

# Słownik wyrażen̄ związan̄ych z testowaniem

## Wersja 2.0

Wersja angielska **słownika wyrażen̄ związan̄ych z testowaniem** została opracowana przez ‘Glossary Working Party’ International Software Testing Qualifications Board w składzie

Edytor: Erik van Veenendaal (Holandia)

### Członkowie

Rex Black (USA)  
Sigrid Eldh (Szwecja)  
Isabel Evans (UK)  
Dorothy Graham (UK)  
Julian Harty (UK)  
David Hayman (UK)  
Juha Itkonen (Finlandia)  
Vipul Kocher (Indie)  
Fernando Lamas de Oliveira (Portugalia)  
Tilo Linz (Niemcy)  
Peter Morgan (UK)  
Thomas Müller (Szwajcaria)  
Avi Ofer (Izrael)  
Dale Perry (USA)  
Horst Pohlmann (Niemcy)  
Meile Posthuma (Holandia)  
Erkki Pöyhönen (Finlandia)  
Maaret Pyhäjärvi (Finlandia)  
Andy Redwood (UK)  
Stuart Reid (UK)  
Piet de Roo (Holandia)  
Steve Sampson (UK)  
Shane Saunders (UK)  
Hans Schaefer (Norwegia)  
Jurriën Seubers (Holandia)  
Dave Sherratt (UK)  
Mike Smith (UK)  
Andreas Spillner (Niemcy)  
Lucjan Stapp (Polska)  
Richard Taylor (UK)  
Geoff Thompson (UK)  
Stephanie Ulrich (Niemcy)  
Matti Vuori (Finlandia)  
Gearrel Welvaart (Holandia)  
Pete Williams (UK)

Przetłumaczył na język polski zespół w składzie:

Kamila Dec

Mariusz Janczewski

Joanna Kazun

Jan Sabak

Lucjan Stapp (kierownik Zespołu)

Agata Szybowska

Piotr Ślęzak

Scalenie i rewizję dokumentu wykonał Zespół w składzie:

Kamila Dec

Lucjan Stapp (kierownik Zespołu)

Piotr Ślęzak

# SŁOWNIK

## Polski

## Angielski

### A

**abstrakcyjny przypadek testowy:** patrz *przypadek testowy wysokiego poziomu*

**akceptacja:** patrz *testowanie akceptacyjne*

**akceptacyjne testy produkcyjne:** Patrz *produkcyjne testy akceptacyjne*

**analiza drzewa usterek (FTA** akronim od angielskiego **Fault Tree Analysis**): Metoda używana do analizy przyczyn usterek (defektów). Technika modeluje wizualnie jak związki logiczne pomiędzy awariami, błędami człowieka i zewnętrznymi zdarzeniami mogą powodować powstawanie specyficznych defektów.

**analiza drzewa usterek oprogramowania:** patrz *analiza drzewa usterek (FTA)*

**analiza dynamiczna:** Proces oceny systemu lub modułu w oparciu o jego zachowanie w działaniu, np. zarządzanie pamięcią, wykorzystaniem procesora [IEEE 610]

**analiza mutacji:** Metoda określenia dokładności zestawu testowego poprzez zmierzenie, w jakim stopniu jest on w stanie odróżnić program od jego nieznacznie różniących się odmian (mutantów).

**analiza pokrycia:** Pomiar pokrycia osiągnięty podczas wykonywania testów wg z góry określonych kryteriów, przeprowadzany w celu określenia czy potrzebne są dodatkowe testy; jeśli odpowiedź brzmi tak, to podejmowana jest decyzja, które przypadki testowe należy wykonać.

**analiza przepływu danych:** Analiza statyczna przeprowadzana w oparciu o definicje i użycie zmiennych.

**analiza przepływu sterowania:** Forma analizy statycznej oparta na reprezentacji wszystkich możliwych sekwencji zdarzeń (ścieżek) mających miejsce podczas działania modułu lub systemu

**abstract test case:** See *high level test case*

**acceptance:** See *acceptance testing*

**production acceptance testing:** See *operational acceptance testing*

**Fault Tree Analysis (FTA):** A technique used to analyze the causes of faults (defects). The technique visually models how logical relationship between failures, human errors and external events can combine to cause specific defects to occur

**Software Fault Tree Analysis (SFTA):** See *Fault Tree Analysis (FTA)*.

**dynamic analysis:** The process of evaluating behavior, e.g. memory performance, CPU usage, of a system or component during execution. [After IEEE 610]

**mutation analysis:** A method to determine test suite thoroughness by measuring the extent to which a test suite can discriminate the program from slight variants (mutants) of the program.

**coverage analysis:** Measurement of achieved coverage to a specified coverage item during test execution referring to predetermined criteria to determine whether additional testing is required and if so, which test cases are needed.

**data flow analysis:** A form of static analysis based on the definition and usage of variables.

**control flow analysis:** A form of static analysis based on representation of all possible sequences of events (paths) in the execution through a component or system.

**analiza przyczyn i skutków awarii (FMEA - akronim od angielskiego Failure Mode and Effect Analysis):** Systematyczne podejście do identyfikacji i analizy ryzyka polegające na wskazywaniu możliwych stanów awarii i zapobieganiu ich wystąpieniu.

**analiza przyczyn i skutków awarii oprogramowania (SFMEA - akronim od angielskiego Software Failure Mode and Effect Analysis):** patrz *analiza przyczyn i skutków awarii (FMEA)*

**analiza przyczyn, skutków i krytyczności awarii (FMECA - akronim od angielskiego Failure Mode, Effect and Criticality Analysis):** Rozszerzenie FMEA o analizę krytyczności, która jest używana do wyznaczania prawdopodobieństw wystąpienia stanów awarii przy ważności ich konsekwencji. Wynikiem jest uwypuklenie stanów awarii z odpowiednio wysokim prawdopodobieństwem i wagą konsekwencji, co pozwala na ukierunkowanie działań zapobiegawczych tam, gdzie przyniosą największe korzyści. Patrz także *analiza przyczyn i skutków awarii (FMEA)*

**analiza przyczyn, skutków i krytyczności awarii oprogramowania (SFMECA - akronim od angielskiego Software Failure Mode, Effect and Criticality Analysis):** patrz *analiza przyczyn, skutków i krytyczności awarii (FMECA)*

**analiza przyczynowo-skutkowa:** Patrz: *tworzenie grafów przyczynowo-skutkowych.*

**analiza przyczyny podstawowej:** Technika analizy zorientowana na identyfikację podstawowych przyczyn defektów. Przez wprowadzenie ukierunkowanych miar na naprawę podstawowych przyczyn defektów, istnieje nadzieja, że prawdopodobieństwo ponownego wystąpienia defektu będzie zminimalizowane.

**analiza punktów funkcyjnych (APF):** Metoda mająca na celu pomiar rozmiaru funkcjonalności systemu informacyjnego. Pomiar ten jest niezależny od technologii i może stanowić podstawę pomiarów produktywności, szacowania potrzebnych zasobów i kontroli przedsięwzięcia.

**Failure Mode and Effect Analysis (FMEA):** A systematic approach to risk identification and analysis of identifying possible modes of failure and attempting to prevent their occurrence.

**Software Failure Mode and Effect Analysis (SFMEA):** See **Failure Mode and Effect Analysis (FMEA)**.

**Failure Mode, Effect and Criticality Analysis (FMECA):** An extension of FMEA, as in addition to the basic FMEA, it includes a criticality analysis, which is used to chart the probability of failure modes against the severity of their consequences. The result highlights failure modes against the severity of their consequences. The result highlights failure modes with relatively high probability and severity of consequences, allowing remedial effort to be directed where it produce the greatest values. See also *Failure Mode and Effect Analysis (FMEA)*:

**Software Failure Mode Effect, and Criticality Analysis (SFMECA):** See *Failure Mode and Effect, and Criticality Analysis (FMECA)*.

**cause-effect analysis:** See *cause-effect graphing*.

**root cause analysis:** An analysis technique aimed at identifying the root causes of defects. By directing correcting measures at root causes, it is hoped that the likelihood of defect recurrence will be minimized.

**Function Point Analysis (FPA):** Method aiming to measure the size of the functionality of an information system. The measurement is independent of the technology. This measurement may be used as a basis for the measurement of productivity, the estimation of the needed

resources, and project control.

**analiza punktów testowych ( TPA - Test Point Analysis (TPA):** A formula based test estimation method based on function point analysis. [TMap]

Formuła umożliwiająca estymację czasu testów oparta na analizie punktów funkcyjnych. [TMap]

**analiza ryzyka:** Proces oceny zidentyfikowanych ryzyk mający na celu oszacowanie ich wpływu i prawdopodobieństwa ujawnienia się.

**risk analysis:** The process of assessing identified risks to estimate their impact and probability of occurrence (likelihood).

**analiza statyczna kodu:** Analiza kodu źródłowego przeprowadzona bez wykonywania oprogramowania.

**static code analysis:** Analysis of source code carried out without execution of that software.

**analiza statyczna:** Analiza artefaktów oprogramowania, np. wymagań bądź kodu programu przeprowadzona bez wykonywania tych artefaktów.

**static analysis:** Analysis of software artifacts, e.g. requirements or code, carried out without execution of these software artifacts.

**analiza wartości brzegowych:** Czarno-skrzynkowa technika projektowania przypadków testowych, w której przypadki te są projektowane w oparciu o wartości brzegowe. Patrz *wartość brzegowa*.

**boundary value analysis:** A black box test design technique in which test cases are designed based on boundary values. See also *boundary value*.

**analiza wpływu:** Oszacowanie zmiany w dokumentacji projektowej, testowej oraz zmian w modułach koniecznych do zrealizowania żądanej zmiany określonych wymagań.

**impact analysis:** The assessment of change to the layers of development documentation, test documentation and components, in order to implement a given change to specified requirements.

**analiza zagrożeń:** Technika używana do charakteryzowania elementów ryzyka. Wynik analizy zagrożeń rzutuje na wybór metod używanych do wytwarzania i testowania oprogramowania. Patrz także *analiza ryzyka*

**hazard analysis:** A technique used to characterize the element of risk. The result of hazard analysis will drive the methods used for development and testing of a system. See also *risk analysis*

**analizator kodu:** Patrz: *analizator statyczny kodu*.

**code analyzer:** See *static code analyzer*.

**analizator statyczny kodu:** Narzędzie wykonujące analizę statyczną kodu. Narzędzie takie sprawdza kod źródłowy pod względem pewnych właściwości, takich jak zgodność ze standardami kodowania, metryki jakości lub anomalie przepływu danych.

**static code analyzer:** A tool that carries out static code analysis. The tool checks source code, for certain properties such as conformance to coding standards, quality metrics or data flow anomalies.

**analizator statyczny:** Narzędzie wykonujące analizę statyczną.

**static analyzer:** A tool that carries out static analysis.

**analizator:** Patrz *analizator statyczny*.

**analyzer:** See *static analyzer*.

**anomalia:** Dowolny warunek, który odchodzi od oczekiwań bazujących na specyfikacji wymagań, dokumentacji projektowej, dokumentacji użytkownika, standardach, i innych; albo od czyjejs percepcji lub doświadczenia. Anomalie mogą zostać znalezione podczas: przeglądów, testowania, analizy, kompilacji, albo podczas użytkowania oprogramowania lub odpowiedniej dokumentacji. [IEEE 1044] Patrz *błąd, defekt, odchylenie, pomyłka, usterka, awaria, incydent, problem.*

**atak na oprogramowanie:** Patrz *atak*

**atak usterek:** Patrz *atak*

**atak:** Ukierunkowane działanie mające na celu ocenę jakości, w szczególności niezawodności przedmiotu testu, poprzez wymuszanie wystąpienia określonej awarii.

**atrakcyjność:** Zdolność oprogramowania do bycia atrakcyjnym dla użytkownika. [ISO 9126]. Patrz: *użyteczność.*

**atrybut jakościowy:** Cecha lub właściwość, która wpływa na jakość obiektu. [IEEE 610]

**audyt konfiguracji:** Sprawdzenie zawartości bibliotek zawierającej elementy konfiguracji np. na zgodność ze standardami [IEEE610]

**audyt:** Niezależna ocena oprogramowania lub procesów w celu ustalenia zgodności ze standardami, wytycznymi, specyfikacjami oraz/lub procedurami, oparta na obiektywnych kryteriach, wliczając dokumenty, które określają:

- (1) postać lub zawartość produkowanego produktu
- (2) proces wg którego produkt powinien być produkowany
- (3) jak stosowanie się do standardu lub specyfikacji jest mierzone [IDEE 1028]

**automat skończony:** Model obliczeniowy składający się ze skończonej liczby stanów i przejść pomiędzy tymi stanami, możliwie z towarzyszącymi im akcjami. [IEEE 610]

**automatyzacja testowania:** Użycie oprogramowania do wykonania lub wspierania czynności testowych, np. zarządzania testami, projektowania testów,

**anomaly:** Any condition that deviates from expectation based on requirements specifications, design documents, user documents, standards, etc. or from someone's perception or experience. Anomalies may be found during, but not limited to, reviewing, testing, analysis, compilation, or use of software products or applicable documentation. [IEEE 1044] See also *bug, defect, deviation, error, fault, failure, incident, problem.*

**software attack:** See *attack.*

**fault attack:** See *attack*

**attack:** Directed and focused attempt to evaluate the quality especially of a test object by attempting to force specific failures to occur.

**attractiveness:** The capability of the software product to be attractive to the user. [ISO 9126] See also *usability.*

**quality attribute:** A feature or characteristic that affects an item's quality. [IEEE 610]

**configuration auditing:** The function to check on the contents of libraries of configuration items, e.g. for standards compliance. [IEEE 610]

**audit:** An independent evaluation of software products or processes to ascertain compliance to standards, guidelines, specifications, and/or procedures based on objective criteria, including documents that specify:

- (1) the form or content of product to be produced
- (2) the process by which the product shall be produced
- (3) how compliance to standards or guidelines shall be measured [IEEE 1028]

**finite state machine:** A computational model consisting of a finite number of states and transitions between those states, possibly with accompanying actions. [IEEE 610]

**test automation:** The use of software to perform or support test activities, e.g. test management, test design, test execution and results checking.

wykonania testów i sprawdzania wyników.

