

Egzamin przykładowy – odpowiedzi

Zbiór B

wersja 1.6

(wersja PL 1.0.0.5)

Sylabus ISTQB® Certyfikowany tester – poziom podstawowy

zgodny z syllabusem w wersji 4.0

International Software Testing Qualifications Board



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 1.4 i 1.5: Monika Petri-Starego, Adam Roman.

Aktualizacja do wersji 1.6: 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.

Certyfikowany tester — poziom podstawowy

Zbiór B

Egzamin przykładowy — odpowiedzi



Odpowiedzialność za dokument

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

Obsługą dokumentu zajmuje się podstawowy zespół ISTQB®, w skład którego wchodzą przedstawiciele Grupy roboczej ds. sylabusa 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. sylabusa i Radom Krajowym za ich sugestie i wskazówki.

Historia zmian

Wersja	Data	Uwagi
1.6	08.07.2024 r.	Poprawka pytania nr 32.
1.5	29.05.2024 r.	Drobna poprawka w pytaniu nr 28.
1.4	12.04.2024 r.	Poprawka odpowiedzi do pytania nr 2.
1.3	8.01.2024 r.	Poprawki zgodnie z dokumentem pytań.
1.2	5.12.2023 r.	Korekta celu nauczania dla pytania nr 15.
1.1	15.11.2023 r.	Dopasowanie odpowiedzi do zmian w pytaniu 35.
1.0	16.10.2023 r.	Pierwsza wersja.

Historia zmian polskiej wersji

Wersja	Data	Uwagi
1.0.0.5	29.10.2024 r.	Poprawka pytania nr 32 – poprawna odpowiedź c, zamiana uzasadnienia odpowiedzi „b” z „c”.
1.0.0.4	23.10.2024 r.	Poprawka pytania nr 32 (oznaczenie poprawnej odpowiedzi w kolumnie „poprawna odpowiedź” i w tabeli z kluczem odpowiedzi).
1.0.0.3	28.08.2024 r.	Poprawka pytania nr 32.
1.0.0.2	17.06.2024 r.	Aktualizacja do wersji 1.5 dokumentu. Drobna poprawka w pyt. 28.
1.0.0.1	23.05.2024 r.	Aktualizacja do wersji 1.4 dokumentu. Poprawki zgodnie z dokumentem pytań. Poprawki do pytania nr 2, 35, korekta celu nauczania dla pytania nr 15.
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

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	d	FL-1.2.1	K2	1
2	b	FL-1.2.2	K1	1
3	d	FL-1.3.1	K2	1
4	a	FL-1.4.1	K2	1
5	c	FL-1.4.2	K2	1
6	b	FL-1.4.4	K2	1
7	b	FL-1.5.1	K2	1
8	d	FL-1.5.2	K1	1
9	b	FL-2.1.1	K2	1
10	b	FL-2.1.2	K1	1
11	a	FL-2.1.3	K1	1
12	b	FL-2.1.4	K2	1
13	a	FL-2.2.1	K2	1
14	d	FL-2.2.3	K2	1
15	b	FL-3.1.3	K2	1
16	c	FL-3.2.1	K1	1
17	d	FL-3.2.2	K2	1
18	c	FL-3.2.3	K1	1
19	d	FL-4.1.1	K2	1
20	a	FL-4.2.1	K3	1

Nr pytania	Poprawna odpowiedź	Cel nauczania	Poziom wiedzy	Punkty
21	d	FL-4.2.2	K3	1
22	b	FL-4.2.3	K3	1
23	c	FL-4.2.4	K3	1
24	b	FL-4.3.1	K2	1
25	c	FL-4.3.2	K2	1
26	a, e	FL-4.4.2	K2	1
27	d	FL-4.4.3	K2	1
28	b	FL-4.5.2	K2	1
29	d	FL-4.5.3	K3	1
30	a	FL-5.1.3	K2	1
31	b	FL-5.1.4	K3	1
32	c	FL-5.1.5	K3	1
33	d	FL-5.1.7	K2	1
34	c	FL-5.2.4	K2	1
35	a	FL-5.3.1	K1	1
36	a	FL-5.3.3	K2	1
37	a	FL-5.4.1	K2	1
38	b	FL-5.5.1	K3	1
39	c	FL-6.1.1	K2	1
40	a	FL-6.2.1	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
1	d	<p>a) Odpowiedź niepoprawna. Często możliwe jest zastosowanie testowania dynamicznego do spowodowania awarii przedmiotu testów w sposób nieosiągalny dla użytkowników, np. za pomocą wstrzykiwania błędów. Jednak jeśli dana awaria nie może nigdy wystąpić w przypadku korzystania z produktu przez rzeczywistych użytkowników końcowych, wykrycie takich sytuacji nie ma istotnego znaczenia, ponieważ ostatecznym celem testowania jest poprawa jakości produktu pracy z punktu widzenia użytkowników końcowych. Poświęcenie czasu na testowanie pod kątem awarii, które nie mogą wystąpić u rzeczywistych użytkowników, to niezbyt efektywny sposób wykorzystania pracy testerów.</p> <p>b) Odpowiedź niepoprawna. Testowanie statyczne jest formą analizy statycznej stosowaną przez programistów w celu wykrycia defektów w kodzie programu wcześniej niż można to osiągnąć za pomocą testowania dynamicznego. Należy jednak zauważyć, że testowanie statyczne (i analiza statyczna) służy do wykrywania defektów, a nie awarii, które są wykrywane podczas testowania dynamicznego. Zatem z powodu użycia określenia „awarie” odpowiedź należy uznać za niepoprawną.</p> <p>c) Odpowiedź niepoprawna. Analiza statyczna bezpośrednio wykrywa w kodzie anomalie, które mogą być defektami. Działania takie wykonuje się zwykle na potrzeby programistów, a nie klientów. Sformułowanie o potwierdzeniu poprzez analizę statyczną możliwości oddania do eksploatacji elementów, które nie udostępniają danych wyjściowych, jest pozbawione sensu.</p> <p>d) Odpowiedź poprawna. Przeglądy to forma testowania statycznego, którą można zastosować już na samym początku cyklu wytwarzania oprogramowania. Wykorzystuje się je do znalezienia defektów, które da się usunąć wcześniej, dzięki czemu nie trzeba niepotrzebnie poświęcać czasu na realizację niepoprawnych wymagań. Jeśli defekty nie zostaną wcześnie wykryte i usunięte, to po ich znalezieniu konieczne będzie wprowadzenie zmian w pochodnych produktach</p>	FL-1.2.1	K2	1

