim kryteria wejścia i kryteria wyjścia. Kryteria pokrycia kodu są przykładem takich kryteriów na poziomie testów modułowych.</p> <p>c) Odpowiedź niepoprawna. Szablony dokumentacji to typowa zawartość planu testów. Ułatwiają one komunikację między interesariuszami dzięki zdefiniowaniu standardowego sposobu przekazywania informacji lub raportowania.</p> <p>d) Odpowiedź niepoprawna. Jednym z celów planu testów jest wykazanie, że testowanie będzie odbywać się zgodnie z dotychczasową polityką testów i strategią testów, ewentualnie uzasadnienie odstępstw od polityki i strategii testów. To przykład uzasadnienia odstępstw dotyczących poziomów testów, które zostaną (lub nie zostaną) uwzględnione.</p>	FL-5.1.1	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
31	c	<p>Z wykresu wiemy, że $A(4)=6$ i $A(3)=8$ (dwa ostatnie szare prostokąty).</p> <p>Po podstawieniu do wzoru otrzymujemy:</p> $E(5) = (3 \cdot A(4) + A(3)) / 4 = (3 \cdot 6 + 8) / 4 = 26 / 4 = 6,5 \text{ osobodnia.}$ <p>Z tego powodu POPRAWNĄ odpowiedzią jest odpowiedź c).</p>	FL-5.1.4	K3	1
32	a	<p>Chcemy wykonywać przypadki testowe według priorytetów, ale musimy również uwzględnić zależności między nimi.</p> <p>Jeśli weźmiemy pod uwagę jedynie priorytety, najpierw należy wykonać przypadki PT 5 i PT 7 (o najwyższym priorytecie), potem PT 1, PT 3 i PT 4, a na koniec PT 2 i PT 6 (o najniższym priorytecie).</p> <p>Jednak żeby wykonać PT 7, musimy najpierw wykonać PT 4.</p> <p>Żeby wykonać PT 5, musimy wykonać PT 4 i PT 2, jednak PT 2 jest zależny od PT 1. Należy go zatem wykonać przed PT 2.</p> <p>Jeśli zatem chcemy jak najwcześniej wykonać przypadki testowe o priorytecie 1, kolejność pierwszych pięciu przypadków będzie następująca: PT 4 – PT 7 – PT 1 – PT 2 – PT 5.</p> <p>Następnie trzeba wykonać PT 3, który ma wyższy priorytet niż PT 6.</p> <p>Pełny harmonogram ma zatem postać: PT 4 – PT 7 – PT 1 – PT 2 – PT 5 – PT 3 – PT 6.</p> <p>Szóstym przypadkiem testowym jest więc PT 3.</p> <p>Dlatego POPRAWNĄ odpowiedzią jest odpowiedź a).</p>	FL-5.1.5	K3	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
33	b	<p>a) Odpowiedź niepoprawna. Model piramidy testów nie uwzględnia informacji o priorytetach testów.</p> <p>b) Odpowiedź poprawna. Piramida testów to model odzwierciedlający fakt, że testy mogą różnić się poziomem szczegółowości.</p> <p>c) Odpowiedź niepoprawna. Model piramidy testów nie opiera się na kryteriach pokrycia.</p> <p>d) Odpowiedź niepoprawna. Model piramidy testów nie odzwierciedla relacji między różnymi testami.</p>	FL-5.1.6	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
34	d	<p>a) Odpowiedź niepoprawna. Kwadranty testowe służą do odrębnego grupowania poziomów testów i typów testów zgodnie z różnymi kryteriami. Nie reprezentują kombinacji poziomów testów i typów testów i nie są związane z miejscem w ramach cyklu wytwarzania oprogramowania. W modelu kwadrantów testowych poziomy testów i typy testów są rozpatrywane odrębnie.</p> <p>b) Odpowiedź niepoprawna. Kwadranty testowe służą do grupowania poziomów testów i typów testów zgodnie z różnymi kryteriami. Nie opisują poziomu szczegółowości różnych typów testów wykonywanych na każdym poziomie testów. Taki model – uwzględniający poziomy testów – jest nazywany piramidą testów.