**automatyzacja wykonania testu:** Użycie oprogramowania, np. narzędzia rejestrująco-odtwarzającego, w celu kontrolowania wykonania testu, porównania rezultatów rzeczywistych z oczekiwanymi, ustawienia warunków wstępnych testu i innych funkcji kontroli i raportowania testu.

**awaria:** Odchyłka modułu lub systemu od oczekiwanego zachowania lub rezultatu działania. [Fenton]

**test execution automation:** The use of software, e.g. capture/playback tools, to control the execution of tests, the comparison of actual results to expected results, the setting up of test preconditions, and other test control and reporting functions.

**failure:** Deviation of the component or system from its expected delivery, service or result. [After Fenton]

## B

**bezpieczeństwo:** Zdolność oprogramowania do osiągania akceptowalnych poziomów ryzyka wystąpienia szkody w stosunku do ludzi, biznesu, oprogramowania, majątku lub środowiska w określonym kontekście użycia. [ISO 9126]

**biało-skrzynkowe**

**projektowania:** Procedura tworzenia i/ lub wybierania przypadków testowych oparta na analizie wewnętrznej struktury modułu lub systemu.

**biało-skrzynkowe techniki:** Patrz *biało-skrzynkowe techniki projektowania*

**blok podstawowy:** Jedna lub wiele następujących po sobie instrukcji bez rozgałęzień.

Uwaga: Węzeł w grafie przepływów reprezentuje blok podstawowy.

**błąd:** Działanie człowieka powodujące powstanie nieprawidłowego rezultatu. [IEEE 610]

**brudne testowanie:** Patrz *testowanie negatywne*

**bufor:** Urządzenie lub pamięć (baza) używana do czasowego przechowywania danych. Wykorzystywany jest w sytuacjach różnic czasowych w przekazywaniu danych, różnic w wystąpieniu zdarzeń w procesie przesyłania danych lub w przypadku różnej ilości danych jaką mogą przyjąć urządzenia lub procesy zaangażowane w przesyłanie lub wykorzystywanie tych danych [IEEE 610]

**safety:** The capability of the software product to achieve acceptable levels of risk of harm to people, business, software, property or the environment in a specified context of use. [ISO 9126]

**white-box test design technique:** Procedure to derive and/or select test cases based on an analysis of the internal structure of a component or system.

**white-box techniques:** See *white-box test design techniques*.

**basic block:** A sequence of one or more consecutive executable statements containing no branches. Note: A node in a control flow graph represents a basic block.

**error:** A human action that produces an incorrect result. [After IEEE 610]

**dirty testing:** See *negative testing*.

**buffer:** A device or storage used to store data temporarily for differences in rates of data flow, time or occurrences of events or amounts of data that can be handled by the devices or processes involved in the transfer or use the data [IEEE 610]

## C

**CASE:** (Akronim od angielskiego Computer Aided Software Engineering) Inżynieria oprogramowania wspomagana komputerowo

**CASE:** Acronym for Computer Aided Software Engineering.

**CAST:** (Akronim od angielskiego Computer Aided Software Testing) Testowanie oprogramowania wspomagane komputerowo. Patrz *automatyzacja testowania*

**cecha oprogramowania:** Patrz *cecha*

**cecha:** Atrybut modułu lub systemu wyspecyfikowany w dokumentacji wymagań lub wywnioskowany z niej (na przykład: niezawodność, użyteczność, ograniczenia projektowe). [IEEE 1008]

**cel testu:** Przyczyna zaprojektowania i przeprowadzenia testu.

**certyfikacja:** Proces potwierdzający że moduł, system lub osoba spełnia określone wymagania, Certyfikat można uzyskać np. poprzez zdanie egzaminu.

**charakterystyka czasowa:** Patrz *wydajność charakterystyka*

**charakterystyka jakości**

**oprogramowania:** Patrz *atrybut jakościowy charakterystyka oprogramowania:* Patrz *atrybut jakościowy*

**codzienne buildowanie:** Aktywność, której celem jest kompilacja i integracja kompletna systemu każdej doby (zwykle nocą), tak że zintegrowany system wraz z ostatnimi zmianami jest dostępny w każdej chwili.

**COTS:** Akronim od ang. Commercial Off-The-Shelf Software. Patrz *oprogramowanie z półki*

**cykl testowy:** Wykonanie procesu testowego w stosunku do pojedynczego, możliwego do zidentyfikowania wydania testowanego obiektu.

**cykl życia oprogramowania:** Okres czasu rozpoczynający się kiedy pojawił się pomysł na oprogramowanie i kończący się gdy oprogramowanie nie jest już dostępne do użytku. Zazwyczaj cykl życia oprogramowania zawiera fazę koncepcji, fazę wymagań, fazę projektowania, fazę implementacji, fazę testów, fazę instalacji i zastępowania, fazę wykorzystania produkcyjnego i pielęgnowania oraz czasami, fazę wycofania. Uwaga: te fazy mogą na siebie zachodzić lub mogą być wykonywane iteracyjnie.

**CAST:** Acronym for Computer Aided Software Testing. See also *test automation*.

**software feature:** See *feature*.

**feature:** An attribute of a component or system specified or implied by requirements documentation (for example reliability, usability or design constraints). [After IEEE 1008]

**test objective:** A reason or purpose for designing and executing a test.

**certification:** The process of confirming that a component, system or person complies with its specified requirements, e.g. by passing an exam.

**time behavior:** See *performance*.

**software quality characteristic:** See *quality attribute*.

**software product characteristic:** See *quality attribute*.

**daily build:** a development activity where a complete system is compiled and linked every day (usually overnight), so that a consistent system is available at any time including all latest changes.

**COTS:** Acronym for Commercial Off-The-Shelf software. See *off-the-shelf software*.

**test cycle:** Execution of the test process against a single identifiable release of the test object.

**software life cycle:** The period of time that begins when a software product is conceived and ends when the software is no longer available for use. The software life cycle typically includes a concept phase, requirements phase, design phase, implementation phase, test phase, installation and checkout phase, operation and maintenance phase, and sometimes, retirement phase. Note these phase may overlap or be performed iteratively.



**czarno-skrzynkowa technika projektowania przypadków testowych:** Procedura wywodząca i/lub wybierająca przypadki testowe w oparciu o analizę specyfikacji, funkcjonalnej lub niefunkcjonalnej, modułu lub systemu bez odniesienia do jego wewnętrznej struktury.

## D

**dane testowe:** Dane, które istnieją (przykładowo w bazie danych) przed wykonaniem testu, i które mają wpływ na testowany moduł lub system, lub na które wywiera wpływ testowany moduł lub system.

**debuger:** Patrz *narzędzie do debugowania*

**debugowanie:** Proces wyszukiwania, analizowania i usuwania przyczyn awarii oprogramowania.

**decyzja:** Punkt w programie, w którym przepływ sterowania ma dwie lub więcej alternatywne drogi. Węzeł grafu przepływu sterowania, z którego wychodzą dwie lub więcej gałęzi.

**defekt:** Wada modułu lub systemu, która może spowodować, że moduł lub system nie wykona zakładanej czynności, np. niepoprawne wyrażenie lub definicja danych. Defekt, który wystąpi podczas uruchomienia programu, może spowodować awarię modułu lub systemu.

**definicja danej:** Instrukcja przypisująca wartość zmiennej.

**diagram przepływu sterowania:** Abstrakcyjna prezentacja możliwych sekwencji zdarzeń (w postaci ścieżek) mających miejsce podczas uruchomienia modułu lub systemu..

**diagram stanów:** Diagram, który przedstawia stany, jakie moduł lub system może przyjąć oraz pokazuje zdarzenia lub okoliczności, które powodują zmiany stanów i/lub wynikają z tych zmian. [IEEE 610]

**dojrzałość:** (1) Zdolność utrzymania przez organizację skuteczności oraz efektywności w realizowanych przez nią procesach i stosowanych praktykach. Patrz także *Model Dojrzałości Organizacyjnej (CMM)*, *Model Dojrzałości Testów (TMM)*.

(2) Zdolność oprogramowania do uniknięcia

**test data:** Data that exists (for example, in a database) before a test is executed, and that affects or is affected by the component or system under test.

**debugger:** See *debugging tool*.

**debugging:** The process of finding, analyzing and removing the causes of failures in software.

**decision:** A program point at which the control flow has two or more alternative routes. A node with two or more links to separate branches.

**defect:** A flaw in a component or system that can cause the component or system to fail to perform its required function, e.g. an incorrect statement or data definition. A defect, if encountered during execution, may cause a failure of the component or system.

**data definition:** An executable statement where a variable is assigned a value.

**control flow graph:** An abstract representation of all possible sequences of events (paths) in the execution through a component or system.

**state diagram:** A diagram that depicts the states that a component or system can assume, and shows the events or circumstances that cause and/or result from a change from one state to another. [IEEE 610]

**maturity:** (1) The capability of an organization with respect to the effectiveness and efficiency of its processes and work practices. See also *Capability Maturity Model*, *Test Maturity Model*.

(2) The capability of the software product to avoid failure as a result of defects in the

awarii jako rezultatu defektów. [ISO 9126]. software. [ISO 9126] See also *reliability*.  
Patrz także *niezawodność*.

**dokładność:** Zdolność oprogramowania do zapewnienia właściwych lub uzgodnionych rezultatów lub efektów z wymaganym poziomem precyzji [ISO 9126]. Patrz także *testowanie funkcjonalne*

**doskonalenie procesu:** Program działań zaprojektowany tak by poprawić wydajność i dojrzałość procesów w organizacji. Także wyniki takiego programu [CMMI]

**dostępność:** Stopień w jakim komponent lub system działa i jest dostępny, gdy wymagane jest jego użycie; często wyrażane w procentach [IEEE 610].

**drzewo klasyfikacji:** Drzewo obrazujące hierarchiczną zależność podzbiorów równoważności używane do projektowania przypadków testowych metodą drzewa klasyfikacji. Patrz także *metoda drzewa klasyfikacji*

**dynamiczne porównywanie:**

Porównywanie oczekiwanych i rzeczywistych rezultatów podczas wykonywania testów, na przykład za pomocą narzędzi do automatyzacji testów.

**dziedzina danych wejściowych:** Zbiór, z którego pochodzą dozwolone wartości wejść. Patrz również *dziedzina*.

**dziedzina danych wyjściowych:** Zbiór wszystkich możliwych wartości wyjściowych. Patrz także *dziedzina*.

**dziedzina:** Zbiór, z którego wybierane są wartości wejścia lub wyjścia.

**Dziki wskaźnik:** Wskaźniki które wskazują ważne obiekty odpowiadającego typu, lub do wyróżnionych wskaźników pustych (null) w językach, które to umożliwiają

**accuracy:** The capability of the software product to provide the right or agreed results or effects with the needed degree of precision [ISO 9126]. See also *functional testing*

**process improvement:** A program of activities design to improve the performance and maturity of the organization's processes, and the result of such a program [CMMI]

**availability:** The degree to which a component or system is operational and accessible when required to use, often expected as a percentage [IEEE 610]

**classification tree;** a tree showing equivalence partition hierarchy ordered, which is used to design test cases in the classification tree method. See also *classification tree method*

**dynamic comparison:** Comparison of actual and expected results, performed while the software is being executed, for example by a test execution tool.

**input domain:** The set from which valid input values can be selected. See also *domain*.

**output domain:** The set from which valid output values can be selected. See also *domain*.

**domain:** The set from which valid input and/or output values can be selected.

**wild pointer:** Pointers that do point to a valid object of the appropriate type, or to a distinguished null pointer value in languages which support this.

## E

**efekt próbnika:** Efekt wpływu elementu pomiarowego na moduł lub system podczas dokonywania pomiaru, np. poprzez narzędzie do testów wydajnościowych. Przykładowo wydajność testowanego oprogramowania może być nieznacznie gorsza, kiedy stosowane jest narzędzie do testów wydajnościowych

**efekt testu:** Zbiór kryteriów wyjścia

**probe effect:** The effect on the component or system by the measurement instrument when the component or system is being measured, e.g. by a performance testing tool or monitor. For example performance may be slightly worse when performance testing tools are being used.

**test target:** A set of exit criteria.

**efektywność:** Zdolność oprogramowania do zapewnienia odpowiedniego poziomu wydajności, relatywnie do ilości zużytych zasobów, w określonych warunkach. [ISO 9126]

**ekwipunek testowy:** Patrz *środowisko testowe*

**element konfiguracji:** Zestaw składający się ze sprzętu, oprogramowania lub ich obu, który jest poddany zarządzaniu konfiguracją i traktowany jako pojedynczy byt w procesie zarządzania konfiguracją. [IEEE 610]

**element testowy:** Pojedynczy element, który należy przetestować. Na ogół jest jeden przedmiot testów i wiele elementów testowych. Patrz także *przedmiot testów*.

**emulator:** Urządzenie, program komputerowy lub system, który przyjmuje takie same wejścia i generuje takie same wyjścia jak dany system. [IEEE 610]. Patrz również *symulator*.

**etap testów:** Patrz *poziom testów*

**ewaluacja:** Patrz *testowanie*

**efficiency:** The capability of the software product to provide appropriate performance, relative to the amount of resources used under stated conditions. [ISO 9126]

**test rig:** See *test environment*.

**configuration item:** An aggregation of hardware, software or both, that is designated for configuration management and treated as a single entity in the configuration management process. [IEEE 610]

**test item:** The individual element to be tested. There usually is one test object and many test items. See also *test object*.

**emulator:** A device, computer program, or system that accepts the same inputs and produces the same outputs as a given system. [IEEE 610] See also *simulator*.