		pracy, np. w projekcie lub w kodzie, ponieważ będą one oparte na błędnych wymaganiach.			
--	--	--	--	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
2	b	<p>a) Odpowiedź niepoprawna. Zapewnienie jakości koncentruje się na doskonaleniu i wdrażaniu procesów, stosując podejście prewencyjne w celu uniknięcia błędów i defektów, podczas gdy testowanie jest formą kontroli jakości, która służy do wykrywania defektów.</p> <p>b) Odpowiedź poprawna. Kontrola jakości ma na celu osiągnięcie odpowiednich poziomów jakości poprzez skupienie się na identyfikacji i korygowaniu defektów produktu. Testowanie jest istotną częścią kontroli jakości i pomaga odkryć te defekty.</p> <p>c) Odpowiedź niepoprawna. Chociaż testowanie jest istotną częścią kontroli jakości i pomaga odkryć defekty, inne (nietestowe) techniki wykorzystywane w kontroli jakości obejmują metody formalne, takie jak sprawdzanie modelu i dowód poprawności, a także symulację i prototypowanie.</p> <p>d) Odpowiedź niepoprawna. Zapewnienie jakości koncentruje się na doskonaleniu i wdrażaniu procesów, stosując podejście prewencyjne w celu uniknięcia błędów i defektów, podczas gdy testowanie jest formą kontroli jakości, która służy do wykrywania defektów.</p>	FL-1.2.2	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
3	d	<p>Zasada „testowanie gruntowne jest niemożliwe” mówi o tym, że z wyjątkiem najprostszych przypadków nie jest wykonalne przetestowanie każdej możliwej kombinacji wartości wejściowych z uwzględnieniem wszystkich potencjalnych okoliczności. W testach używa się technik testowania, ustalania priorytetów przypadków testowych oraz testowania opartego na ryzyku w celu wyboru pewnej próbki z całego zbioru możliwości i ukierunkowania prac związanych z testowaniem.</p> <p>a) Odpowiedź niepoprawna. Zasada mówi, że z wyjątkiem najprostszych przypadków nie jest możliwe przetestowanie wszystkiego. Przetestowanie wszystkiego oznaczałoby konieczność sprawdzenia każdej możliwej kombinacji danych wejściowych do testów z uwzględnieniem wszystkich potencjalnych okoliczności. Jest to zwykle niewykonalne, gdyż może oznaczać praktycznie nieskończoną liczbę przypadków. Przetestowanie wszystkich możliwych danych wejściowych nie rozwiąże tego problemu, ponieważ relacje między danymi wejściowymi i określonymi danymi wyjściowymi mogą być różne dla różnych przedmiotów testów. Czasami liczba możliwych danych wyjściowych może być praktycznie nieskończona (np. gdy występuje kilka zmiennych reprezentujących liczby rzeczywiste); z kolei w innych sytuacjach mogą występować tylko dwie konkretne wartości wyjściowe, np. w przypadku jednej zmiennej, która może przybierać wartości „prawda” lub „fałsz”.</p> <p>b) Odpowiedź niepoprawna. Zasada mówi o tym, że nie jest wykonalne przetestowanie każdej możliwej kombinacji wartości wejściowych z uwzględnieniem wszystkich potencjalnych okoliczności. Wynika to z faktu, że w przypadku nietrywalnych systemów liczba kombinacji może być praktycznie nieskończona. Dlatego udokumentowanie wszystkich możliwych kombinacji danych wejściowych do testów byłaby niepraktyczne, ponieważ zajęłoby nieskończonie wiele czasu.</p>	FL-1.3.1	K2	1

		<p>c) Odpowiedź niepoprawna. Rozpoczynanie testów na jak najwcześniejszym etapie z wykorzystaniem przeglądów i innych podejść do testowania statycznego nie rozwiązuje problemu wynikającego ze zbyt dużej liczby możliwych przypadków testowych. Zasada „wczesne testowanie oszczędza czas i pieniądze” dotyczy usuwania defektów na wcześnieym etapie w celu uniknięcia ich wystąpienia w pochodnych produktach pracy, co pozwala obniżyć koszty i zmniejszyć prawdopodobieństwo awarii.</p> <p>d) Odpowiedź poprawna. Zastosowanie podziału na klasy równoważności oraz analizy wartości brzegowych do generowania przypadków testowych to jeden ze sposobów uwzględnienia omawianej zasady, ponieważ takie techniki testowania stanowią systematyczny sposób wyprowadzania skończonego podzbioru wszystkich przypadków testowych.</p>		
--	--	---	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
4	a	<p>a) Odpowiedź poprawna. W projektowaniu testów używa się warunków testowych do tworzenia przypadków testowych i innych niezbędnych testaliów, np. wymagań dotyczących danych testowych i kart opisu testu na potrzeby testowania eksploracyjnego. Wymagania dotyczące środowiska testowego również są specyfikowane, z uwzględnieniem niezbędnnej infrastruktury i narzędzi.</p> <p>b) Odpowiedź niepoprawna. Wykonywanie testów obejmuje wykonywanie przypadków testowych (w ramach procedur testowych), jednak bezpośrednio nie uwzględnia innych testaliów wymienionych w pytaniu, np. wymagań dotyczących danych testowych, wymagań dotyczących środowiska testowego i warunków testowych.</p> <p>c) Odpowiedź niepoprawna. Analiza testów służy do określenia cech wymagających przetestowania. Podstawa testów jest analizowana i definiowana w postaci warunków testowych, którym następnie przypisywane są priorytety wraz z powiązanymi ryzykami. Analiza obejmuje pracę z warunkami testowymi, jednak nie uwzględnia innych testaliów wymienionych w pytaniu, np. wymagań dotyczących danych testowych, wymagań dotyczących środowiska testowego i przypadków testowych.</p> <p>d) Odpowiedź niepoprawna. Implementacja testów obejmuje generowanie procedur testowych, np. skryptów testów manualnych i automatycznych tworzonych na podstawie przypadków testowych i potencjalnie łączonych w zestawy testowe. Procedury testowe są szeregowane według priorytetów i porządkowane w ramach harmonogramu wykonywania testów. Następnie tworzy się dane testowe i buduje środowisko testowe, którego konfiguracja podlega zweryfikowaniu. W ramach implementacji bezpośrednio pracujemy z przypadkami testowymi, możemy również korzystać z wymagań dotyczących danych testowych i wymagań dotyczących</p>	FL-1.4.1	K2	1