</p> <p>c) Odpowiedź niepoprawna. Stwierdzenie nie jest poprawne, ponieważ w ogólności test dowolnego typu może zostać przeprowadzony na dowolnym poziomie testów.</p> <p>d) Odpowiedź poprawna. Kwadranty testowe służą do grupowania poziomów testów, typów testów, czynności, technik testowania i produktów pracy w kontekście zwinnego wytwarzania oprogramowania. W opisywanym modelu testy mogą mieć cel biznesowy lub technologiczny oraz mogą służyć wspieraniu zespołu (tzn. poprzez ukierunkowanie procesu wytwarzania oprogramowania) lub krytyce produktu (tzn. poprzez mierzenie jego zachowania względem oczekiwań). Wypadkowa obu aspektów decyduje o przynależności do jednego z czterech kwadrantów.</p>	FL-5.1.7	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
35	c	<p>a) Odpowiedź niepoprawna. Monitorowanie ryzyka to element kontroli ryzyka, a nie analizy ryzyka.</p> <p>b) Odpowiedź niepoprawna. Samo zidentyfikowanie ryzyka nie umożliwia wdrożenia działań łagodzących ryzyko. Działania łagodzące są definiowane w fazie kontroli ryzyka.</p> <p>c) Odpowiedź poprawna. To rzeczywiście przykład wpływu analizy ryzyka na staranność i zakres testowania.</p> <p>d) Odpowiedź niepoprawna. Elementy pokrycia są definiowane z wykorzystaniem technik testowania, a nie w wyniku analizy ryzyka.</p>	FL-5.2.3	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
36	b	<p>a) Odpowiedź niepoprawna. Raporty o postępie testów są używane głównie w ramach monitorowania testów, nadzoru nad testami i ukończenia testów, a nie podczas projektowania testów.</p> <p>b) Odpowiedź poprawna. Sumaryczny raport z testów jest sporządzany na etapie ukończenia testów – po zakończeniu realizacji projektu bądź po wykonaniu testów danego poziomu lub typu oraz, w sytuacji idealnej, po spełnieniu kryteriów wyjścia. W raporcie tym wykorzystywane są dane z raportów o postępie testów oraz inne informacje.</p> <p>c) Odpowiedź niepoprawna. Raporty o postępie testów są używane głównie w ramach monitorowania testów, nadzoru nad testami i ukończenia testów, a nie podczas analizy testów.</p> <p>d) Odpowiedź niepoprawna. Raporty o postępie testów są używane głównie w ramach monitorowania testów, nadzoru nad testami i ukończenia testów, a nie podczas planowania testów.</p>	FL-5.3.2	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
37	d	<p>a) Odpowiedź niepoprawna. Kiedy użytkownik zgłasza awarię oprogramowania, dzięki jednoznaczemu identyfikowaniu zatwierdzonych elementów można zestawić pliki z wersji oprogramowania, z której korzysta użytkownik (wraz z odpowiednimi wersjami skryptów testowych), a następnie odtworzyć awarię i szybciej zlokalizować defekt.</p> <p>b) Odpowiedź niepoprawna. Jeśli zmiana w środowisku testowym powoduje nieoczekiwane problemy podczas testowania, dzięki zarządzaniu konfiguracją testerzy mogą przywrócić poprzednią wersję środowiska. Pozwala to kontynuować testowanie i uniknąć wpływu wprowadzonej zmiany.</p> <p>c) Odpowiedź niepoprawna. Dzięki zarządzaniu konfiguracją wszystkie zidentyfikowane dokumenty (np. specyfikacje wymagań) i elementy oprogramowania są przywoływane w sposób jednoznaczny w dokumentacji testów (np. w planach testów).</p> <p>d) Odpowiedź poprawna. Zapewnia to zarządzanie defektami, a nie zarządzanie konfiguracją.</p>	FL-5.4.1	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