**test stage:** See *test level*.

**evaluation:** See *testing*.

## F

**faza testów:** Wyróżniony zbiór aktywności testowych zebrany w podlegającą zarządzaniu fazę projektu, np. wykonanie testów na jakimś poziomie testów. [Gerrard]

**faza wykonania testu:** Okres w cyklu tworzenia oprogramowania, w trakcie którego moduły oprogramowania są wykonywane, a samo oprogramowanie jest oceniane w celu zdecydowania czy wymagania zostały spełnione. [IEEE 610]

**faza wymagań:** Przedział czasu w cyklu życia oprogramowania, podczas którego wymagania na oprogramowanie są zbierane i dokumentowane. [IEEE 610]

**funkcjonalność:** Zdolność oprogramowania do zapewnienia funkcji odpowiadających zdefiniowanym i przewidywanym potrzebom, gdy oprogramowanie jest używane w określonych warunkach. [ISO 9126]

**test phase:** A distinct set of test activities collected into a manageable phase of a project, e.g. the execution activities of a test level. [After Gerrard]

**test execution phase:** The period of time in a software development life cycle during which the components of a software product are executed, and the software product is evaluated to determine whether or not requirements have been satisfied. [IEEE 610]

**requirements phase:** The period of time in the software life cycle during which the requirements for a software product are defined and documented. [IEEE 610]

**functionality:** The capability of the software product to provide functions which meet stated and implied needs when the software is used under specified conditions. [ISO 9126]

## G

**generator testu:** Patrz *narzędzie do przygotowywania danych testowych*.

**test generator:** See *test data preparation tool*.

**gęstość błędów:** liczba usterek znalezionych w module lub systemie przypadająca na jednostkę wielkości modułu lub systemu (wyrażoną za pomocą standardowej miary oprogramowania, np. w liniach kodu, liczbie klas, punktach funkcyjnych)

**gęstość usterek:** Patrz *gęstość błędów*

**główny plan testów:** Plan testów, który odnosi się do wielu poziomów testów. Patrz także *plan testów*.

**graf przyczynowo-skutkowy:** Graficzna reprezentacja wejść i/lub bodźców (przyczyn) z odpowiadającymi im wyjściami (efektami), które mogą być wykorzystane do zaprojektowania przypadków testowych.

**defect density:** The number of defects identified in a component or system divided by the size of the component or system (expressed in standard measurement terms, e.g. lines-of-code, number of classes or function points).

**fault density:** See *defect density*.

**master test plan:** A test plan that typically addresses multiple test levels. See also *test plan*.

**cause-effect graph:** A graphical representation of inputs and/or stimuli (causes) with their associated outputs (effects), which can be used to design test cases.

## H

**harmonogram testów:** Lista aktywności, zadań lub zdarzeń z procesu testowego, określająca ich zamierzoną datę rozpoczęcia i zakończenia i/lub czas, oraz ich współzależności.

**hiperłącze:** Wskaźnik w dokumencie elektronicznym, który stanowi odwołanie do innego dokumentu elektronicznego.

**test schedule:** A list of activities, tasks or events of the test process, identifying their intended start and finish dates and/or times, and interdependencies.

**hyperlink:** A pointer with a web page that leads to the other web pages

## I

**identyfikacja konfiguracji:** Element zarządzania konfiguracją składający się z wyboru elementów konfiguracji, z których ma składać się system oraz zapisu ich funkcjonalnych i fizycznych właściwości w dokumentacji technicznej [patrz IEEE610]

**identyfikacja ryzyka:** Proces identyfikacji ryzyk wykorzystujący takie techniki jak burza mózgów, listy kontrolne, historie awarii.

**incydent testowy:** Patrz *incident*

**incydent:** Każde zdarzenie wymagające zbadania [IEEE 1008]

**Zgłoszenie incydentu:** dokument opisujący jakiegokolwiek zdarzenie, które nastąpiło np. podczas testowania wymagające dalszego zbadania. [IEEE 1008]

**informacja o statusie:** Element zarządzania konfiguracją składający się z rejestrowania i raportowania informacji potrzebnych do efektywnego zarządzania konfiguracją. Informacje te zawierają zestawienie zaakceptowanych elementów konfiguracji, status zaproponowanych zmian konfiguracji

**configuration identification:** An element of configuration management, consisting of selecting the configuration items for a system and recording their functional and physical characteristics in technical documentation. [IEEE 610]

**risk identification:** The process of identifying risks using techniques such as brainstorming, checklists and failure history.

**software test incident:** See *incident*.

**incident:** Any event occurring that requires investigation [After IEEE 1008]

**incident report:** A document reporting on any event that occurred, e.g. during the testing, which requires investigation. [After IEEE 829]

**status accounting:** An element of configuration management, consisting of the recording and reporting of information needed to manage a configuration effectively. This information includes a listing of the approved configuration identification, the status of proposed

oraz status implementacji zaakceptowanych zmian. [IEEE 610]

changes to the configuration, and the implementation status of the approved changes. [IEEE 610]

**infrastruktura testu:** Organizacyjne artefakty niezbędne do przeprowadzenia testu, składające się ze środowisk testowych, narzędzi testowych, wyposażenia biurowego i procedur.

**test infrastructure:** The organizational artifacts needed to perform testing, consisting of test environments, test tools, office environment and procedures.

**inspekcja:** Rodzaj przeglądu koleżeńkiego polegający na wizualnej weryfikacji dokumentów w celu wykrycia defektów, np. niezgodności ze standardami projektowymi lub dokumentacją wyższego poziomu. Jest to najbardziej formalna technika przeglądu, zawsze oparta na udokumentowanej procedurze. [IEEE 610, IEEE 1028] Zobacz również: *przeгляд koleżeński*

**inspection:** A type of peer review that relies on visual examination of documents to detect defects, e.g. violations of development standards and non-conformance to higher level documentation. The most formal review technique and therefore always based on a documented procedure. [After IEEE 610, IEEE 1028] See also *peer review*.

**inspektor:** Patrz *przeглядający*

**inspector:** See *reviewer*.

**instalowalność:** Zdolność oprogramowania do bycia zainstalowanym w wyspecyfikowanym środowisku. Patrz też: *przenaszalność*.

**installability:** The capability of the software product to be installed in a specified environment [ISO 9126]. See also *portability*.

**instrukcja kodu źródłowego:** Patrz *instrukcja*.

**source statement:** See *statement*.

**instrukcja wykonywalna:** Wyrażenie, które w trakcie kompilacji jest tłumaczone na kod binarny, i które będzie wykonywane proceduralnie podczas działania programu. Może ono wykonywać akcje na danych programu.

**executable statement:** A statement which, when compiled, is translated into object code, and which will be executed procedurally when the program is running and may perform an action on data.

**instrukcja:** Element języka programowania, który jest zwykle najmniejszą niepodzielną jednostką wykonania.

**statement:** An entity in a programming language, which is typically the smallest indivisible unit of execution.

**instrumentalizacja:** Dodanie kodu do programu w celu zbierania informacji o jego zachowaniu podczas wykonania, np. w celu pomiaru pokrycia kodu.

**instrumentation:** The insertion of additional code into the program in order to collect information about program behavior during execution, e.g. for measuring code coverage.

**instrumentalizator programowy:** Patrz *instrumentalizator*

**program instrumenter:** See *instrumenter*.

**instrumentalizator:** Narzędzie programowe użyte do wykonania instrumentalizacji.

**instrumenter:** A software tool used to carry out instrumentation.

**integracja podstawowych funkcjonalności systemu:** Metoda integracji, w której moduły lub systemy łączy się jak najwcześniej w celu uzyskania działającej podstawowej funkcjonalności. Patrz także: *testowanie integracyjne*

**functional integration:** An integration approach that combines the components or systems for the purpose of getting a basic functionality working early. See also *integration testing*.

**integracja:** Proces łączenia modułów lub systemów w większe zespoły.

**integration:** The process of combining components or systems into larger assemblies.

**iteracyjny model wytwarzania:** Metoda wytwarzania oprogramowania, w której projekt jest podzielony na większą liczbę iteracji. Iteracja stanowi zamknięty cykl wytwórczy dający w wyniku działającą wersję produktu (wewnętrzną lub zewnętrzną) będącą podzbiorem finalnego produktu, który rozrasta się z iteracji na iterację aż do produktu końcowego.

**iterative development model:** A development life cycle where a project is broken into a usually large number of iterations. An iteration is a complete development loop resulting in a release (internal or external) of an executable product, a subset of the final product under development, which grows from iteration to iteration to become the final product.

## J

**jakość oprogramowania:** Ogół funkcjonalności i cech oprogramowania, które charakteryzują się zdolnością zaspokajania stwierdzonych lub przewidywanych potrzeb. [ISO 9126]

**software quality:** The totality of functionality and features of a software product that bear on its ability to satisfy stated or implied needs. [After ISO 9126]

**jakość:** Stopień, w jakim moduł, system lub proces spełnia określone wymagania i/ lub spełnia potrzeby i oczekiwania klienta lub użytkownika. [IEEE 610]

**quality:** The degree to which a component, system or process meets specified requirements and/or user/customer needs and expectations. [After IEEE 610]

**jarzmo testowe:** Środowisko testowe, składające się z zaslepek i sterowników potrzebnych do wykonania testu.

**test harness:** A test environment comprised of stubs and drivers needed to execute a test.

**jednopoziomowy plan testów:** Plan testu, który odnosi się do jednego poziomu testowania. Patrz także *plan testów*.

**level test plan:** A test plan that typically addresses one test level. See also *test plan*.

**jednostka:** Patrz *moduł*

**unit:** See *component*.

**język skryptowy:** Język programowania, w którym pisane są skrypty testowe wykorzystywane przez narzędzie do uruchamiania testów (np. narzędzie rejestrująco-odtworzące).

**scripting language:** A programming language in which executable test scripts are written, used by a test execution tool (e.g. a capture/playback tool).

## K

**kamień milowy:** Punkt w czasie realizacji projektu, dla którego określone zostały (pośrednie) produkty oraz wyniki.

**milestone:** A point in time in a project at which defined (intermediate) deliverables and results should be ready.

**kierownik testów:** Osoba odpowiedzialna za zarządzanie projektem w zakresie czynności związanych z testowaniem, w zakresie zasobów uczestniczących w testowaniu oraz oceny przedmiotu testu. Osoba, która kieruje, zarządza, administruje, planuje i reguluje ocenę przedmiotu testu.

**test manager:** The person responsible for project management of testing activities and resources, and evaluation of a test object. The individual who directs, controls, administers, plans and regulates the evaluation of a test object.

**klasa równoważności:** Patrz *podzbiór równoważności*

**equivalence class:** See *equivalence partition*.

**kluczowy wskaźnik wydajności:** Patrz *wskaźnik wydajności*

**key performance indicator:** See *performance indicator*.

**kod:** Instrukcje komputera i definicje danych wyrażone w języku programowania lub w postaci wyjściowej w asemblerze, kompilatorze lub translatorze. [IEEE 610].

**koegzystencja:** Zdolność produktu oprogramowania do działania z innym niezależnym oprogramowaniem we wspólnym środowisku dzieląc wspólne zasoby. [ISO 9126] Patrz: *przenaszalność*.

**komercyjne oprogramowanie z półki:** Patrz *oprogramowanie z półki*

**komitet kontroli zmiany:** Patrz: *rada kontroli zmian*

**komparator testowy:** Narzędzie testowe do przeprowadzania automatycznego porównania rezultatów rzeczywistych z oczekiwanymi.

**komparator:** Patrz: *komparator testowy*

**kompilator:** Narzędzie, które tłumaczy programy napisane w języku wysokiego poziomu na ich odpowiedniki w języku maszynowym. [IEEE 610]

**kompletne testowanie:** Patrz *testowanie gruntowne*.

**konfiguracja:** Układ modułów lub systemów zdefiniowany za pomocą liczb, poprzez swoją naturę, oraz przez połączenia pomiędzy częściami składowymi.

**konkretny przypadek testowy:** Patrz *przypadek testowy niskiego poziomu*

**kontrola konfiguracji:** Element zarządzania konfiguracją składający się z oceny, koordynacji oraz udzielenia lub nieudzielenia zgody na zmianę elementów konfiguracji po formalnej identyfikacji elementu konfiguracji [IEEE 610]

**kontrola ryzyka:** Proces, w którym podejmuje się decyzje i implementuje metryki w celu redukcji ryzyka lub utrzymania go na określonym poziomie.

**kontrola testu:** Zadanie z zakresu zarządzania testem, którego celem jest opracowanie i zastosowanie działań korygujących projekt testowy, kiedy monitoring pokazuje odchylenie od planu. Patrz też *zarządzanie testami*.

**kontrola wersji:** patrz *kontrola*

**code:** Computer instructions and data definitions expressed in a programming language or in a form output by an assembler, compiler or other translator. [IEEE 610]

**co-existence:** The capability of the software product to co-exist with other independent software in a common environment sharing common resources. [ISO 9126] See also *portability*.

**commercial off-the-shelf software:** See *off-the-shelf software*.

**change control board:** See *configuration control board*.

**test comparator:** A test tool to perform automated test comparison of actual results with expected results.

**comparator:** See *test comparator*.

**compiler:** A software tool that translates programs expressed in a high order language into their machine language equivalents. [IEEE 610]

**complete testing:** See *exhaustive testing*.

**configuration:** The composition of a component or system as defined by the number, nature, and interconnections of its constituent parts.

**concrete test case:** See *low level test case*.

**configuration control:** An element of configuration management, consisting of the evaluation, co-ordination, approval or disapproval, and implementation of changes to configuration items after formal establishment of their configuration identification. [IEEE 610]

**risk control:** The process through which decisions are reached and protective measures are implemented for reducing risks to, or maintaining risks within, specified levels.

**test control:** A test management task that deals with developing and applying a set of corrective actions to get a test project on track when monitoring shows a deviation from what was planned. See also *test management*.

**version control:** See *configuration control*.

*konfiguracji*

**kontrola zmiany:** Patrz: *kontrola konfiguracji* **change control:** See *configuration control*.

**kontroler:** Patrz: *przeglądający*

**checker:** See *reviewer*.

**koszt jakości:** Całkowity koszt związany z jakością, na który składają się koszty działań prewencyjnych, ocen, koszty awarii wewnętrznych i zewnętrznych.

**cost of quality:** The total cost spent on quality activities and issues and often split into prevention costs, appraisal costs, internal failure costs and external failure costs.

**kreator instalacji:** Oprogramowanie dostarczone na odpowiednich nośnikach, które prowadzi instalatora przez proces instalacji. Zazwyczaj wykonuje proces instalacji, informuje o jego wynikach i prosi o wybór opcji.

**installation wizard:** Supplied software on any suitable media, which leads the installer through the installation process. It normally runs the installation process, provides feedback on installation results, and prompts for options.