		środowiska testowego w celu utworzenia takich danych i środowiska, jednak nie uwzględniamy warunków testowych.			
--	--	--	--	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
5	c	<p>a) Odpowiedź niepoprawna. Dział marketingu organizacji raczej nie będzie wykonywał zbyt wielu testów (choć w przypadku niektórych organizacji pracownicy tego działu mogą uczestniczyć w testowaniu akceptacyjnym), więc średni poziom ich doświadczenia (na ogół w obszarze marketingu) raczej nie wpłynie na sposób testowania danego przedmiotu testów.</p> <p>b) Odpowiedź niepoprawna. Poziom wiedzy użytkowników o budowaniu dla nich nowego systemu raczej nie wpłynie na sposób prowadzenia testów. Ewentualny udział użytkowników, który może wpływać na sposób wykonywania testów, będzie raczej wynikać z decyzji podjętych przez testerów, klienta i kierownika projektu.</p> <p>c) Odpowiedź poprawna. Liczba lat doświadczenia członków zespołu przeprowadzającego testy pozwala określić ich umiejętności i wiedzę (np. znajomość różnych narzędzi i typów defektów), które mogą zostać wykorzystane podczas testowania.</p> <p>d) Odpowiedź niepoprawna. Struktura organizacyjna będzie się różnić dla różnych użytkowników końcowych. Użytkownicy mogą nawet nie wiedzieć, że aplikacja aktualnie podlega testom, więc struktura organizacyjna, w której działa użytkownik końcowy, ma niewielki wpływ na sposób wykonywania testów.</p>	FL-1.4.2	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
6	b	<p>a) Odpowiedź niepoprawna. Śledzenie powiązań między złagodzonymi ryzykami a zaliczonymi przypadkami testowymi nie daje zbyt wielu informacji, ponieważ aby ryzyka zostały złagodzone (za pomocą testowania), i tak muszą z nimi wiązać się odpowiednie zaliczone przypadki testowe. Jeśli mamy mieć możliwość oceny ryzyka rezydualnego, musi być dostępne śledzenie powiązań między wszystkimi ryzykami a wynikami testów, tak aby ryzyka, dla których nie ma odpowiednich zaliczonych testów, można było zidentyfikować jako ryzyka rezydualne.</p> <p>b) Odpowiedź poprawna. Śledzenie powiązań między wymaganiami użytkowników a rezultatami wykonania testów pozwala określić, które wymagania użytkowników zostały do tej pory przetestowane, a zatem mierzyć postępy w realizacji projektu (w kontekście testowania) w odniesieniu do celów biznesowych.</p> <p>c) Odpowiedź niepoprawna. Nie jest wcale oczywiste, że niezaliczone przypadki testowe w większym stopniu świadczą o umiejętnościach testerów niż przypadki zaliczone. W pewnym zakresie zależy to od celu danego testu (np. może nim być budowanie zaufania do jakości przedmiotu testów albo powodowanie awarii). Taki pomiar jakości pracy testerów oparty na liczbie zaliczonych i niezaliczonych przypadków testów może poza tym przynieść efekty przeciwnie do zamierzonego, ponieważ testerzy mogą zacząć optymalizować swoje działania na podstawie tej metryki, a nie na podstawie celu testu.</p> <p>d) Odpowiedź niepoprawna. Śledzenie powiązań między zidentyfikowanymi ryzykami a zapisanymi warunkami testowymi pozwala określić warunki testowe, które należy jeszcze zapisać. Ustalenie, dla których ryzyk warto wykonać testy, jest częścią zarządzania ryzykiem, a konkretnie działań związanych z łagodzeniem ryzyka.</p>	FL-1.4.4	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
7	b	<p>a) Odpowiedź niepoprawna. Dobre umiejętności komunikacyjne oraz umiejętność aktywnego słuchania i pracy w zespole pozwalają testerom sprawnie komunikować się ze wszystkimi interesariuszami. Głęboka znajomość rynku gier komputerowych nie jest jednak przykładem ogólnej umiejętności istotnej dla testerów.</p> <p>b) Odpowiedź poprawna. Wiedza merytoryczna pozwalająca zrozumieć użytkowników i przedstawicieli jednostek biznesowych oraz sprawnie się z nimi porozumiewać jest jedną z ogólnych umiejętności, które powinni opanować testerzy. Tester będący doświadczonym pilotem może lepiej ocenić kryteria akceptacji systemu sterowania śmigłowcem.</p> <p>c) Odpowiedź niepoprawna. Chociaż kompetencje programistyczne można uznać za wiedzę techniczną pozwalającą zwiększyć efektywność testowania podczas korzystania z niektórych narzędzi testowych, jest mało prawdopodobne, że takie umiejętności usprawnią komunikację z analitykami biznesowymi.</p> <p>d) Odpowiedź niepoprawna. Staranność, ostrożność, ciekawość, dbałość o szczegóły i podejście metodyczne (niezbędne do identyfikowania defektów, zwłaszcza defektów trudnych do wykrycia) są rzeczywiście przydatnymi dla testera umiejętnościami ogólnymi. Jest jednak wątpliwe, że tester zajmie się generowaniem przypadków testowych przed rozpoczęciem testowania eksploracyjnego, jednym z założeń tej techniki jest bowiem generowanie przypadków testowych w trakcie testowania, a nie zapisywanie ich na wcześniejszym etapie.</p>	FL-1.5.1	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
8	d	<p>a) Odpowiedź niepoprawna. W ramach podejścia „cały zespół” każdy członek zespołu, który dysponuje niezbędną wiedzą i umiejętnościami, może wykonywać dowolne zadania, nie oznacza to jednak, że każdy może przyjmować w dowolnym momencie dowolną rolę. Zwykle członkowie zespołu przyjmują role, do których wypełniania mają odpowiednie kompetencje. Nie oczekuje się również, że każdy członek zespołu będzie w stanie realizować dowolną rolę.</p> <p>b) Odpowiedź niepoprawna. Podejście „cały zespół” dotyczy pracy jednego zespołu (zwykle w przypadku zwinnego wytwarzania oprogramowania), jednak nie opisuje sposobu pracy wielu zespołów w większych projektach. Nie sugeruje również, że tylko jeden zespół jest niezbędny do realizacji całego projektu.</p> <p>c) Odpowiedź niepoprawna. W podejściu „cały zespół” nie oczekuje się, że wszyscy członkowie zespołu będą uczestniczyć w podejmowaniu każdej decyzji. Nie ma na przykład potrzeby, żeby przedstawiciel jednostki biznesowej (np. właściciel produktu) uczestniczył w każdej decyzji technicznej, która nie ma wpływu na cele biznesowe. Zastosowanie takiego podejścia niepotrzebnie spowolniłoby pracę zespołu.</p> <p>d) 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>	FL-1.5.2	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
9	b	<p>a) Odpowiedź niepoprawna. W przypadku zwinnego wytwarzania oprogramowania produkty powstają w każdej iteracji, a częste dostarczanie wersji przyrostowych wymaga szeroko zakrojonego testowania regresji. Niektóre lub wszystkie testy regresji mogą zostać zautomatyzowane, jednak automatyzacja testów systemowych nie jest w stanie zastąpić testowania regresji (niezależnie od tego, czy podlega ono automatyzacji).</p> <p>b) Odpowiedź poprawna. Jeśli stosowany jest sekwencyjny model wytwarzania, w początkowych fazach cyklu życia nie jest dostępny kod przeznaczony do wykonania, dlatego w tym czasie wykonuje się testowanie statyczne (np. przeglądy). W późniejszych fazach, kiedy taki kod staje się dostępny, możliwe jest testowanie dynamiczne. Należy jednak pamiętać, że przygotowanie do testowania dynamicznego często odbywa się już we wczesnych fazach dowolnego cyklu wytwarzania oprogramowania.</p> <p>c) Odpowiedź niepoprawna. Jeśli stosowany jest iteracyjny model wytwarzania, np. w przypadku zwinnego wytwarzania oprogramowania, testy modułowe mogą być dobrym sposobem testowania regresji w każdej iteracji. W takiej sytuacji wiele przemawia za automatyzacją testów modułowych, ponieważ muszą one być często uruchamiane. Mało prawdopodobne, żeby istniało poważne uzasadnienie konieczności manualnego wykonywania testów modułowych przez programistów.</p> <p>d) Odpowiedź niepoprawna. W większości przyrostowych modeli wytwarzania produkty powstają w każdym przyroście, co wymaga zastosowania zarówno testowania statycznego, jak i dynamicznego na wszystkich poziomach testów dla każdego dostarczanego przyrostu.</p>	FL-2.1.1	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
10	b	<p>a) Odpowiedź niepoprawna. Testerzy powinni przeprowadzić przegląd produktów pracy natychmiast po udostępnieniu wersji roboczych tych produktów. Pozwala to wykonać wcześniejsze testowanie w ramach podejścia „przesunięcie w lewo”. Jeśli czekaliby z tym do następnej fazy wytwarzania, pojawiłaby się możliwość niepotrzebnego rozpoczęcia prac programistycznych (oraz związanych z testowaniem) opartych na wadliwych, niepodanych przeglądом produktach pracy.</p> <p>b) Odpowiedź poprawna. Testerzy powinni przeprowadzić przegląd produktów pracy natychmiast po udostępnieniu wersji roboczych tych produktów. Pozwala to wykonać wcześniejsze testowanie w ramach podejścia „przesunięcie w lewo”.</p> <p>c) Odpowiedź niepoprawna. Testerzy zwykle prowadzą przeglądy produktów pracy stanowiących podstawę testów w ramach analizy testów, a nie przed rozpoczęciem analizy i projektowania testów.</p> <p>d) Odpowiedź niepoprawna. Testerzy powinni przeprowadzić przegląd produktów pracy natychmiast po udostępnieniu wersji roboczych tych produktów. Pozwala to wykonać wcześniejsze testowanie w ramach podejścia „przesunięcie w lewo”. Gdyby testerzy czekali na publikację, do opublikowanej wersji dokumentu mogłyby trafić defekty, które udałoby się wcześniej znaleźć w trakcie przeglądu.</p>	FL-2.1.2	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
11	a	<p>a) Odpowiedź poprawna. Wytwarzanie sterowane testami (TDD) to ogólnie znany przykład podejścia typu „najpierw test” do wytwarzania oprogramowania.</p> <p>b) Odpowiedź niepoprawna. Wytwarzanie sterowane pokryciem nie jest poprawnym przykładem podejścia typu „najpierw test” do wytwarzania oprogramowania.</p> <p>c) Odpowiedź niepoprawna. Wytwarzanie sterowane jakością nie jest poprawnym przykładem podejścia typu „najpierw test” do wytwarzania oprogramowania.</p> <p>d) Odpowiedź niepoprawna. Wytwarzanie sterowane cechami nie jest przykładem podejścia typu „najpierw test” do wytwarzania oprogramowania. To zwinna metodyka wytwarzania oprogramowania oparta na realizacji pewnych cech (w odróżnieniu od historyjek użytkownika w metodyce Scrum).</p>	FL-2.1.3	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
12	b	<p>a) Odpowiedź niepoprawna. Metodyka DevOps wzbogaca testowanie na kilka sposobów, m.in. poprzez szybkie otrzymywanie informacji zwrotnych na temat jakości kodu, zmniejszenie ryzyka związanego z regresją dzięki automatycznym testom regresji oraz promowanie podejścia „przesunięcie w lewo” opierającego się na dostarczaniu wysokiej jakości kodu i wykonywaniu testów modułowych. Odbywa się to głównie dzięki ciągłej integracji, w ramach której programiści dostarczają testy modułowe (jednostkowe) wraz z nowym kodem. Kod musi pomyślnie przejść takie testy, żeby mógł zostać uwzględniony podczas budowania wersji. Dlatego programiści muszą wykonywać testowanie modułowe.</p> <p>b) Odpowiedź poprawna. Metodyka DevOps wzbogaca testowanie na kilka sposobów, m.in. poprzez szybkie otrzymywanie informacji zwrotnych na temat jakości kodu, zmniejszenie ryzyka związanego z regresją dzięki automatycznym testom regresji oraz promowanie podejścia „przesunięcie w lewo” opierającego się na dostarczaniu wysokiej jakości kodu i wykonywaniu testów modułowych.</p> <p>c) Odpowiedź niepoprawna. Metodyka DevOps wzbogaca testowanie na kilka sposobów, m.in. poprzez szybkie otrzymywanie informacji zwrotnych na temat jakości kodu, zmniejszenie ryzyka związanego z regresją dzięki automatycznym testom regresji oraz promowanie podejścia „przesunięcie w lewo” opierającego się na dostarczaniu wysokiej jakości kodu i wykonywaniu testów modułowych. Testerzy nie próbują jednakowo traktować programistów i osób odpowiedzialnych za eksploatację, poświęcając więcej czasu na testowanie wydań, chociaż podejście „przesunięcie w prawo” (testowanie w środowisku produkcyjnym) może być stosowane.</p> <p>d) Odpowiedź niepoprawna. W metodyce DevOps zautomatyzowane procesy takie jak ciągła integracja i ciągłe dostarczanie ułatwiają tworzenie stabilnych środowisk testowych i ograniczają konieczność wykonywania testów manualnych. Nie należy</p>	FL-2.1.4	K2	1