38	b	<p>a) Odpowiedź niepoprawna. To istotne dane, ale nie tak ważne jak informacje o elementach środowiska testowego.</p> <p>b) Odpowiedź poprawna. Istotne informacje, których brakuje w raporcie, to określenie przeglądarki i urządzenia wykorzystywanego w trakcie testowania. Informacje o przeglądarce i urządzeniu są istotne, ponieważ opisany defekt może być zależny od tych elementów. Przycisk logowania może na przykład działać poprawnie w jednej przeglądarce (lub w pewnej wersji konkretnej przeglądarki), a nie działać w innej. Dlatego informacje o przeglądarce i urządzeniu mogą ułatwić programistom odtworzenie defektu i szybsze ustalenie jego podstawowej przyczyny.</p> <p>c) Odpowiedź niepoprawna. Przedmiot testów został określony (aplikacja WebShop ver. 0.99).</p> <p>d) Odpowiedź niepoprawna. Wskazano wpływ defektu – to jego krytyczność (wysoka).</p>	FL-5.5.1	K3	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
39	d	<p>a) Odpowiedź niepoprawna. Narzędzia do wykonywania testów i pomiaru pokrycia ułatwiają automatyczne wykonywanie przypadków testowych i mierzenie pokrycia osiąganego po wykonaniu tych przypadków. Nie ułatwiają jednak zarządzania defektami i zarządzania konfiguracją.</p> <p>b) Odpowiedź niepoprawna. Narzędzia do projektowania i implementacji testów ułatwiają generowanie przypadków testowych, danych i procedur testowych, nie wspierają jednak zarządzania defektami i zarządzania konfiguracją .</p> <p>c) Odpowiedź niepoprawna. Narzędzia do zarządzania defektami służą do zarządzania defektami, ale nie są narzędziami do testowania i nie służą do porządkowania przypadków testowych i zarządzania konfiguracją.</p> <p>d) Odpowiedź poprawna. Narzędzia do zarządzania testowaniem zwiększają efektywność procesu testowego poprzez ułatwienie zarządzania cyklem wytwarzania oprogramowania, wymaganiami, testami, defektami i konfiguracją.</p>	FL-6.1.1	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
40	d	<p>a) Odpowiedź niepoprawna. Nie jest możliwe generowanie przypadków testowych bez dostępu do podstawy testów. Generowanie przypadków testowych zarówno przez testerów, jak i przez narzędzia wymaga dostępu do podstawy testów.</p> <p>b) Odpowiedź niepoprawna. Uzyskanie większego pokrycia dzięki bardziej obiektywnej ocenie nie jest bezpośrednią korzyścią z automatyzacji testów. Dzięki automatyzacji testów możliwa jest bardziej obiektywna ocena pokrycia, jednak taka obiektywna ocena nie powoduje jego zwiększenia. Dopiero po określeniu pokrycia można napisać kolejne przypadki testowe i ewentualnie zwiększyć pokrycie.</p> <p>c) Odpowiedź niepoprawna. Sformułowanie „zwiększenie czasu wykonywania testów dzięki większej mocy obliczeniowej” zawiera sprzeczność, ponieważ ze wzrostem mocy obliczeniowej zwykle wiąże się skrócenie czasu wykonywania testów. Zwiększenie czasu nie jest korzyścią – testowanie w takim wypadku trwa dłużej.</p> <p>d) Odpowiedź poprawna. Zapobieganie błędom ludzkim poprzez zwiększenie spójności i powtarzalności jest jedną z korzyści automatyzacji testów, ponieważ w testach automatycznych nie mogą wystąpić tego rodzaju problemy. Oznacza to na przykład, że testy są wyprowadzane w spójny sposób z wymagań, dane testowe są tworzone w systematyczny sposób, a testy są wykonywane przy użyciu narzędzia w tej samej kolejności i z tą samą częstotliwością.</p>	FL-6.2.1	K1	1