**Kryteria akceptacji:** Kryteria wyjścia, które moduł lub system musi spełniać, aby został zaakceptowany przez użytkownika, klienta lub inny uprawniony podmiot [IEEE 610]

**acceptance criteria:** The exit criteria that a component or system must satisfy in order to be accepted by a user, customer or other authorized entity [IEEE 610]

**kryteria wznowienia:** Aktywności testowe, które muszą być powtórzone podczas ponownego rozpoczynania prac testowych po ich wstrzymaniu. [za IEEE 829]

**resumption criteria:** The testing activities that must be repeated when testing is restarted after a suspension. [After IEEE 829]

**kryteria zaliczenia/nie zaliczenia:** Reguły decyzyjne wykorzystywane do określenia czy obiekt testowany (funkcja) lub cecha zaliczyła test. [IEEE 829]

**pass/fail criteria:** Decision rules used to determine whether a test item (function) or feature has passed or failed a test. [IEEE 829]

**kryteria zawieszenia:** Kryteria używane do (tymczasowego) zatrzymania wszystkich lub części aktywności testowych na elementach testowych. [IEEE 829]

**suspension criteria:** The criteria used to (temporarily) stop all or a portion of the testing activities on the test items. [After IEEE 829]

**kryterium ukończenia:** Patrz: *kryterium wyjścia*.

**completion criteria:** See *exit criteria*.

**kryterium wejścia:** Zbiór ogólnych i specyficznych warunków, których spełnienie jest wymagane do kontynuacji procesu od określonego zadania, np. fazy testów. Celem kryterium wejścia jest ochrona przed rozpoczęciem zadania, w sytuacji, gdy pociąga to za sobą więcej (zmarowanych) nakładów pracy w porównaniu z nakładem pracy potrzebnym do osiągnięcia stanu spełnienia kryterium wejścia. [Gilb i Graham]

**entry criteria:** The set of generic and specific conditions for permitting a process to go forward with a defined task, e.g. test phase. The purpose of entry criteria is to prevent a task from starting which would entail more (wasted) effort compared to the effort needed to remove the failed entry criteria. [Gilb and Graham]



**kryterium wyjścia:** Zbiór ogólnych i specyficznych warunków, uzgodnionych z udziałowcami, których spełnienie jest wymagane do oficjalnego zakończenia procesu.

Celem kryterium wyjścia jest ochrona przez uznaniem zadania za ukończone w przypadku, gdy jakieś jego elementy nie są jeszcze w pełni wykonane. Kryteria wyjścia są stosowane jako argument przeciwko zakończeniu testów oraz do planowania, kiedy można to zrobić. [Gilb i Graham]

**kryterium zakończenia testu:** Patrz *kryterium wyjścia*

**kwifikacje:** Proces demonstrowania zdolności do spełnienia wyspecyfikowanych wymagań. Uwaga: termin „kwifikowany” jest używany do wybierania odpowiadającego statusu [ISO 9000]

**kwestionariusz oceny użyteczności oprogramowania (od akronimu ang. SUMI - Software Usability Measurement Inventory):** Technika testowania użyteczności oparta na kwestionariuszu mającym na celu ocenę użyteczności (np. satysfakcji użytkownika) modułu lub systemu. [Veenendaal]

**exit criteria:** The set of generic and specific conditions, agreed upon with the stakeholders, for permitting a process to be officially completed. The purpose of exit criteria is to prevent a task from being considered completed when there are still outstanding parts of the task which have not been finished. Exit criteria are used to report against and to plan when to stop testing. [After Gilb and Graham]

**test completion criteria:** See *exit criteria*.

**qualification:** The process of demonstrating the ability to fulfill specified requirements. Note the term ‘qualified’ is used to designate the corresponding status [ISO 9000]

**Software Usability Measurement Inventory (SUMI):** A questionnaire based usability test technique to evaluate the usability, e.g. user-satisfaction, of a component or system. [Veenendaal]

## L

**liczba cyklomatyczna:** Patrz *złożoność cyklomatyczna*

**lider testów:** Patrz *kierownik testów*

**log (dziennik) testów:** Chronologiczny zapis szczegółów związanych z wykonaniem testów. [IEEE 829]

**log przebiegu testu:** Patrz *log (dziennik) testów*

**logiczny przypadek testowy:** Patrz *przypadek testowy wysokiego poziomu*

**logowanie testu:** Proces rejestrowania informacji o wykonanych testach do logu testu.

**LSKiS: Liniowa Sekwencja Kodu i Skok (ang. A Linear Code Sequence And Jump),** składająca się z trzech następujących punktów (zwyczajowo identyfikowanych po numerze linii w kodzie źródłowym): rozpoczęcie liniowej sekwencji wykonywanych instrukcji, koniec sekwencji liniowej i docelowa linia, do której wykonywanie programu jest przekazywane

**cyclomatic number:** See *cyclomatic complexity*

**test leader:** See *test manager*.

**test log:** A chronological record of relevant details about the execution of tests. [IEEE 829]

**test run log:** See *test log*.

**logical test case:** See *high level test case*.

**test logging:** The process of recording information about tests executed into a test log.

**LCSAJ: A Linear Code Sequence And Jump,** consisting of the following three items (conventionally identified by line numbers in a source code listing): the start of the linear sequence of executable statements, the end of the linear sequence, and the target line to which control flow is transferred at the end of the linear sequence.

po zakończeniu liniowej sekwencji.

**łatwość nauki:** Zdolność oprogramowania do umożliwienia użytkownikowi nauki jego zastosowania. [ISO 9126] Patrz także *użyteczność*

**łatwość obsługi:** Cecha/zdolność oprogramowania do zapewnienia użytkownikowi możliwości jego obsługi i kontroli. [ISO 9126] Patrz także *użyteczność*.

**łoże testowe:** Patrz *środowisko testowe*. **test bed:** See *test environment*.

## M

**małpie testowanie:** Metoda testowania polegająca na losowym wyborze z szerokiego zakresu wejść i losowym naciskaniu przycisków, ignorując sposób, w jaki produkt powinien być używany.

**martwy kod:** Patrz *nieosiągalny kod*

**maskowanie defektów:** Sytuacja, w której występowanie jednego defektu uniemożliwia wykrycie innego [IEEE610]

**maskowanie usterek:** Patrz *maskowanie defektów*

**metoda drzewa klasyfikacji:** Czarno-skrzynkowa technika projektowania przypadków testowych, w której przypadki testowe, opisane za pomocą drzewa klasyfikacji, projektowane są do wykonania kombinacji reprezentantów wejść i / lub przetworzonych wyjść. [Grochtmann]

**metryka:** Skala pomiaru i sposób jej stosowania. [ISO 14598]

**metryki pokrycia Chowa.** Patrz: *pokrycie N-przebieg*. [Chow]

**miara:** Liczba bądź kategoria przypisana do atrybutu/ cechy obiektu poprzez wykonanie pomiaru. [ISO 14598]

**międzyoperacyjność:** Zdolność oprogramowania do współdziałania z jednym lub większą liczbą wskazanych modułów lub systemów [zgodnie z ISO 9126]. Patrz także *funkcjonalność*

**learnability:** The capability of the software product to enable the user to learn its application. [ISO 9126] See also *usability*.

**operability:** The capability of the software product to enable the user to operate and control it. [ISO 9126] See also *usability*.

**monkey testing:** Testing by means of a random selection from a large range of inputs and by randomly pushing buttons, ignorant on how the product is being used

**dead code:** See *unreachable code*.

**defect masking:** An occurrence in which one defect prevents the detection of another. [After IEEE 610]

**fault masking:** See *defect masking*.

**classification tree method:** A black box test design technique in which test cases, described by means of a classification tree, are designed to execute combinations of representatives of input and/or output domains. [Grochtmann]

**metric:** A measurement scale and the method used for measurement. [ISO 14598]

**Chow's coverage metrics:** See *N-switch coverage*. [Chow]

**measure:** The number or category assigned to an attribute of an entity by making a measurement. [ISO 14598]

**interoperability:** The capability of the software product to interact with one or more specified components or systems. [After ISO 9126] See also *functionality*.

**Model Dojrzałości Organizacyjnej (CMM Capability Maturity Model (CMM):** Akronim od angielskiego Capability Maturity Model): Pięć poziomów stanowiących strukturę, która opisuje kluczowe elementy efektywnego procesu oprogramowania. Model zdolności dojrzałości organizacyjnej składa się z najlepszych praktyk w planowaniu, inżynierii, zarządzaniu i konserwacji oprogramowania [CMM]. Patrz także *zintegrowany model dojrzałości organizacyjnej (CMMI)*.

**Model Dojrzałości Testów (TMM akronim od angielskiego Test Maturity Model):** Pięciostopniowa podstawa działań na rzecz ulepszenia procesu testowego, zgodna z Modelem Dojrzałości Organizacyjnej (Capability Maturity Model CMM), który opisuje kluczowe elementy efektywnego procesu testowego.

**model V:** Opis czynności cyklu życia wytwarzania oprogramowania od specyfikacji wymagań do pielęgnacji. Model V ilustruje jak czynności testowe mogą być integrowane z każdym etapem cyklu życia wytwarzania oprogramowania.

**model wzrostu niezawodności:** Model pokazujący poprawę niezawodności modułu lub systemu jako wynik poprawiania defektów w czasie.

**moderator:** Lider i główna osoba odpowiedzialna za prowadzenie inspekcji lub przeglądu.

**moduł:** Minimalny element programu, który będzie testowany w izolacji - niezależnie od innych modułów.

**moduł:** Minimalny element programu, który będzie testowany w izolacji - niezależnie od innych modułów.

**monitor:** Narzędzie programistyczne albo sprzętowe, które działa równolegle z testowanym modułem lub systemem i nadzoruje, rejestruje oraz analizuje zachowanie modułu lub systemu. [IEEE 610]

**monitorowanie testów:** Zadanie w zarządzaniu testowaniem, zajmujące się okresowym sprawdzaniem statusu projektu testowego. Przygotowywane są raporty

**Capability Maturity Model (CMM):** A five level staged framework that describes the key elements of an effective software process. The Capability Maturity Model covers best-practices for planning, engineering and managing software development and maintenance. [CMM] See also *Capability Maturity Model Integration (CMMI)*.

**Test Maturity Model (TMM):** A five level staged framework for test process improvement, related to the Capability Maturity Model (CMM) that describes the key elements of an effective test process.

**V-model:** A framework to describe the software development life cycle activities from requirements specification to maintenance. The V-model illustrates how testing activities can be integrated into each phase of the software development life cycle.

**reliability growth model:** A model that shows the improvement of reliability of a component or system as a result of correction of defect over time.

**moderator:** The leader and main person responsible for an inspection or other review process.

**component:** A minimal software item that can be tested in isolation.

**module:** See *component*.

**monitor:** A software tool or hardware device that runs concurrently with the component or system under test and supervises, records and/or analyses the behavior of the component or system. [After IEEE 610]

**test monitoring:** A test management task that deals with the activities related to periodically checking the status of a test project. Reports are prepared that compare

porównujące stan aktualny z planowanym. the actuals to that which was planned. See Patrz też *zarządzanie testami*. also *test management*.

## N

**najlepsza praktyka:** Zalecana metoda albo nowatorska praktyka, która przyczynia się do lepszych wyników organizacji w danym kontekście, zwykle uznawana za "najlepszą" przez inne podobne organizacje.

**narzędzie do analizy dynamicznej:** Narzędzie podające informacje o stanie wykonywanego programu. Takie narzędzia używane są najczęściej do znajdowania nieprzypisanych wskaźników, sprawdzanie alokacji, użycia i dealokacji pamięci oraz do oznaczania jej wycieków.

**narzędzie do analizy statycznej:** Patrz *analizator statyczny*.

**narzędzie do debugowania:** Narzędzie używane przez programistów do odtwarzania awarii, śledzenia stanu programu i odszukiwania usterek. Debugery umożliwiają programistom wykonywanie programów krok po kroku, zatrzymywanie wykonania programu na dowolnej instrukcji oraz sprawdzanie i ustawianie wartości zmiennych

**narzędzie do modelowania:** Narzędzie, które wspiera walidację modeli oprogramowania lub systemów [Graham]

**narzędzie do pomiaru pokrycia:** Patrz *narzędzie mierzące pokrycie*

**narzędzie do posiewu usterek:** Narzędzie do posiewu (tzn. celowego dodawania) usterek w module lub systemie

**narzędzie do projektowania testu:** Narzędzie, które wspiera projektowanie testu generując wejścia ze specyfikacji (przechowywanej na przykład w repozytorium systemu typu CASE jak narzędzie zarządzania wymaganiami), z wyspecyfikowanych warunków testu przechowywanych w samym narzędziu lub z kodu.

**narzędzie do przeglądów:** Narzędzie wspomagające proces przeglądu. Typowe cechy to: wsparcie planowania i śledzenia postępu prac, wsparcie komunikacji oraz repozytorium do zbierania i publikowania metryk.

**best practice:** A superior method or innovative practice that contributes to the improved performance of an organization under given context, usually recognized as 'best' by other peer organizations.

**dynamic analysis tool:** A tool that provides run-time information on the state of the software code. These tools are most commonly used to identify unassigned pointers, check pointer arithmetic and to monitor the allocation, use and de-allocation of memory and to flag memory leaks.

**static analysis tool:** See *static analyzer*.

**debugging tool:** A tool used by programmers to reproduce failures, investigate the state of programs and find the corresponding defect. Debuggers enable programmers to execute programs step by step, to halt a program at any program statement and to set and examine program variables.

**modelling tool:** A tool that supports the validation of models of the software or system [Graham].

**coverage measurement tool:** See *coverage tool*.

**fault seeding tool:** A tool for seeding (i.e. intentionally inserting) faults in a component or system

**test design tool:** A tool that supports the test design activity by generating test inputs from a specification that may be held in a CASE tool repository, e.g. requirements management tool, from specified test conditions held in the tool itself, or from code.

**review tool:** A tool that provides support to the review process. Typical features include review planning and tracking support, communication support, collaborative reviews and a repository for collecting and reporting of metrics.

**narzędzie do przygotowywania danych testowych:** Rodzaj narzędzia testowego, które zezwala na wybranie danych z istniejącej bazy danych lub ich stworzenie, wygenerowanie, przetworzenie i edycję dla użycia w testowaniu.

**narzędzie do śledzenia błędów:** Patrz *narzędzie do zarządzania defektami*.

**narzędzie do śledzenia defektów:** Patrz *narzędzie do zarządzania defektami*.

**Narzędzie do testowania hiperłączy:** Narzędzie używane do sprawdzenia czy na stronie internetowej nie znajdują się nieprawidłowe hiperłącza.