		jednak lekceważyć znaczenia testów manualnych, zwłaszcza z perspektywy użytkownika.			
--	--	---	--	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
13	a	<p>a) Odpowiedź poprawna. Testowanie systemowe bada zachowanie i możliwości całego systemu i obejmuje testowanie niefunkcjonalne charakterystyk jakościowych, m.in. testy zabezpieczeń. Ten typ testowania często jest wykonywany przez niezależny zespół testowy na podstawie specyfikacji systemu.</p> <p>b) Odpowiedź niepoprawna. Interfejsy łączące system podlegający testowaniu z innymi systemami oraz usługami zewnętrznymi sprawdzane są w ramach testowania integracji systemów.</p> <p>c) Odpowiedź niepoprawna. Testowanie beta to typ testowania akceptacyjnego wykonywanego w zewnętrznym ośrodku przez osoby występujące w rolach spoza organizacji twórczej.</p> <p>d) Odpowiedź niepoprawna. Interfejsy i interakcje między modułami systemu, np. interfejsem użytkownika i bazą danych, sprawdzane są w ramach testowania integracji modułów.</p>	FL-2.2.1	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
14	d	<p>a) Odpowiedź niepoprawna. Liczba testów regresji rośnie w miarę postępów w realizacji projektu, ponieważ zwykle po wprowadzeniu zmian w systemie wymagane są nowe testy regresji. Na ogół wzrasta również liczba testów potwierdzających, ponieważ potrzebne są testy dla każdej nowej poprawki wprowadzonej w systemie.</p> <p>b) Odpowiedź niepoprawna. Występuje odwrotna sytuacja. Testy potwierdzające są tworzone i uruchamiane po wprowadzeniu poprawek w przedmiocie testów, natomiast testy regresji są uruchamiane (w idealnej sytuacji) po wprowadzeniu udoskonaleń (zmian) w przedmiocie testów.</p> <p>c) Odpowiedź niepoprawna. Podczas testowania potwierdzającego sprawdzamy, czy defekt został poprawnie usunięty, zatem rzeczywiście polega ono na testowaniu zmian w przedmiocie testów. Jednak testowanie regresji wiąże się ze sprawdzeniem, czy wprowadzone zmiany (w tym zmiany w środowisku operacyjnym) nie spowodowały negatywnych konsekwencji w oprogramowaniu, które nie było modyfikowane. Nie chodzi w nim zatem o sprawdzenie, czy nie zmieniło się środowisko operacyjne.</p> <p>d) Odpowiedź poprawna. Testowanie regresji pozwala sprawdzić, czy wprowadzone zmiany nie spowodowały negatywnych konsekwencji w oprogramowaniu, które nie było modyfikowane. Podczas testowania potwierdzającego sprawdzamy, czy defekt został usunięty, zatem rzeczywiście polega ono na testowaniu zmian w zmodyfikowanym kodzie.</p>	FL-2.2.3	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
15	b	<p>a) Odpowiedź niepoprawna. Brak użyteczności interfejsu użytkownika można wykryć w trakcie przeglądu za pomocą odpowiedniej listy kontrolnej, jednak można to również zrobić za pomocą dynamicznych testów interfejsu prowadzonych przez kilku typowych użytkowników, którzy przekażą informacje zwrotne na temat jego użyteczności.</p> <p>b) Odpowiedź poprawna. W trakcie przeglądu kodu można wykryć fragmenty kodu, do których nie prowadzi żadna ścieżka, natomiast w testach dynamicznych można sprawdzić tylko kod wykonywalny i nie można określić, że kod nie może zostać osiągnięty bez uruchomienia każdej możliwej kombinacji wejść i stanów wejściowych, co jest niepraktyczne w przypadku rzeczywistego kodu.</p> <p>c) Odpowiedź niepoprawna. Bez uruchamiania kodu, czyli za pomocą samego testowania statycznego, trudno wykryć, że w przypadku większości oczekiwanych użytkowników systemu występuje długi czas odpowiedzi. Zatem w tej sytuacji defekt można wykryć w testach dynamicznych, ale raczej nie w testach statycznych.</p> <p>d) Odpowiedź niepoprawna. W trakcie przeglądu kodu prowadzonego przez osobę, która zna wymagane cechy systemu, można wykryć, że takie cechy nie zostały zaimplementowane. Można to jednak wykryć również podczas testów dynamicznych.</p>	FL-3.1.3	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
16	c	<p>a) Odpowiedź niepoprawna. Informacje zwrotne pochodzą od interesariuszy (np. użytkowników końcowych i przedstawicieli jednostek biznesowych), a nie od programistów, nie należy więc oczekiwac, że dzięki nim kierownicy zdobędą wiedzę o mniejszej bądź większej produktywności programistów.</p> <p>b) Odpowiedź niepoprawna. Wczesne i często otrzymywane informacje zwrotne od interesariuszy nie służą kierownikom projektów do określania priorytetów interakcji z różnymi interesariuszami.</p> <p>c) Odpowiedź poprawna. Informacje zwrotne od interesariuszy otrzymywane w procesie wytwarzania oprogramowania w odpowiednim czasie i z odpowiednią częstotliwością mogą przynieść wiele korzyści, ponieważ możliwe staje się wtedy wczesne sygnalizowanie potencjalnych problemów związanych z jakością. Ponadto pozwala to zapobiec ewentualnym nieporozumieniom w kwestii wymagań, a także umożliwia szybsze analizowanie zmian w wymaganiach interesariuszy i realizację takich wymagań.</p> <p>d) Odpowiedź niepoprawna. Dzięki odpowiednio wcześnie i odpowiednio często przekazywanym informacjom zwrotnym można zapobiec powstaniu produktu, który nie spełnia potrzeb interesariuszy i wymaga kosztownych poprawek, co wiązałoby się z niedotrzymaniem terminów. W idealnej sytuacji nie powinno być zatem opóźnień. Ponadto informacje przekazywane są przez interesariuszy, a nie interesariuszom. Wiedza użytkowników końcowych nie zwiększy się dzięki przekazaniu przez nich informacji.</p>	FL-3.2.1	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
17	d	<p>W odniesieniu do poszczególnych zadań z podanej listy:</p> <ol style="list-style-type: none"> 1. Wybór charakterystyk jakościowych poddawanych ocenie oraz kryteriów wyjścia – planowanie (C): w fazie planowania określa się zakres przeglądu, cel przeglądu, produkt pracy będący jego przedmiotem, oceniane charakterystyki jakościowe, obszary wymagające szczególnej uwagi, kryteria wyjścia, informacje pomocnicze (np. standardy), nakład pracy oraz ramy czasowe. 2. Dostęp do produktu pracy dla wszystkich uczestników – rozpoczęcie przeglądu (B): sprawdzenie, czy każdy uczestnik ma dostęp do produktu pracy będącego przedmiotem przeglądu oraz niezbędnych zasobów, zna swoją rolę i swoje obowiązki. 3. Zidentyfikowanie anomalii w produkcie pracy – przegląd indywidualny (A): ocena jakości produktu pracy będącego przedmiotem przeglądu oraz zidentyfikowanie ewentualnych anomalii, zaleceń i pytań, z wykorzystaniem różnych technik przeglądu (takich jak przegląd oparty na liście kontrolnej czy przegląd oparty na scenariuszach). 4. Analiza i omówienie anomalii – przekazanie informacji i analiza (D): przeanalizowanie i omówienie każdej anomalii, określenie jej statusu, wyznaczenie osoby odpowiedzialnej i wskazanie wymaganych działań; podjęcie decyzji związanych z przeglądem, zwykle podczas spotkania. Może się również okazać, że niezbędne będzie przeprowadzenie kolejnego przeglądu. <p>Zatem POPRAWNE powiązanie to 1C, 2B, 3A, 4D, czyli POPRAWNA odpowiedzią jest odpowiedź d).</p>	FL-3.2.2	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
18	c	<p>W odniesieniu do poszczególnych ról z podanej listy:</p> <ol style="list-style-type: none"> 1. Protokolant (zwany także rejestrującym) – jest odpowiedzialny za gromadzenie informacji przekazanych przez przeglądających i protokołowanie informacji związanych z przeglądkiem, w tym informacji o podjętych decyzjach oraz o nowych anomaliiach zidentyfikowanych podczas spotkania związanego z przeglądkiem. (Rejestruje informacje związane z przeglądkiem, w tym informacje o podjętych decyzjach oraz o nowych anomaliiach stwierdzonych w trakcie spotkania związanego z przeglądkiem – B). 2. Lider przeglądu – jest odpowiedzialny za ogólny nadzór nad procesem przeglądu, m.in. wybór uczestników przeglądu, zaplanowanie spotkań i zadanie o skuteczne przeprowadzenie przeglądu. (Ponosi ogólną odpowiedzialność za przegląd, w tym określa miejsce i termin przeglądu – D). 3. Facylitator (zwany także moderatorem) – jest odpowiedzialny za sprawny przebieg spotkań związanych z przeglądkiem. Występuje w roli mediatora, zarządza czasem oraz zapewnia bezpieczne warunki, w których każdy uczestnik przeglądu może swobodnie wyrażać swoje zdanie. (Dba o sprawny przebieg spotkań związanych z przeglądkiem i zapewnia bezpieczne środowisko przeglądu – A). 4. Kierownik – jest odpowiedzialny za określenie przedmiotu przeglądu i przydział zasobów, m.in. wyznaczenie pracowników oraz określenie ram czasowych przeglądu. (Decyduje o tym, co ma być przedmiotem przeglądu, a także udostępnia niezbędne zasoby, m.in. wyznacza pracowników oraz określa ramy czasowe przeglądu – C). 	FL-3.2.3	K1	1