**narzędzie do testowania przeciążającego:** Narzędzie, które wspiera testowanie przeciążające.

**narzędzie do testów wydajnościowych:** Narzędzie wspierające testowanie wydajnościowe, zazwyczaj mające dwie funkcjonalności: generacja obciążenia i pomiar transakcji. Generowane obciążenie może symulować zarówno wielu użytkowników, jak i dużą ilość wprowadzanych danych. W czasie wykonywania testów pomiary są logowane tylko z wybranych transakcji. Narzędzia do testów wydajnościowych zazwyczaj dostarczają raporty bazujące na logowanych transakcjach oraz wykresy obciążenia w zależności od czasów odpowiedzi.

**narzędzie do testów zabezpieczeń:** Narzędzie wspierające testowanie charakterystyk zabezpieczeń i podatności.

**narzędzie do wykonywania testu:** Rodzaj narzędzia testowego, które jest w stanie wykonać inne oprogramowanie używając testowych skryptów automatycznych, np. narzędzie rejestrująco-odtwarzające. [Fewster i Graham]

**narzędzie do zabezpieczeń:** Narzędzie, które wspiera zabezpieczenia operacyjne.

**narzędzie do zarządzania defektami:** Narzędzie wspomagające zapisywanie informacji o defektach i śledzenie ich statusu oraz zmian. Takie narzędzia często posiadają funkcje śledzenia i nadzorowania przepływu pracy związanego z przypisywaniem, poprawianiem i retestowaniem defektów oraz posiadają funkcje raportowe. Patrz też

**test data preparation tool:** A type of test tool that enables data to be selected from existing databases or created, generated, manipulated and edited for use in testing.

**bug tracking tool:** See *defect management tool*.

**defect tracking tool:** See *defect management tool*.

**hyperlink tool:** A tool used to check that no broken hyperlinks are presented on a web site

**stress testing tool:** A tool that supports stress testing.

**performance testing tool:** A tool to support performance testing and that usually has two main facilities: load generation and test transaction measurement. Load generation can simulate either multiple users or high volumes of input data. During execution, response time measurements are taken from selected transactions and these are logged. Performance testing tools normally provide reports based on test logs and graphs of load against response times.

**security testing tool:** A tool that provides support for testing security characteristics and vulnerabilities.

**test execution tool:** A type of test tool that is able to execute other software using an automated test script, e.g. capture/playback. [Fewster and Graham]

**security tool:** A tool that supports operational security.

**defect management tool:** A tool that facilitates the recording and status tracking of defects and changes. They often have workflow-oriented facilities to track and control the allocation, correction and re-testing of defects and provide reporting facilities. See also *incident management tool*.

## *narzędzie do zarządzania incydentami*

**narzędzie do zarządzania incydentami:** Narzędzia ułatwiające rejestrację incydentów i śledzenie ich statusów. Często oferują funkcje śledzenia i kontroli przepływu pracy związanego z przydziałem, naprawą i retestami. Zapewniają również możliwość raportowania. Patrz również: *narzędzie do zarządzania defektami*

**incident management tool:** A tool that facilitates the recording and status tracking of incidents. They often have workflow-oriented facilities to track and control the allocation, correction and re-testing of incidents and provide reporting facilities. See also *defect management tool*.

**narzędzie do zarządzania konfiguracją:** Narzędzie wspomagające identyfikację oraz kontrolę elementów konfiguracji, ich statusów i wersji podczas wykonywania zmian oraz tworzenia wydań/publikacji, w których skład wchodzi.

**configuration management tool:** A tool that provides support for the identification and control of configuration items, their status over changes and versions, and the release of baselines consisting of configuration items.

**narzędzie do zarządzania testami:** Narzędzie wspomagające zarządzanie testami i sterowanie częścią procesu testowego. Zwykle ma kilka możliwości takich jak zarządzanie testami, planowanie testów w czasie, logowanie rezultatów, śledzenie postępu prac, zarządzanie incydentami oraz raportowanie wyników testów.

**test management tool:** A tool that provides support to the test management and control part of a test process. It often has several capabilities, such as testware management, scheduling of tests, the logging of results, progress tracking, incident management and test reporting.

**narzędzie do zarządzania wymaganiami:** Narzędzie wspierające rejestrowanie wymagań i ich atrybutów posiadające zdolność śledzenia powiązań między warstwami wymagań i zarządzania zmianami w wymaganiach. Niektóre narzędzia posiadają funkcjonalności umożliwiające analizę statyczną np. sprawdzanie spójności wymagań lub odstępstw od zdefiniowanych reguł dotyczących wymagań.

**requirements management tool:** A tool that supports the recording of requirements, requirements attributes (e.g. priority, knowledge responsible) and annotation, and facilitates traceability through layers of requirements and requirements change management. Some requirements management tools also provide facilities for static analysis, such as consistency checking and violations to pre-defined requirements rules.

**narzędzie mierzące pokrycie:** Narzędzie dające obiektywną miarę, które elementy strukturalne (np. wyrażenia, gałęzie) zostały sprawdzone przez zestaw testowy

**coverage tool:** A tool that provides objective measures of what structural elements, e.g. statements, branches have been exercised by a test suite.

**narzędzie monitorujące:** Patrz *monitor*

**monitoring tool:** See *monitor*.

**narzędzie nagrywająco-odtworzące:** Patrz *narzędzie rejestrująco-odtworzące*

**record/playback tool:** See *capture/playback tool*.

**narzędzie rejestrująco-nagrywające:** Patrz *narzędzie rejestrująco-odtworzące*.

**capture/replay tool:** See *capture/playback tool*.

**narzędzie rejestrująco-odtworzące:** **capture/playback tool:** A type of test execution tool where inputs are recorded during manual testing in order to generate automated test scripts that can be executed later (i.e. replayed). These tools are often used to support automated regression testing.

Rodzaj narzędzia do wykonywania testu, w którym dane wejściowe są nagrane podczas testów manualnych w celu wygenerowania automatycznych skryptów, które mogą być uruchomione później (tzn. powtórzone). Narzędzia te są często używane do wspierania automatycznego testowania regresyjnego.

**narzędzie testowe:** Oprogramowanie, który wspomaga jedną lub kilka czynności testowych, takich jak planowanie, zarządzanie, specyfikacja, budowa początkowych plików i danych, wykonanie i analiza testów [TMap]. Patrz *CAST*

**test tool:** A software product that supports one or more test activities, such as planning and control, specification, building initial files and data, test execution and test analysis. [TMap] See also *CAST*.

**narzędzie wspomagające przegląd:** Narzędzie, które zapewnia wsparcie w procesie przeglądu. Typowe cechy narzędzia to planowanie przeglądu i wspomaganie śledzenia, wspomaganie komunikacji, współpracę przy przeglądach oraz prowadzenie repozytorium do gromadzenia i raportowania metryk

**review tool:** A tool that provides support to the review process. Typical features include review planning and tracking support, communication support, collaborative reviews and a repository for collecting and reporting of metrics.

**nieosiągalny kod:** Kod, który nie może być osiągnięty i dlatego też nie jest wykonywany.

**unreachable code:** Code that cannot be reached and therefore is impossible to execute.

**niewykonalna ścieżka:** Ścieżka, dla której nie istnieje zestaw danych wejściowych, przy których przejście tej ścieżki jest możliwe.

**infeasible path:** A path that cannot be exercised by any set of possible input values.

**niezależność testowania:** Rozdzielenie odpowiedzialności, które sprzyja zapewnieniu obiektywności testowania. [DO 178b]

**independence of testing:** Separation of responsibilities, which encourages the accomplishment of objective testing. [After DO-178b]

**niezawodność:** Zdolność oprogramowania do wykonywania wymaganych funkcji w określonych warunkach przez określony czas lub dla określonej liczby operacji [ISO 9126]

**reliability:** The ability of the software product to perform its required functions under stated conditions for a specified period of time, or for a specified number of operations. [ISO 9126]

**niezdany:** Test uznaje się za niezdany, jeśli jego rzeczywisty rezultat nie zgadza się z oczekiwanym.

**fail:** A test is deemed to fail if its actual result does not match its expected result.

**niezgodność:** Niespełnienie konkretnego wymagania. [ISO 9000]

**non-conformity:** Non fulfillment of a specified requirement. [ISO 9000]

## O

**obsługa wyjątków:** Zachowanie modułu lub systemu w odpowiedzi na błędne wejście wprowadzone przez użytkownika lub inny moduł czy system.

**exception handling:** Behavior of a component or system in response to erroneous input, from either a human user or from another component or system, or to an internal failure.

**ocena heurystyczna:** Technika statycznego testowania użyteczności mającą na celu określenie zgodności interfejsu użytkownika z uznanymi zasadami użyteczności (tzw. "heurystykami").

**oczekiwany rezultat:** Zachowanie modułu lub systemu w określonych warunkach określone na podstawie specyfikacji lub innego źródła.

**oczekiwany wynik:** Patrz *oczekiwany rezultat*

**odchylenie:** Patrz *incydent*

**odporność:** Stopień, w jakim system lub moduł może działać prawidłowo przy nieprawidłowych danych lub przy dużym obciążeniu. [IEEE 610] Patrz także *tolerowanie błędów, tolerowanie usterek*

**odpowiedniość** Zdolność oprogramowania do dostarczenia odpowiedniego zestawu funkcji dla określonych zadań i celów użytkownika. [ISO 9126] Patrz też *funkcjonalność*.

**odsetek wykrytych błędów (OWB):** Liczba usterek wykrytych w fazie testowania podzielona przez liczbę usterek wykrytych w całym cyklu życia oprogramowania (w fazie testów i później)

**odsetek wykrywania usterek (OWU):** Patrz *odsetek wykrytych błędów (OWB)*

**odtwarzalność:** Zdolność oprogramowania do osiągnięcia określonego poziomu wydajności i przywracania danych uszkodzonych przez awarię [ISO 9126]. Patrz również *niezawodność*

**opis przekazywanego produktu:** Patrz *uwaga do wydania*

**oprogramowanie na zamówienie:** Oprogramowanie projektowane dla grupy użytkowników lub klientów. Przeciwieństwem jest oprogramowanie standardowe lub oprogramowanie z półki.

**oprogramowanie standardowe:** Patrz *oprogramowanie z półki*

**oprogramowanie z półki:** Oprogramowanie stanowiące produkt wytworzony na szeroki rynek, tj. dla dużej liczby klientów, które jest dostarczane wielu klientom w identycznej postaci.

**heuristic evaluation:** A static usability test technique to determine the compliance of a user interface with recognized usability principles (the so-called "heuristics").

**expected result:** The behavior predicted by the specification, or another source, of the component or system under specified conditions.

**expected outcome:** See *expected result*.

**deviation:** See *incident*.

**robustness:** The degree to which a component or system can function correctly in the presence of invalid inputs or stressful environmental conditions. [IEEE 610] See also *error-tolerance, fault-tolerance*.

**suitability:** The capability of the software product to provide an appropriate set of functions for specified tasks and user objectives. [ISO 9126] See also *functionality*.

**Defect Detection Percentage (DDP):** The number of defects found by a test phase, divided by the number found by that test phase and any other means afterwards.

**Fault Detection Percentage (FDP):** See *Defect Detection Percentage (DDP)*.

**recoverability:** The capability of the software product to re-establish a specified level of performance and recover the data directly affected in case of failure. [ISO 9126] See also *reliability*.

**item transmittal report:** See *release note*.

**bespoke software:** Software developed specifically for a set of users or customers. The opposite is off-the-shelf software.

**standard software:** See *off-the-shelf software*.

**off-the-shelf software:** A software product that is developed for the general market, i.e. for a large number of customers, and that is delivered to many customers in identical format.



**oprogramowanie:** Programy komputerowe, procedury, dokumentacja i dane mające związek z działaniem systemu komputerowego. [IEEE 610]

**software:** Computer programs, procedures, and possibly associated documentation and data pertaining to the operation of a computer system. [IEEE 610]

## P

**para definicja-użycie:** Połączenie definicji zmiennej i jej użycia. Zmienna może zostać użyta w obliczeniach (np. w mnożeniu) lub w określeniu decyzji w ścieżce (tzw. użycie "predykatywne").

**definition-use pair:** The association of the definition of a variable with the use of that variable. Variable uses include computational (e.g. multiplication) or to direct the execution of a path ("predicate" use).

**pielęgnowalność:** Łatwość, z którą oprogramowanie może być modyfikowane w celu naprawy defektów, dostosowania do nowych wymagań, modyfikowane w celu ułatwienia przyszłego utrzymania lub dostosowania do zmian zachodzących w jego środowisku. [ISO 9126]

**maintainability:** The ease with which a software product can be modified to correct defects, modified to meet new requirements, modified to make future maintenance easier, or adapted to a changed environment. [ISO 9126]

**plan projektu testowego:** Patrz *główny plan testów*

**project test plan:** See *master test plan*.

**plan testów dla fazy:** Plan testów określający jedną fazę testowania. Patrz także *plan testów*

**phase test plan:** A test plan that typically addresses one test phase. See also *test plan*.

**plan testów:** Dokument opisujący zakres, metody, zasoby oraz harmonogram zamierzonych czynności testowych. Określa - między innymi - elementy testowe, testowane cechy, zadania testowe, kto będzie te zadania wykonywał, stopień niezależności testerów, środowisko testowe, technikę projektowania testów oraz kryteria wejścia i wyjścia, przesłanki ich użycia, a także ryzyka wymagające ciągłego planowania. Jest to zapis procesu planowania testów [wg IEEE 829].

**test plan:** A document describing the scope, approach, resources and schedule of intended test activities. It identifies amongst others test items, the features to be tested, the testing tasks, who will do each task, degree of tester independence, the test environment, the test design techniques and entry and exit criteria to be used, and the rationale for their choice, and any risks requiring contingency planning. It is a record of the test planning process. [After IEEE 829]

**plan wykonania testu:** Schemat wykonania procedur testowych. Procedury testowe są zawarte w planie wykonania testu w ich kontekście i w kolejności, w jakiej mają być wykonane.

**test execution schedule:** A scheme for the execution of test procedures. The test procedures are included in the test execution schedule in their context and in the order in which they are to be executed.

**planowanie testów:** Tworzenie planów testów lub wprowadzanie do nich zmian.

**test planning:** The activity of establishing or updating a test plan.

**pluskwa:** Patrz *defekt*

**bug:** See *defect*.

**podejście do testu:** Implementacja strategii testów dla konkretnego projektu. Zwykle zawiera decyzje podjęte na podstawie celów i analizy ryzyka projektu (testowego), punkty startowe procesu testowego, techniki projektowania testu do wykorzystania, kryteria wyjścia i typy testu do wykonania.

**test approach:** The implementation of the test strategy for a specific project. It typically includes the decisions made that follow based on the (test) project's goal and the risk assessment carried out, starting points regarding the test process, the test design techniques to be applied, exit criteria

and test types to be performed.

**podstawa testu:** Wszystkie dokumenty, z których można wywnioskować wymagania dla modułu lub systemu. Dokumentacja, na bazie, której oparte są przypadki testowe. Jeśli dokument może być zmieniony tylko poprzez formalną procedurę zmiany, to baza testu nazywana jest zamrożoną bazą testu. [TMap]

**podstawa:** Specyfikacja lub oprogramowanie, które było poddane formalnemu przeglądowi lub uprzednio uzgodnione, będące odniesieniem dla dalszych prac programistycznych, które może ulec zmianie tylko przez formalną procedurę zmian. [IEEE 610]

**podstawowa przyczyna:** Przyczyna defektu, która – gdy zostanie wyeliminowana – wystąpienie tego typu defektu maleje lub jest usunięte [CMMI]

**podstawowe testowanie porównawcze:** czarno-skrzynkowa technika projektowania przypadków testowych, w której przypadki testowe są projektowane w ten sposób, by wykonać kombinację wejść wykorzystując metodę testowania pokrycia warunków

**podstawowy zestaw testów:** Zestaw przypadków testowych powstały na podstawie wewnętrznej struktury modułu lub specyfikacji, Zapewnia on 100% pokrycia rozgałęzień.

**podścieżka:** Ciąg wykonywalnych instrukcji wewnątrz modułu.

**podzbiór równoważności:** Podzbiór dziedziny danych wejściowych lub wyjściowych, dla którego zakłada się, na podstawie specyfikacji, że zachowanie modułu lub systemu jest takie samo.

**podział na podzbiory równoważności:** Technika projektowania przypadków testowych metodą czarnej skrzynki, w której przypadki testowe projektowane są tak, aby użyć elementów z podzbiorów równoważności. W szczególności przypadki testowe projektowane są tak, aby pokryć każdy podzbiór równoważności co najmniej

**test basis:** All documents from which the requirements of a component or system can be inferred. The documentation on which the test cases are based. If a document can be amended only by way of formal amendment procedure, then the test basis is called a frozen test basis. [After TMap]

**baseline:** A specification or software product that has been formally reviewed or agreed upon, that thereafter serves as the basis for further development, and that can be changed only through a formal change control process. [After IEEE 610]

**root cause:** A source of a defect such that if it is removed, the occurrence of the defect type is decreased or removed. [CMMI].

**elementary comparison testing:** A black box test design technique in which test cases are designed to execute combination of inputs using the concept of condition determination coverage [TMap]

**basis test set:** A set of test cases derived from the internal structure of a component or specification to ensure that 100% of a specified coverage criterion will be achieved.

**subpath:** A sequence of executable statements within a component.

**equivalence partition:** A portion of an input or output domain for which the behavior of a component or system is assumed to be the same, based on the specification.

**equivalence partitioning:** A black box test design technique in which test cases are designed to execute representatives from equivalence partitions. In principle test cases are designed to cover each partition at least once.

raz (by z każdego podzbioru równoważności wziąć jednego reprezentanta).

**pokrycie decyzji:** Odsetek możliwych wyników decyzji, które zostały przetestowane przez zestaw testowy; 100% pokrycia decyzji jest równoważny 100% pokrycia gałęzi oraz implikuje 100% pokrycia linii kodu

**pokrycie instrukcji kodu:** Odsetek instrukcji wykonywalnych, które zostały przetestowane przez zestaw testowy.

**pokrycie kodu:** Metoda analityczna, określająca które części programu zostały wykonane (pokryte) przez zestaw testowy, oraz które części nie zostały wykonane, np. pokrycie instrukcji kodu, pokrycie decyzji, pokrycie warunków.

**pokrycie kombinacji warunków w decyzjach:** Patrz *pokrycie wielokrotnych warunków*

**pokrycie kombinacji warunków:** Patrz *pokrycie wielokrotnych warunków*

**pokrycie LSKiS:** Odsetek LSKiS modułu, które zostały wykonane przez zestaw przypadków testowych. Pokrycie 100% LSKiS oznacza sprawdzenie 100% decyzji.

**pokrycie N-przejsć:** Odsetek sekwencji N+1-przejsć, które były wykonane przez zestaw testowy. [Chow]

**pokrycie podzbiorów równoważności:** Odsetek podzbiorów równoważności, które zostały użyte przez zestaw testowy.

**pokrycie przepływu danych:** Odsetek par definicja - użycie zmiennej, które zostały sprawdzone przez zestaw testowy.

**pokrycie rozgałęzień** Odsetek rozgałęzień sprawdzony przez zestaw przypadków testowych. 100% pokrycie rozgałęzień implikuje 100% pokrycie decyzji i 100% pokrycie stanów.

**pokrycie strukturalne:** Miary pokrycia oparte o wewnętrzną strukturę modułu lub systemu.

**pokrycie ścieżek:** Odsetek ścieżek w module wykonanych przez zestaw testowy. Realizacja 100% pokrycia ścieżek oznacza

**decision coverage:** The percentage of decision outcomes that have been exercised by a test suite. 100% decision coverage implies both 100% branch coverage and 100% statement coverage.

**statement coverage:** The percentage of executable statements that have been exercised by a test suite.

**code coverage:** An analysis method that determines which parts of the software have been executed (covered) by the test suite and which parts have not been executed, e.g. statement coverage, decision coverage or condition coverage.

**branch condition combination coverage:** See *multiple condition coverage*.

**condition combination coverage:** See *multiple condition coverage*.

**LCSAJ coverage:** The percentage of LCSAJs of a component that have been exercised by a test suite. 100% LCSAJ coverage implies 100% decision coverage.

**N-switch coverage:** The percentage of sequences of N+1 transitions that have been exercised by a test suite. [Chow]

**equivalence partition coverage:** The percentage of equivalence partitions that have been exercised by a test suite.

**data flow coverage:** The percentage of definition-use pairs that have been exercised by a test suite.

**branch coverage:** The percentage of branches that have been exercised by a test suite. 100% branch coverage implies both 100% decision coverage and 100% statement coverage.

**structural coverage:** Coverage measures based on the internal structure of a component or system.

**path coverage:** The percentage of paths that have been exercised by a test suite. 100% path coverage implies 100% LCSAJ

100% pokrycie LSKiS.

coverage.

**pokrycie testowe:** Patrz *pokrycie*.

**test coverage:** See *coverage*.

**pokrycie wartości brzegowych:** Odsetek wartości brzegowych, który został sprawdzony przez zestaw przypadków testowych.

**boundary value coverage:** The percentage of boundary values that have been exercised by a test suite.

**pokrycie warunków decyzji:** Odsetek wszystkich możliwych wyników warunków prostych w decyzjach, które zostały sprawdzone przez zestaw testowy; 100% pokrycia warunków decyzji implikuje 100% pokrycia warunków oraz 100% pokrycia decyzji

**decision condition coverage:** The percentage of all condition outcomes and decision outcomes that have been exercised by a test suite. 100% decision condition coverage implies both 100% condition coverage and 100% decision coverage.

**pokrycie warunków znaczących:** Białoskrzynkowa technika projektowania przypadków testowych, w której przypadki testowe są projektowane tak by sprawdzić wszystkie wartości warunków prostych, które niezależnie od siebie wpływają na wynik decyzji, które zostały sprawdzone przez zestaw testów; 100% pokrycia warunków znaczących daje 100% pokrycia warunków w decyzji

**condition determination testing:** A white box test design technique in which test cases are independently affect a decision outcome that have been exercised by a test case suite. 100 % condition determination coverage implies 100 % decision condition coverage

**pokrycie warunków:** Odsetek pokrycia warunków, jaki został sprawdzony przez zestaw testów. 100% pokrycia warunków wymaga, aby każdy pojedynczy warunek każdego wyrażenia decyzyjnego został przetestowany zarówno dla wartości Prawda jak i Fałsz

**condition coverage:** The percentage of condition outcomes that have been exercised by a test suite. 100% condition coverage requires each single condition in every decision statement to be tested as True and False.

**pokrycie warunku decyzji:** Patrz *pokrycie warunków*.

**branch condition coverage:** See *condition coverage*.

**pokrycie wielokrotnych warunków w decyzji:** patrz *pokrycie zmodyfikowanych warunków*

**modified multiple condition coverage:** See *condition determination coverage*.

**pokrycie wielokrotnych warunków:** Odsetek kombinacji wszystkich wyjść warunków prostych w jednej instrukcji, które zostały sprawdzone przez zestaw testowy. 100% pokrycie wielokrotnych warunków oznacza 100% pokrycie znaczących warunków.

**multiple condition coverage:** The percentage of combinations of all single condition outcomes within one statement that have been exercised by a test suite. 100% multiple condition coverage implies 100% condition determination coverage.

**pokrycie zmodyfikowanych warunków w decyzji:** Patrz *pokrycie zmodyfikowanych warunków*

**modified condition decision coverage:** See *condition determination coverage*.

**pokrycie zmodyfikowanych warunków:** Odsetek wszystkich wartości warunków prostych, które niezależnie od siebie

**condition determination coverage:** The percentage of all single condition outcomes that independently affect a decision outcome

wpływają na wynik decyzji, które zostały sprawdzone przez zestaw testowy; 100% pokrycia warunków znaczących daje 100% pokrycia warunków w decyzji

**pokrycie:** Stopień, wyrażany w procentach, w jakim zakresie zestaw testowy wykorzystał przedmiot pokrycia.

**polityka testów:** Dokument wysokiego poziomu opisujący zasady, podejście i główne zadania organizacji dotyczące testowania.

**pomiar:** Proces przypisania liczby bądź kategorii do obiektu mający na celu opisanie danej właściwości obiektu. [ISO 14598]

**pomyłka:** Patrz *błąd*

**porównanie po wykonaniu:** Porównanie rezultatów rzeczywistych z oczekiwanymi przeprowadzane po zakończeniu działania oprogramowania.

**porównanie testowe:** Proces identyfikowania różnic pomiędzy rzeczywistymi rezultatami wyprodukowanymi przez testowany moduł lub system a oczekiwanymi rezultatami dla testu. Porównanie testowe może być przeprowadzone podczas wykonania testu (porównanie dynamiczne) bądź po wykonaniu testu.

**posiew błędów:** Patrz *posiew usterek*

**posiew usterek:** Proces celowego dodawania znanych defektów do już istniejących w module lub systemie w celu monitorowania efektywności ich wykrywania i usuwania oraz szacowania liczby defektów niewykrytych. [IEEE 610]

**powtarzalność testów:** Atrybut testu wskazujący czy przy każdym wykonaniu testu otrzymujemy te same wyniki.

**poziom ryzyka:** Określenie istotności ryzyka zdefiniowane przez jego właściwości: wpływ i prawdopodobieństwo. Poziom ryzyka może być użyty do określenia 'poziomu' testów, które należy przeprowadzić.

**poziom testów:** Grupa czynności testowych, które są razem zorganizowane i zarządzane. Poziom testów jest powiązany z poziomami odpowiedzialności w projekcie. Przykładami poziomów testów są testy modułowe, integracyjne, systemowe i akceptacyjne.

that have been exercised by a test case suite. 100% condition determination coverage implies 100% decision condition coverage.

**coverage:** The degree, expressed as a percentage, to which a specified coverage item has been exercised by a test suite.

**test policy:** A high level document describing the principles, approach and major objectives of the organization regarding testing.

**measurement:** The process of assigning a number or category to an entity to describe an attribute of that entity. [ISO 14598]

**mistake:** See *error*.

**post-execution comparison:** Comparison of actual and expected results, performed after the software has finished running.

**test comparison:** The process of identifying differences between the actual results produced by the component or system under test and the expected results for a test. Test comparison can be performed during test execution (dynamic comparison) or after test execution.

**error seeding:** See *Fault seeding*

**fault seeding:** The process of intentionally adding known defects to those already in the component or system for the purpose of monitoring the rate of detection and removal, and estimating the number of remaining defects [IEEE 610]

**test reproduceability:** An attribute of a test indicating whether the same results are produced each time the test is executed.

**risk level:** The importance of a risk as defined by its characteristics impact and likelihood. The level of risk can be used to determine the 'level' of testing to be performed.

**test level:** A group of test activities that are organized and managed together. A test level is linked to the responsibilities in a project. Examples of test levels are component test, integration test, system test and acceptance test. [After TMap]

[TMap]

**priorytet:** Poziom (biznesowej) ważności określony dla elementu np. defektu.

**problem:** Patrz *defekt*

**procedura testowa:** Patrz *specyfikacja procedury testowej*

**proces testowy:** Podstawowy proces testowy składa się z następujących faz: planowanie testów i kierowanie nimi, analiza i projektowanie testów, implementacja i wykonanie, ocena kryteriów wyjścia, raportowanie oraz czynności związane z zakończeniem testów.

**proces:** Powiązane ze sobą działania przetwarzające wejścia w wyjścia. [ISO 12207]

**produkcyjne testy akceptacyjne:** Testowanie produkcyjne w fazie testów akceptacyjnych, zwykle przeprowadzane w środowisku produkcyjnym będącym symulacją rzeczywistego środowiska produkcyjnego. Wykonywane zazwyczaj przez operatora i/lub administratora, zorientowane na takie aspekty jak: odtwarzalność, **zużywanie się zasobów**, instalowalność i zgodność techniczną. Patrz także *testowanie produkcyjne*

**produkt:** Dowolny produkt pracy, który musi być dostarczony komuś innemu niż jego autor

**profil obciążenia:** Specyfikacja aktywności użytkowników, jakiej testowany moduł lub system może doświadczyć w trakcie użycia operacyjnego.