		Zatem POPRAWNE powiązanie to 1B, 2D, 3A, 4C, czyli POPRAWNĄ odpowiedzią jest odpowiedź c).			
--	--	--	--	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
19	d	<p>a) Odpowiedź niepoprawna. Testowanie w oparciu o tablicę decyzyjną jest czarnoskrzynkową techniką testowania, zatem techniką opartą na specyfikacji, a nie na strukturze. Przypadki testowe nie są tworzone na podstawie decyzji zawartych w kodzie źródłowym. W testowaniu gałęzi przypadki testowe są tworzone na podstawie znajomości przepływu sterowania w przedmiocie testów.</p> <p>b) Odpowiedź niepoprawna. Przewidywanie wystąpień potencjalnych defektów jest wykorzystywane w technice zgadywania błędów (która jest techniką opartą na doświadczeniu), a nie w testowaniu gałęzi (która jest techniką opartą na strukturze). W testowaniu w oparciu o tablicę decyzyjną przypadki testowe są tworzone na podstawie specyfikacji opisującej logikę biznesową.</p> <p>c) Odpowiedź niepoprawna. Jeśli przypadek testowy jest oparty na znajomości przepływu sterowania w przedmiocie testów, mamy do czynienia z białośkrzynką techniką testowania. Testowanie w oparciu o tablicę decyzyjną zwykle wykorzystuje analizę logiki biznesowej, zatem jest czarnoskrzynkową techniką testowania. W testowaniu gałęzi przypadki testowe nie są wyprowadzane ze specyfikacji – w takiej sytuacji byłaby to czarnoskrzynkowa technika testowania. Testowanie gałęzi jest białośkrzynkową techniką testowania, w której przypadki testowe tworzy się na podstawie struktury kodu źródłowego.</p> <p>d) Odpowiedź poprawna. Testowanie w oparciu o tablicę decyzyjną jest czarnoskrzynkową techniką testowania, więc jest oparte na analizie określonego zachowania przedmiotu testów bez odwoływania się do jego wewnętrznej struktury.</p>	FL-4.1.1	K2	1

		Oznacza to, że przypadki testowe są niezależne od sposobu implementacji oprogramowania. Testowanie gałęzi jest białyoskrzynką techniką testowania, więc przypadki testowe tworzy się na podstawie analizy wewnętrznej struktury i przetwarzania przedmiotu testów. Przypadki testowe są uzależnione od sposobu, w jaki zaprojektowano dane oprogramowanie i opracowano jego kod, w związku z czym można je tworzyć dopiero po zaprojektowaniu lub zaimplementowaniu przedmiotu testów.		
--	--	--	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
20	a	<p>a) Odpowiedź poprawna. Wartość 19 pokrywa klasę „brak upustu”, wartość 20 klasę „50% upustu”, a wartość 30 klasę „10% upustu”. Trzy podane liczby pokrywają wszystkie poprawne klasy równoważności.</p> <p>b) Odpowiedź niepoprawna. Wartości 11 i 12 pokrywają klasę „brak upustu”, a wartość 20 klasę „50% upustu”. Pokryte są zatem dwie z trzech poprawnych klas równoważności.</p> <p>c) Odpowiedź niepoprawna. Wartość 1 pokrywa klasę „brak upustu”, a wartości 10 i 50 klasę „10% upustu”. Klasa „50% upustu” nie jest pokryta, więc w pokryciu uwzględnione są dwie z trzech poprawnych klas równoważności.</p> <p>d) Odpowiedź niepoprawna. Wartości 29 i 31 pokrywają klasę „brak upustu”, a wartości 10 i 30 klasę „10% upustu”. Klasa „50% upustu” nie jest pokryta, więc w pokryciu uwzględnione są dwie z trzech poprawnych klas równoważności.</p>	FL-4.2.1	K3	1
21	d	<p>Dziedzina zmiennej „długość hasła” ma trzy klasy równoważności:</p> <ul style="list-style-type: none"> • zbyt krótkie hasła: {0, 1, ..., 4, 5}; • poprawne hasła: {6, 7, ..., 11, 12}; • zbyt długie hasła: {13, 14, ...}. <p>Jeśli chcemy uzyskać pełne pokrycie dla trójpunktowej analizy wartości brzegowych, musimy przetestować następujące wartości:</p> <p>0, 1, 4, 5, 6, 7, 11, 12, 13, 14.</p>	FL-4.2.2	K3	1

	<p>Ponieważ mamy już pokrycie dla dwupunktowej analizy wartości brzegowych, oznacza to, że przetestowaliśmy już hasła o następującej długości:</p> <p>0, 5, 6, 12 i 13.</p> <p>Oznacza to, że aby przejść z dwupunktowej analizy wartości brzegowych na trójpunktową analizę wartości brzegowych, musimy dodatkowo sprawdzić następujące długości:</p> <p>1, 4, 7, 11 i 14.</p> <p>Z tego powodu POPRAWNA odpowiedzią jest odpowiedź d).</p>		
--	--	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
22	b	<p>W tablicy decyzyjnej jest pięć kolumn. Każdy przypadek testowy pokrywa jedną z nich.</p> <p>PT1 i PT2 pokrywają Regułę 4.</p> <p>PT3 i PT4 pokrywają Regułę 2.</p> <p>PT5 pokrywa Regułę 5.</p> <p>Zatem pięć podanych przypadków testowych pokrywa trzy z pięciu kolumn, czyli pokrycie wynosi $(3/5)*100\% = 60\%$. Z tego powodu POPRAWNA odpowiedzią jest odpowiedź b).</p>	FL-4.2.3	K3	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
23	C	<p style="text-align: center;">Dodaj [N < 2] / N := N + 1</p> <pre> graph LR START((START)) -- "Dodaj / N := 1" --> NIEPELNY((NIEPEŁNY)) NIEPELNY -- "E1" --> NIEPELNY NIEPELNY -- "E2" --> NIEPELNY NIEPELNY -- "E3" --> PELENNY((PEŁNY)) PELENNY -- "E4" --> PELENNY PELENNY -- "E5" --> NIEPELNY PELENNY -- "Usuń / N := N - 1" --> NIEPELNY </pre> <p style="text-align: center;">Usuń [N > 0] / N := N - 1</p> <p>Oznaczmy poszczególne przejścia symbolami E1, ..., E5, jak pokazano na rysunku. Zmienna N oznacza liczbę aktualnie przechowywanych elementów. Każda akcja „Dodaj” zwiększa ją o 1, a każda akcja „Usuń” zmniejsza ją o 1. Zwróćmy uwagę, że jeśli zdarzenie „Dodaj” występuje w stanie NIEPEŁNY, stan zmienia się na PEŁNY jedynie wówczas, gdy N = 2. Jeśli N < 2, system pozostaje w stanie NIEPEŁNY. Jeśli N = 0, nie można wykonać akcji „Usuń”. Podobnie jeśli N = 3, nie można wykonać akcji „Dodaj”.</p> <p>Test a) można zapisać jako E1, E3, E3, E2, E4 (zatem pokrywa 4 z 5 poprawnych przejść, czyli pozwala osiągnąć 80% pokrycia poprawnych przejść).</p> <p>Test b) jest niewykonalny, ponieważ po trzech pierwszych akcjach „Dodaj” system znajdzie się w stanie PEŁNY, a nie ma poprawnych przejść prowadzących ze stanu PEŁNY wywoływanych przez zdarzenie „Dodaj”. Po trzech pierwszych przejściach uzyskujemy pokrycie jedynie 60% poprawnych przejść.</p> <p>Test c) można zapisać jako E1, E2, E4, E5, E3 (zatem pokrywa 5 z 5 poprawnych przejść, czyli pozwala osiągnąć 100% pokrycia poprawnych przejść).</p> <p>Test d) można zapisać jako E1, E2, E4, E5, E4 (zatem pokrywa 4 z 5 poprawnych przejść, czyli pozwala osiągnąć 80% pokrycia poprawnych przejść).</p>	FL-4.2.4	K3	1