Profil obciążenia składa się z określonej liczby wirtualnych użytkowników, którzy przetwarzają zdefiniowany zbiór transakcji w określonym przedziale czasowym i zgodnie z predefiniowanym profilem operacyjnym. Patrz także *profil produkcyjny*

**profil produkcyjny:** Reprezentacja wyróżnionego zbioru zadań wykonywanego przez moduł lub system, prawdopodobnie oparta na zachowaniach użytkownika współdziałającego z modułem lub systemem i prawdopodobieństwie ich występowania. Zadania są raczej logiczne niż fizyczne i

**priority:** The level of (business) importance assigned to an item, e.g. defect.

**problem:** See *defect*.

**test procedure:** See *test procedure specification*.

**test process:** The fundamental test process comprises test planning and control, test analysis and design, test implementation and execution, evaluating exit criteria and reporting, and test closure activities.

**process:** A set of interrelated activities, which transform inputs into outputs. [ISO 12207]

**operational acceptance testing:** Operational testing in the acceptance test phase, typically performed in a simulated real-life operational environment by operator and/or administrator focusing on operational aspects, e.g. recoverability, resource-behavior, installability and technical compliance. See also *operational testing*,

**deliverable:** Any (work) product that must be delivered to someone other than the (work) product's author.

**load profile:** A specification of the activity which a component or system being tested may experience in production. A load profile consists of designated number of virtual users who process a defined set of transactions in a specified time period and according to a predefined operational profile. See also *operational profile*

**operational profile:** The representation of a distinct set of task performed by component or system possibly based on user behavior when interacting with the component or system and their probabilities of occurrence. A task is logical rather than physical and can be executed over several machines or be

mogą być wykonywane na wielu maszynach, lub być wykonywane w nieciągłych przedziałach czasowych.

**profilowanie wydajności:** Definicje profili użytkownika w testach wydajnościowych, obciążeniowych i/lub przeciążających. Profile powinny odzwierciedlać aktualne użycie produkcyjnego profilu modułu lub systemu, i tym samym oczekiwanego obciążenia. Patrz także *profil obciążenia*, *profil produkcyjny*.

**programowanie parami:** Metoda wytwarzania oprogramowania, w której linie kodu (produkcyjne i/ lub testowe) modułu są pisane przez dwóch programistów siedzących przy jednym komputerze. Domyślnie oznacza to odbywający się w czasie rzeczywistym przegląd kodu.

#### **projekt testu:**

1. Patrz *specyfikacja projektu testu*.
2. Proces przekształcania ogólnych celów testowania w uszczegółowione warunki testowe i przypadki testowe

**projekt:** Projekt jest to zestaw skoordynowanych i kontrolowanych aktywności o określonym czasie rozpoczęcia i zakończenia. Powoływany jest w celu osiągnięcia celu zgodnie z określonymi wymaganiami włączając w nie ograniczenia czasowe, kosztowe i zasobowe. [ISO 9000]

**projektowanie funkcjonalnych przypadków testowych:** Procedura projektowania i/lub wyboru przypadków testowych w oparciu o analizę specyfikacji funkcjonalnej modułu lub systemu bez zagładania w jego wewnętrzną strukturę. Patrz również: *czarno-skrzynkowa technika projektowania przypadków testowych*

**projektowanie przypadków testowych w oparciu o specyfikację:** Patrz *czarno-skrzynkowa technika projektowania przypadków testowych*.

**projektowanie strukturalnych przypadków testowych:** Patrz: *biało-skrzynkowe techniki projektowania*

**protokółant (skryba):** Osoba, która podczas spotkania przeglądownego rejestruje w dzienniku przeglądu każdy zgłoszony defekt lub sugestię odnośnie usprawnienia procesu. Protokółant musi zapewnić, że

executed in non-contiguous time segments.

**performance profiling:** Definitions of user profiles in performance, load and/or stress testing. Profiles should reflect actual usage on an operational profile of a component or system, and hence the expected workload. See also *load profile*, *operational profile*.

**pair programming:** A software development approach whereby lines of code (production and/or test) of a component are written by two programmers sitting at a single computer. This implicitly means ongoing real-time code reviews are performed.

#### **test design:**

1. See *test design specification*.
2. The process of transforming general testing objectives into tangible test conditions and test cases

**project:** A project is a unique set of coordinated and controlled activities with start and finish dates undertaken to achieve an objective conforming to specific requirements, including the constraints of time, cost and resources. [ISO 9000]

**functional test design technique:** Procedure to derive and/or select test cases based on an analysis of the specification of the functionality of a component or system without reference to its internal structure. See also *black box test design technique*.

**specification-based test design technique:** See *black box test design technique*.

**structural test design technique:** See *white box test design technique*.

**scribe:** The person who records each defect mentioned and any suggestions for process improvement during a review meeting, on a logging form. The scribe has to ensure that the logging form is readable and

dziennik przeglądu jest czytelny i understandable.  
zrozumiały.

**przewodzący inspekcję:** Patrz *moderator*.

**inspection leader:** See *moderator*.

**przebieg testu:** Wykonanie testu na określonej wersji przedmiotu testu.

**test run:** Execution of a test on a specific version of the test object.

**przedmiot pokrycia:** Obiekt lub właściwość wykorzystywane jako punkt wyjścia dla zaprojektowania testów pokrycia, np. klasy równoważności lub pokrycie kodu.

**coverage item:** An entity or property used as a basis for test coverage, e.g. equivalence partitions or code statements.

**przedmiot testów:** Moduł lub system podlegający testowaniu. Patrz też *element testowy*.

**test object:** The component or system to be tested. See also *test item*.

**przeгляд ad hoc:** Patrz *przeгляд nieformalny*

**ad hoc review:** See *informal review*

**przeгляд formalny:** Przeгляд charakteryzujący się udokumentowanymi procedurami i wymaganiami, np. inspekcja

**formal review:** A review characterized by documented procedures and requirements, e.g. inspection.

**przeгляд kierowniczy:** Systematyczna ocena procesów zakupu, dostawy, wytworzenia, działania lub utrzymania oprogramowania wykonywana przez albo w imieniu kierownictwa, która monitoruje postępy, określa status planów i harmonogramów, zatwierdza wymagania oraz ich alokację, ocenia skuteczność metod zarządzania mających na celu osiągnięcie zakładanych celów/funkcjonalności. [IEEE 610, IEEE 1028]

**management review:** A systematic evaluation of software acquisition, supply, development, operation, or maintenance process, performed by or on behalf of management that monitors progress, determines the status of plans and schedules, confirms requirements and their system allocation, or evaluates the effectiveness of management approaches to achieve fitness for purpose. [After IEEE 610, IEEE 1028]

**przeгляд koleżeński:** Przeгляд produktów powstałych podczas wytwarzania oprogramowania przeprowadzany przez kolegów ich twórcy mający na celu wskazanie defektów i możliwości poprawek. Przykładami przeglądów są inspekcje, przeгляд techniczny oraz przejrzanie.

**peer review:** A review of a software work product by colleagues of the producer of the product for the purpose of identifying defects and improvements. Examples are inspection, technical review and walkthrough.

**przeгляд nieformalny:** Przeгляд, który nie jest oparty na formalnej (udokumentowanej) procedurze.

**informal review:** A review not based on a formal (documented) procedure.

**przeгляд techniczny:** Dyskusja w grupie współpracowników skupiająca się na osiągnięciu porozumienia w zakresie aspektów technicznych. [Gilb i Graham, IEEE 1028] Patrz również *przeгляд koleżeński*

**technical review:** A peer group discussion activity that focuses on achieving consensus on the technical approach to be taken. [Gilb and Graham, IEEE 1028] See also *peer review*.



**przeгляд testowalności:** Szczegółowe sprawdzenie bazy testów w celu określenia czy jest ona na odpowiednim poziomie jakości, by mogła posłużyć jako dokument wejściowy dla procesu testowego. [TMap]

**przeгляд:** Ocena produktu lub statusu projektu mająca na celu stwierdzenie rozbieżności od planowanych założeń i rekomendację usprawnień. Przykłady: przeгляд kierowniczy, przeгляд nieformalny, przeгляд techniczny, inspekcja, przejrzenie. [IEEE 1028]

**przeглядający:** Osoba zaangażowana w przeгляд, która identyfikuje i opisuje odstępstwa w przeглядanym produkcie lub projekcie. Przeглядający mogą być dobierani tak, aby reprezentować różne punkty widzenia i pełnić różne role w procesie przeglądu.

**przejrzenie ustrukturalizowane:** Patrz *przejrzenie*.

**przejrzenie:** Przedstawienie przez autora, krok po kroku, dokumentu w celu zebrania informacji i ustalenia wspólnego rozumienia jego zawartości. [Freedman i Weinberg, IEEE 1028] Patrz *przeгляд koleżeński*

**przekroczenie bufora:** Wyjątek dostępu do pamięci na skutek usiłowania umieszczenia przez proces danych poza granicami wyznaczonego bufora. W rezultacie nadpisane zostają sąsiednie obszary pamięci. Patrz także *bufor*

**przenaszalność:** Łatwość z jaką oprogramowanie może być przeniesione z jednego środowiska sprzętowego lub programowego do innego środowiska. [ISO 9126]

**przeplływ danych:** Abstrakcyjna reprezentacja sekwencji i możliwych zmian stanu obiektu danych, gdzie dostępne stany obiektu to utworzenie, użycie lub usunięcie. [Beizer]

**przeplływ sterowania:** Sekwencja zdarzeń (w postaci ścieżki) mająca miejsce podczas pracy modułu lub systemu.

**przetestowany:** Element oprogramowania określa się jako przetestowany przez przypadek testowy, kiedy wartość wejściowa powoduje wykonanie tego

**testability review:** A detailed check of the test basis to determine whether the test basis is at an adequate quality level to act as an input document for the test process. [After TMap]

**review:** An evaluation of a product or project status to ascertain discrepancies from planned results and to recommend improvements. Examples include management review, informal review, technical review, inspection, and walkthrough. [After IEEE 1028]

**reviewer:** The person involved in the review that identifies and describes anomalies in the product or project under review. Reviewers can be chosen to represent different viewpoints and roles in the review process.

**structured walkthrough:** See *walkthrough*.

**walkthrough:** A step-by-step presentation by the author of a document in order to gather information and to establish a common understanding of its content. [Freedman and Weinberg, IEEE 1028] See also *peer review*.

**buffer overflow:** Memory access exception due to the attempt by a process to store data beyond the boundaries of a fixed length buffer, resulting in overwrite of adjacent memory areas, See also *buffer*

**portability:** The ease with which the software product can be transferred from one hardware or software environment to another. [ISO 9126]

**data flow:** An abstract representation of the sequence and possible changes of the state of data objects, where the state of an object is any of: creation, usage, or destruction. [Beizer]

**control flow:** A sequence of events (paths) in the execution through a component or system.

**exercised:** A program element is said to be exercised by a test case when the input value causes the execution of that element, such as a statement, decision, or other structural

elementu, np. instrukcji, decyzji lub innego elementu strukturalnego.

**przewidywany wynik:** Patrz *oczekiwany rezultat* **predicted outcome:** See *expected result*.

**przewodnik instalacji:** Instrukcje dostarczone na odpowiednich nośnikach, które prowadzą instalatora przez proces instalacji. Może to być podręcznik, procedura krok po kroku, kreator lub dowolny inny opis procesu. **installation guide:** Supplied instructions on any suitable media, which guides the installer through the installation process. This may be a manual guide, step-by-step procedure, installation wizard, or any other similar process description.

**przyczynowo-skutkowa tablica decyzyjna:** Patrz: *tablica decyzyjna* **cause-effect decision table:** See *decision table*.

**przypadek testowy niskiego poziomu:** Przypadek testowy z konkretnymi (na poziomie implementacji) wartościami wejściowymi i wynikami oczekiwanymi. Logiczne operatory z przypadków testowych wysokiego poziomu są zamieniane na konkretne wartości, które odpowiadają celom logicznych operatorów. Patrz również *przypadek testowy wysokiego poziomu*. **low level test case:** A test case with concrete (implementation level) values for input data and expected results. Logical operators from high level test cases are replaced by actual values that correspond to the objectives of the logical operators. See also *high level test case*.

**przypadek testowy wysokiego poziomu:** Przypadek testowy bez konkretnych (poziom implementacji) wartości danych wejściowych i oczekiwanych rezultatów. Używane są operatory logiczne; rzeczywiste wartości nie są jeszcze zdefiniowane i/ lub dostępne. Patrz również *przypadek testowy niskiego poziomu*. **high level test case:** A test case without concrete (implementation level) values for input data and expected results. Logical operators are used; instances of the actual values are not yet defined and/or available. See also *low level test case*.

**przypadek testowy:** Zbiór danych wejściowych, wstępnych warunków wykonania, oczekiwanych rezultatów i końcowych warunków wykonania opracowany w określonym celu lub dla warunku testowego, jak wykonanie pewnej ścieżki programu lub zweryfikowanie zgodności z konkretnym wymaganiem. [IEEE 610] **test case:** A set of input values, execution preconditions, expected results and execution postconditions, developed for a particular objective or test condition, such as to exercise a particular program path or to verify compliance with a specific requirement. [After IEEE 610]

**przypadek użycia:** Ciąg transakcji w dialogu pomiędzy użytkownikiem a systemem z namacalnym rezultatem. **use case:** A sequence of transactions in a dialogue between a user and the system with a tangible result.

**przyrostowy model wytwarzania:** Model wytwarzania oprogramowania, w którym przedsięwzięcie realizowane jest przyrostowo, w cyklach, z których każdy dostarcza część funkcjonalności z całego zbioru wymagań. Wymagania są porządkowane według priorytetów i realizowane w tej kolejności w odpowiednich przyrostach. W niektórych, **incremental development model:** A development life cycle where a project is broken into a series of increments, each of which delivers a portion of the functionality in the overall project requirements. The requirements are prioritized and delivered in priority order in the appropriate increment. In some (but not all) versions of this life cycle model, each subproject follows a 'mini

(ale nie wszystkich) wersjach tego modelu V-model' with its own design, coding and wytwarzania, każde pod-przedsięwzięcie jest testing phases. realizowane zgodnie z modelem V z fazami: projektowania, kodowania i testowania.

**pseudolosowy:** Ciąg, który wygląda na losowy, ale w rzeczywistości został wygenerowany według pewnej uprzednio przygotowanej sekwencji.