		Dlatego odpowiedzią poprawną jest odpowiedź c).			
--	--	---	--	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
24	b	<p>a) Odpowiedź niepoprawna. Pokrycie jest zawsze obliczane jako wskaźnik procentowy liczby pokrytych elementów, dlatego nie może przekraczać 100%.</p> <p>b) Odpowiedź poprawna. Gdyby zbiory instrukcji w testach T1 i T2 były rozłączne, pokrycie zestawu testowego {T1, T2} wyniosłoby 105%, co nie jest możliwe – patrz odpowiedź a). Dlatego co najmniej 5% wykonywalnych instrukcji musiało zostać wykonane zarówno w teście T1, jak i w teście T2.</p> <p>c) Odpowiedź niepoprawna. Pokrycie instrukcji nie mówi nic o liczbie niewykonywalnych instrukcji w kodzie.</p> <p>d) Odpowiedź niepoprawna. Nawet jeśli zestaw testowy daje pełne pokrycie instrukcji, nie wynika z tego pełne pokrycie gałęzi.</p>	FL-4.3.1	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
25	c	<p>Testowanie gałęzi to białośkrzynkowa technika testowania, w której elementami pokrycia są gałęzie. Gałąź oznacza przekazanie sterowania między dwoma wierzchołkami w grafie przepływu sterowania przedstawiającym możliwe sekwencje wykonywania instrukcji kodu źródłowego w przedmiocie testów. Każde przekazanie sterowania może odbywać się albo bezwarunkowo (kod liniowy), albo warunkowo (wynik decyzji). Pokrycie mierzy się jako iloraz liczby gałęzi sprawdzonych za pomocą co najmniej jednego przypadku testowego przezłączną liczbę gałęzi, a uzyskaną wartość wyraża się w procentach. Reasumując:</p> <ul style="list-style-type: none"> a) Odpowiedź niepoprawna. Wynik decyzji to gałąź warunkowa. W testowaniu gałęzi wartość X uwzględnia nie tylko gałęzie warunkowe, ale również bezwarunkowe. b) Odpowiedź niepoprawna. W testowaniu gałęzi uwzględnia się nie tylko gałęzie warunkowe, ale również bezwarunkowe. c) Odpowiedź poprawna. Pokrycie gałęzi mierzy się jako iloraz liczby gałęzi sprawdzonych za pomocą przypadków testowych przezłączną liczbę gałęzi, a uzyskaną wartość wyraża się w procentach. d) Odpowiedź niepoprawna. Zarówno X, jak i Y uwzględniają jedynie gałęzie warunkowe, a nie biorą pod uwagę gałęzi bezwarunkowych. 	FL-4.3.2	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
26	a, e	<p>Testowanie eksploracyjne jest przydatne w przypadku niepełnych lub niewłaściwie sporządzonych specyfikacji bądź w przypadku testowania pod presją czasu. Ponadto może być uzupełnieniem innych, bardziej formalnych technik testowania. Wśród czynników zwiększających skuteczność testowania eksploracyjnego można wymienić doświadczenie i wiedzę merytoryczną testera oraz wysoki poziom niezbędnych umiejętności, takich jak umiejętność analizy, ciekawość i kreatywność. Reasumując:</p> <ul style="list-style-type: none"> a) Odpowiedź poprawna. Testowanie eksploracyjne jest przydatne w przypadku niepełnych lub niewłaściwie sporządzonych specyfikacji bądź w przypadku testowania pod presją czasu. b) Odpowiedź niepoprawna. Testowanie eksploracyjne nie jest czarnoskrzynkową techniką testowania. c) Odpowiedź niepoprawna. Testowanie eksploracyjne jest przydatne w przypadku niewłaściwie sporządzonych specyfikacji. d) Odpowiedź niepoprawna. Umiejętności programistyczne nie mają w zasadzie nic wspólnego z testowaniem eksploracyjnym. e) Odpowiedź poprawna. Wśród czynników zwiększających skuteczność testowania eksploracyjnego można wymienić doświadczenie i wiedzę merytoryczną testera oraz wysoki poziom niezbędnych umiejętności, takich jak umiejętność analizy, ciekawość i kreatywność. 	FL-4.4.2	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
27	d	<p>a) Odpowiedź niepoprawna. Listy kontrolne powinny zawierać warunki testowe, które można zweryfikować. To przykład błędu, a nie warunku testowego. Nawet jeśli tester byłby w stanie na podstawie przykładów błędów opracować potencjalne warunki testowe, przedstawiony opis błędu jest zbyt ogólny.</p> <p>b) Odpowiedź niepoprawna. Listy kontrolne nie powinny zawierać elementów, które nadają się raczej na kryteria wyjścia. Podany element jest przykładem kryterium wyjścia.</p> <p>c) Odpowiedź niepoprawna. Listy kontrolne nie powinny zawierać elementów, które są zbyt ogólne. Podany element ma bardzo ogólny charakter i w zasadzie opisuje ogólny cel testowania.</p> <p>d) Odpowiedź poprawna. To przykład warunku testowego, który może zostać sprawdzony przez człowieka.</p>	FL-4.4.3	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
28	b	<p>a) Odpowiedź niepoprawna. Format ukierunkowany na reguły zawiera takie formaty jak lista weryfikacyjna w punktach lub przypisanie danych wejściowych do danych wyjściowych w formie tabelarycznej. W jawnym sposób określa reguły, których należy przestrzegać. Format Given/When/Then to format ukierunkowany na scenariusze, ponieważ opisuje scenariusz, który należy zweryfikować.</p> <p>b) Odpowiedź poprawna. To format Given/When/Then, który jest ukierunkowany na scenariusze.</p> <p>c) Odpowiedź niepoprawna. Nie ma czegoś takiego jak format kryteriów akceptacji ukierunkowany na produkt.</p> <p>d) Odpowiedź niepoprawna. Nie ma czegoś takiego jak format kryteriów akceptacji ukierunkowany na proces.</p>	FL-4.5.2	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
29	d	<p>a) Odpowiedź niepoprawna. Przypadek testowy jest związany z wyświetlaniem poprzednich zamówień z historii zamówień.</p> <p>b) Odpowiedź niepoprawna. Przypadek testowy jest związany z wyświetlaniem poprzednich zamówień.</p> <p>c) Odpowiedź niepoprawna. Przypadek testowy jest związany z wyświetlaniem poprzednich zamówień z historii zamówień.</p> <p>d) Odpowiedź poprawna. Przypadek testowy jest związany z procesem rejestracji, który nie jest uwzględniony w historyjce użytkownika. Historyjka użytkownika dotyczy wyświetlania poprzednich zamówień.</p>	FL-4.5.3	K3	1
30	a	<p>a) Odpowiedź poprawna. To warunek, który można (i należy) sprawdzić przed wprowadzeniem kodu do systemu kontroli wersji.</p> <p>b) Odpowiedź niepoprawna. To kryterium można sprawdzić po wykonaniu kroku (2), ponieważ raport o konfliktach scalania można utworzyć dopiero po wprowadzeniu i scaleniu kodu.</p> <p>c) Odpowiedź niepoprawna. Opisany warunek nadaje się jako kryterium wejścia dla kroku (3).</p> <p>d) Odpowiedź niepoprawna. Opisany warunek nadaje się jako kryterium wyjścia dla kroku (3).</p>	FL-5.1.3	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
31	b	<p>Średnia wartość pracochłonności wytwarzania wynosi 900 000 USD, a pracochłonności testowania 90 000 USD (wartości obliczone na podstawie czterech projektów).</p> <p>Średni wskaźnik pracochłonności testowania do wytwarzania wynosi 1:10 (90 000 USD : 900 000 USD), a zatem średnia pracochłonność testowania w projekcie wynosi 10% pracochłonności wytwarzania.</p> <p>Jeśli zatem koszt prac twórczych jest szacowany na 800 000 USD, to koszt testowania można oszacować jako:</p> $10\% * 800\,000 \text{ USD} = 0,1 * 800\,000 \text{ USD} = 80\,000 \text{ USD}$ <p>Dlatego POPRAWNA odpowiedzią jest odpowiedź b).</p>	FL-5.1.4	K3	1
32	c	<p>Zależności logiczne oznaczają, że dla każdego produktu należy uruchomić WYSZUKIWANIE → WYSWIETLANIE → DODAWANIE przed uruchomieniem ZAMAWIANIA. Możesz dodać więcej produktów (używając tego samego przepływu), zanim uruchomisz ZAMAWIANIE.</p> <p>W oparciu o to, PT1 lub PT2 muszą być wykonane jako pierwsze, w przeciwnym razie nie będzie możliwy postęp.</p> <p>W pierwszej kolejności należy nadać priorytet WYSWIETLANIE i DODAWANIE produktowi B, ponieważ jego przypadki testowe (PT6, PT4) mają wyższy priorytet. Tak więc pierwsze 3 testy do wykonania to PT2 -> PT4 -> PT6.</p> <p>Teraz musimy rozważyć, czy najpierw uruchomić PT7, a następnie cały przepływ dla produktu produktu A, czy też najpierw uruchomić PT dla produktu A. Jeśli PT7 ma niższy priorytet niż pozostałe testy, powinny one zostać przetestowane jako pierwsze. Dlatego cały przepływ powinien wyglądać następująco:</p> $\text{PT2} \rightarrow \text{PT4} \rightarrow \text{PT6} \rightarrow \text{PT1} \rightarrow \text{PT3} \rightarrow \text{PT5} \rightarrow \text{PT7}$	FL-5.1.5	K3	1

	<p>a) Odpowiedź niepoprawna. PT1 musi zostać wykonany przed PT3. b) Odpowiedź niepoprawna. Jak pokazano powyżej, PT7 jest wykonywany jako ostatni. c) Odpowiedź poprawna. PT1 zostanie wykonany jako czwarty przypadek testowy. d) Odpowiedź niepoprawna. Produkt B musi zostać wykonany przed produktem A.</p>		
--	---	--	--

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
33	d	<p>a) Odpowiedź niepoprawna. Testowanie użyteczności ma cel biznesowy i służy krytyce produktu (kwadrant Q3).</p> <p>b) Odpowiedź niepoprawna. Testowanie funkcjonalne ma cel biznesowy (kwadrant Q2).</p> <p>c) Odpowiedź niepoprawna. Testowanie akceptacyjne przez użytkownika ma cel biznesowy i służy krytyce produktu (kwadrant Q3).</p> <p>d) Odpowiedź poprawna. Testowanie integracji modułów ma cel technologiczny i służy wspieraniu zespołu poprzez ukierunkowanie procesu wytwarzania oprogramowania (kwadrant Q1).</p>	FL-5.1.7	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
34	c	<p>Rozważmy poszczególne ryzyka i sposoby ich łagodzenia:</p> <ol style="list-style-type: none"> 1. Długi czas odpowiedzi (1) można sprawdzić w ramach testowania wydajnościowego (B). 2. Na zmiany preferencji konsumentów (2) zwykle nie mamy wpływu, więc na ogół akceptujemy takie ryzyko (A). 3. Zalanie serwerowni wodą (3) może spowodować poważne szkody, więc powinniśmy dokonać przeniesienia ryzyka, np. poprzez zakup polisy ubezpieczeniowej (D). 4. Wiemy, że pacjenci, którzy przekroczyli pewien wiek, otrzymują niepoprawne raporty (4), a to może oznaczać potencjalny problem dotyczący wartości brzegowej, który może być skutecznie wykryty poprzez zastosowanie techniki w rodzaju analizy wartości brzegowych (C). <p>Właściwa kombinacja ryzyk i sposobów ich łagodzenia to 1B, 2A, 3D i 4C, a zatem POPRAWNA odpowiedzią jest odpowiedź c).</p>	FL-5.2.4	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
35	a	<p>a) Odpowiedź poprawna. Metryki jakości produktu służą do pomiaru charakterystyk jakościowych. Średni czas do awarii określa dojrzałość produktu, zatem jest metryką jakości produktu.</p> <p>b) Odpowiedź niepoprawna. To przykład metryki dotyczącej defektów, a nie jakości produktu.</p> <p>c) Odpowiedź niepoprawna. To przykład metryki dotyczącej pokrycia, a nie jakości produktu.</p> <p>d) Odpowiedź niepoprawna. To przykład metryki dotyczącej defektów, a nie jakości produktu.</p>	FL-5.3.1	K1	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
36	a	<p>a) Odpowiedź poprawna. Klient znajduje się w innej lokalizacji i w innej strefie czasowej, więc bezpośrednią komunikację może okazać się trudna.</p> <p>b) Odpowiedź niepoprawna. Tablice wskaźników zwykle są dostępne w dowolnym momencie dla każdego użytkownika, więc różne strefy czasowe nie będą w tym przypadku dużą przeszkodą w komunikacji – w odróżnieniu od bezpośrednią komunikacji słownej.</p> <p>c) Odpowiedź niepoprawna. Różnica czasu między Europą i Ameryką wynosi kilka godzin i mogą się z tym wiązać pewne niedogodności, jednak bez wątpienia nie tak wielkie jak w przypadku bezpośrednią komunikacji.</p> <p>d) Odpowiedź niepoprawna. Videokonferencje to wygodny sposób komunikacji. Chociaż komunikacja między Europą i Ameryką w godzinach roboczych zwykle wymaga od jednej strony nawiązania połączenia wczesnym rankiem lub w późnych godzinach wieczornych, nie wiążą się z tym jednak tak duże utrudnienia jak w przypadku bezpośrednią komunikacji.</p>	FL-5.3.3	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
37	a	<p>a) Odpowiedź poprawna. W przypadku złożonego elementu konfiguracji (np. środowiska testowego) system zarządzania konfiguracją rejestruje elementy, z których ten element się składa, a także ich powiązania i wersje.</p> <p>b) Odpowiedź niepoprawna. Narzędzia do zarządzania konfiguracją nie służą do wykonywania przypadków testowych i obliczania pokrycia.</p> <p>c) Odpowiedź niepoprawna. Narzędzie do zarządzania konfiguracją nie jest narzędziem do zarządzania licencjami.</p> <p>d) Odpowiedź niepoprawna. Narzędzia do zarządzania konfiguracją nie generują danych testowych.</p>	FL-5.4.1	K2	1
38	b	<p>a) Odpowiedź niepoprawna. Stwierdzenie jest prawdziwe, jednak nie ma zbyt dużej wartości dla programistów.</p> <p>b) Odpowiedź poprawna. Rezultaty testów wskazują, że prawdopodobnie system ignoruje zduplikowane wartości i sortuje listę bez uwzględniania powtórzeń. Zapewne jest to przyczyną awarii w PT3, PT4 i PT5. Takie informacje mogą pomóc programistom w odszukaniu defektu i jego skuteczniejszym usunięciu.</p> <p>c) Odpowiedź niepoprawna. System sortuje liczby ujemne. Problem polega raczej na nieuwzględnianiu powtarzających się wartości.</p> <p>d) Odpowiedź niepoprawna. Przypadki testowe PT3, PT4 i PT5 nie zostały zaliczone, ale nie wiemy, czy wystąpiły w nich jakieś defekty.</p>	FL-5.5.1	K3	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
39	c	<p>Rozpatrzmy poszczególne kategorie narzędzi i ich opisy:</p> <ul style="list-style-type: none"> A. Narzędzia do testowania statycznego – pomagają testerom w prowadzeniu przeglądów i analizy statycznej (4). B. Narzędzia zwiększające skalowalność i standaryzację wdrażania – na przykład maszyny wirtualne i narzędzia do konteneryzacji (3). C. Narzędzia DevOps – wspomagają działanie potoku dostarczania, śledzenie przepływu pracy, zautomatyzowane procesy budowania oraz funkcjonowanie mechanizmów ciągłej integracji i ciągłego dostarczania (1). D. Narzędzia wspomagające współpracę – ułatwiają wymianę informacji (2). <p>Zatem POPRAWNE powiązanie to 1C, 2D, 3B, 4A, czyli POPRAWNĄ odpowiedzią jest odpowiedź c).</p>	FL-6.1.1	K2	1

Nr pytania	Poprawna odpowiedź	Uzasadnienie	Cel nauczania	Poziom wiedzy	Liczba punktów
40	a	<p>a) Odpowiedź poprawna. Automatyzacja testów umożliwia przeprowadzenie pomiarów, które są zbyt skomplikowane, aby mogły zostać dokonane przez ludzi, np. obliczenie miar pokrycia w testowaniu białośkrzynkowym nietrywialnego kodu.</p> <p>b) Odpowiedź niepoprawna. Jeśli używamy narzędzi testowych, odpowiedzialność za testowanie NIE przechodzi częściowo na dostawcę narzędzia. Dostawca nie uczestniczy w testowaniu, za które nadal odpowiadają testerzy. Dostawcę narzędzia można obciążyć odpowiedzialnością jedynie za sytuację, w której narzędzie nie działa zgodnie z oczekiwaniemi i udostępnia niepoprawne wyniki testów.</p> <p>c) Odpowiedź niepoprawna. Testerzy w dalszym ciągu muszą wykazywać się krytycznym myśleniem podczas analizy anomalii w wynikach testów, gdy chcą określić prawdopodobną przyczynę wystąpienia tych anomalii.</p> <p>d) Odpowiedź niepoprawna. Ani testerzy, ani narzędzia nie są w stanie wygenerować przypadków testowych wyłącznie na podstawie analizy kodu programu, ponieważ kod stanowi jedynie implementację i nie zawiera informacji o oczekiwanych rezultatach. Takie informacje muszą znaleźć się w innym elemencie podstawy testów, np. w specyfikacji projektu.</p>	FL-6.2.1	K1	1