**punkt startu:** Pierwsze wyrażenie wykonywalne wewnątrz modułu.

**punkt wyjścia:** Ostatnie wyrażenie wykonywalne wewnątrz modułu.

**pseudo-random:** A series which appears to be random but is in fact generated according to some prearranged sequence.

**entry point:** The first executable statement within a component.

**exit point:** The last executable statement within a component.

## R

**rada kontroli zmian:** Grupa ludzi odpowiedzialna za ocenę oraz udzielenie lub nieudzielenie zgody na wykonanie zmian w elementach konfiguracji oraz za zapewnienie wykonania zaaprobowanych zmian [IEEE 610]

**raport do wydania:** Patrz *uwaga do wydania*

**raport o postępie testów:** Dokument zawierający podsumowanie aktywności testowych i osiągniętych wyników, tworzony regularnie, by zaraportować postęp prac testowych w stosunku do założeń (takich jak oryginalny plan testów) i przedstawiający ryzyka i alternatywy wymagające podjęcia decyzji zarządczych

**raport oceny testu:** Dokument tworzony na końcu procesu testowego, podsumowujący wszystkie aktywności testowe i wyniki. Zawiera również ocenę procesu testowego i wyciągnięte wnioski.

**raport z testów:** Patrz *sumaryczny raport z testów*

**rejestracja incydentu:** Dokumentowanie szczegółów zdarzenia, które wystąpiło, na przykład podczas testów.

**rejestrator:** Patrz *protokólant*

**reprezentacja ciągła:** Struktura modelu dojrzałości organizacyjnej, w której poziomy dojrzałości określają zalecaną kolejność usprawnień procesów w określonym obszarze procesowym. [CMMI]

**configuration control board (CCB):** A group of people responsible for evaluating and approving or disapproving proposed changes to configuration items, and for ensuring implementation of approved changes. [IEEE 610]

**test item transmittal report:** See *release note*.

**test progress report:** A document summarizing testing activities and results, produced at regular intervals, to report progress of testing activities against a baseline (such as the original test plan) and to communicate risks and alternatives requiring a decision to management.

**test evaluation report:** A document produced at the end of the test process summarizing all testing activities and results. It also contains an evaluation of the test process and lessons learned.

**test report:** See *test summary report*.

**incident logging:** Recording the details of any incident that occurred, e.g. during testing.

**recorder:** See *scribe*.

**continous representation:** A capability maturity model structure wherein capability levels provide a recommended order for approaching process improvement within specified process area. {CMMI}

**reprezentacja etapowa:** Struktura modelu, w którym osiągnięcie celów zbioru zakresów procesów określa poziom dojrzałości; każdy poziom stanowi podstawę dla następných poziomów [CMMI]

**retesty:** Testowanie polegające na uruchomieniu przypadków testowych, które podczas ostatniego uruchomienia wykryły błędy, w celu sprawdzenia poprawności naprawy.

**rezultat testu:** Patrz *rezultat*

**rezultat:** Konsekwencja/ wynik wykonania testu. Zawiera on wyjścia na ekran, zmiany danych, raporty oraz wysyłane komunikaty. Patrz także *rzeczywisty rezultat oczekiwany rezultat*

**rozgałęzienie:** Instrukcja podstawowa, używana do sterowania w programie, gdy jedna z dwóch, lub więcej alternatywných ścieżek w programie jest dostępnych, np. instrukcje case, skok, go to, if-then-else.

**ryzyko produktowe:** Ryzyko bezpośrednio powiązane z przedmiotem testów. Patrz także *ryzyko*

**ryzyko projektowe:** Ryzyko związane z zarządzaniem i kontrolą projektu (testowego), np. braki zasobowe, rygorystyczny harmonogram, zmieniające się wymagania itp. Patrz także *ryzyko*

**ryzyko:** Czynn timer, który w przyszłości może skutkować negatywnymi konsekwencjami; zazwyczaj opisywany jako wpływ oraz prawdopodobieństwo.

**rzeczywisty**

Wytworzone/zaobserwowane zachowanie się modułu lub systemu, podczas gdy ten moduł lub system jest testowany

**staged representation:** A model structure wherein attaining the goals of a set of process areas established a maturity level; each level builds a foundation for subsequent levels. [CMMI]

**re-testing:** Testing that runs test cases that failed the last time they were run, in order to verify the success of corrective actions.

**test result:** See *result*.

**result:** The consequence/outcome of the execution of a test. It includes outputs to screens, changes to data, reports, and communication messages sent out. See also *actual result, expected result*.

**branch:** A basic block that can be selected for execution based on a program construct in which one of two or more alternative program paths is available, e.g. case, jump, go to, if-then-else.

**product risk:** A risk directly related to the test object. See also *risk*.

**project risk:** A risk related to management and control of the (test) project, e.g. lack of staffing, strict deadlines, changing requirements, etc. See also *risk*.

**risk:** A factor that could result in future negative consequences; usually expressed as impact and likelihood.

**actual result:** The behavior produced/observed when a component or system is tested

## S

**scenariusz testowy:** Patrz *specyfikacja procedury testowej*

**sesja testowa:** W testowaniu eksploracyjnym, nieprzerywalny okres czasu poświęcony testowaniu. Każda sesja zorientowana na status, ale tester może w tym czasie także odkrywać nowe możliwości lub kwestie. Tester tworzy i wykonuje przypadki testowe w locie oraz zapisuje ich postęp. Patrz także *testowanie eksploracyjne*

**test scenario:** See *test procedure specification*.

**test session:** In exploratory testing, an uninterrupted period of time spent testing. Each session is focused on a charter, but testers can also explore new opportunities or issues during this time. The tester creates and executes test cases on the fly and records their progress. See also *exploratory testing*.

**skala pomiaru:** Skala ograniczająca typ analiz, które mogą być wykonane na danych. [ISO 14598]

**skalowalność:** Zdolność oprogramowania do bycia rozbudowywanym w celu obsłużenia wzrastającego obciążenia. [Gerrard]

**składowanie danych:** Patrz: *zużycie zasobów*

**skrypt testowy:** Powszechnie używana nazwa specyfikacji procedury testowej, zwłaszcza automatycznej.

**specyfikacja modułu:** opis funkcji modułu w formie zależności między wynikiem jego działania a danymi wejściowymi dla określonych warunków wraz z wymaganiami niefunkcjonalnymi dla tego modułu (np. dotyczącymi zużycia zasobów)

**specyfikacja procedury testowej:** Dokument określający ciąg akcji umożliwiający wykonanie testu. Znana także jako skrypt testowy lub manualny skrypt testowy [IEEE 829]

**specyfikacja projektu testu:** Dokument specyfikujący warunki testowe (elementy pokrycia) dla elementu testowego, szczegółowe podejście do testu oraz identyfikujący powiązane przypadki testowe wysokiego poziomu. [IEEE 829]

**specyfikacja przypadków testowych:** Dokument specyfikujący zbiór przypadków testowych (cel, wejścia, akcje testowe, oczekiwane rezultaty i wstępne warunki wykonania) dla elementu testowego. [IEEE 829]

**specyfikacja testowa:** Dokument zawierający specyfikację projektu testów, specyfikacje przypadków testowych i/lub specyfikację procedury testowej

**specyfikacja:** Dokument, który określa, najlepiej w kompletny, precyzyjny i możliwy do weryfikacji sposób, wymagania, projekt, zachowanie lub inne właściwości modułu lub systemu, oraz często procedury sprawdzania, czy te warunki zostały spełnione. [IEEE 610]

**spotkanie post-projektowe:** Spotkanie na końcu projektu, podczas którego członkowie zespołu projektowego oceniają projekt i wyciągają wnioski, które mogą być

**measurement scale:** A scale that constrains the type of data analysis that can be performed on it. [ISO 14598]

**scalability:** The capability of the software product to be upgraded to accommodate increased loads. [After Gerrard]

**storage:** See *resource utilization*.

**test script:** Commonly used to refer to a test procedure specification, especially an automated one.

**component specification:** A description of a component's function in terms of its output values for specified input values under specified conditions, and required non-functional behavior (e.g. resource-utilization).

**test procedure specification:** A document specifying a sequence of actions for the execution of a test. Also known as test script or manual test script. [After IEEE 829]

**test design specification:** A document specifying the test conditions (coverage items) for a test item, the detailed test approach and identifying the associated high level test cases. [After IEEE 829]

**test case specification:** A document specifying a set of test cases (objective, inputs, test actions, expected results, and execution preconditions) for a test item. [After IEEE 829]

**test specification:** A document that consists of a test design specification, test case specification and/or test procedure specification.

**specification:** A document that specifies, ideally in a complete, precise and verifiable manner, the requirements, design, behavior, or other characteristics of a component or system, and, often, the procedures for determining whether these provisions have been satisfied. [After IEEE 610]

**retrospective meeting:** A meeting at the end of a project during which the project team members evaluate the project and learn lessons that can be applied to the next

wykorzystane w następnym projekcie

project

**spójność:** Stopień jednolitości, standaryzacji oraz brak sprzeczności pomiędzy dokumentami oraz częściami modułu lub systemu. [IEEE 610]

**sprawdzanie biurkowe (ręczne):** Testowanie oprogramowania lub specyfikacji poprzez manualną symulację jego wykonania. Patrz też: *analiza statyczna*.

**stabilność:** zdolność produktu oprogramowania do unikania niespodziewanych zachowań z modyfikacji w oprogramowaniu [ISO 9126]. Patrz także *pielęgnowalność*

**stan awarii:** Fizyczny lub funkcjonalny przejaw awarii. Na przykład system w stanie awarii może charakteryzować się powolnym działaniem, błędnymi wyjściami lub całkowitym zaprzestaniem działania. [IEEE 610]

**statut testu:** Deklaracja celów testu oraz ewentualnie pomysłów na testowanie. Statuty testów są przykładowo często używane w testowaniu eksploracyjnym. Patrz też *testowanie eksploracyjne*.

**sterownik testowy:** Patrz *sterownik*

**sterownik:** Moduł oprogramowania lub narzędzie testowe, które zastępuje moduł kontrolujący lub wywołujący funkcje testowanego modułu lub systemu [TMap]

**strategia testów:** Opis, na wysokim poziomie, poziomów testów, które mają być wykonane oraz testów w ramach tych poziomów dla organizacji lub programu (jeden lub kilka projektów).

**struktura do testów jednostkowych:** Narzędzie które dostarcza środowisko do testów jednostkowych lub modułowych; w takim środowisku moduł może być testowany niezależnie (w izolacji) lub z użyciem odpowiednich zaślepek i sterowników. Dostarcza również innego rodzaju wsparcia dla programistów np. możliwość debugowania [Graham]

**sumaryczny raport z testów:** Sumaryczny dokument przedstawiający działania testowe i ich rezultaty. Zawiera także ocenę testowanych elementów pod względem

**consistency:** The degree of uniformity, standardization, and freedom from contradiction among the documents or parts of a component or system. [IEEE 610]

**desk checking:** Testing of software or specification by manual simulation of its execution. See also *static analysis*.

**stability:** The capability of the software product to avoid unexpected effects from modifications in the software. [ISO 9126] See also *maintainability*.

**failure mode:** The physical or functional manifestation of a failure. For example, a system in failure mode may be characterized by slow operation, incorrect outputs, or complete termination of execution. [IEEE 610]

**test charter:** A statement of test objectives, and possibly test ideas on how to test. Test charters are for example often used in exploratory testing. See also *exploratory testing*.

**test driver:** See *driver*.

**driver:** A software component or test tool that replaces a component that takes care of the control and/or the calling of a component or system. [After TMap]

**test strategy:** A high-level description of the test levels to be performed and the testing within those levels for an organization or programme (one or more projects).

**unit test framework:** A tool that provides an environment for unit or component testing in which a component can be tested in isolation or with suitable stubs and drivers. It also provides other support for the developer, such as debugging capabilities. [Graham]

**test summary report:** A document summarizing testing activities and results. It also contains an evaluation of the corresponding test items against exit criteria.

zgodności z kryteriami wyjścia. [IEEE 829] [After IEEE 829]

**symulacja:** Odwzorowanie wybranych charakterystycznych zachowań jednego fizycznego lub abstrakcyjnego systemu przez inny system. [ISO 2382/1]

**symulator:** Urządzenie, program komputerowy albo system używany podczas testowania, który przy zadanym zbiorze wejść zachowuje się lub działa tak jak dany system. [Po IEEE 610, DO178b] Patrz także *emulator*.

**system krytyczny ze względów bezpieczeństwa:** System którego awaria lub nieprawidłowe działanie może skutkować śmiercią lub poważnymi obrażeniami ludzi, utratą lub poważnymi uszkodzeniami urządzeń, lub zanieczyszczeniem środowiska.

**System systemów:** mnogie heterogeniczne rozproszone systemy, które mogą być zagnieżdżone w sieciach na wielu poziomach i w wielu połączonych dziedzinach ukierunkowane na wspólne interdyscyplinarne problemy i zamierzenia o wielkiej skali

**system:** Zbiór modułów zorganizowany tak, by osiągnąć zadaną funkcjonalność. [IEEE 610]

**sytuacja testowa:** Patrz *warunek testowy*

**szacowanie testów:** Obliczona aproksymacja wyniku (np. praca, data zakończenia, związane koszty, ilość przypadków testowych itp.), która jest użyteczna nawet jeśli dane wejściowe są niekompletne, niepewne lub zakłócone

**simulation:** The representation of selected behavioral characteristics of one physical or abstract system by another system. [ISO 2382/1]

**simulator:** A device, computer program or system used during testing, which behaves or operates like a given system when provided with a set of controlled inputs. [After IEEE 610, DO178b] See also *emulator*.

**safety critical system:** A system whose failure or malfunction may result in death or serious injury to people, or loss or severe damage to equipment, or environmental harm.

**system of systems:** Multiple heterogeneous, distributed systems that are embedded in networks at multiple levels and in multiple domains interconnected addressing a large scale inter-disciplinary common problems and purposes.

**system:** A collection of components organized to accomplish a specific function or set of functions. [IEEE 610]

**test situation:** See *test condition*.

**test estimation:** The calculated approximation of a result (e.g. effort spent, completion date, costs involved, number of test cases, etc.) which is usable even if input data may be incomplete, uncertain, or noisy.

## Ś

**ścieżka audytu:** Ścieżka wstecz, wzdłuż której śledzi się oryginalne wejście do procesu (np. dane), rozpoczynając od wyjścia procesu jako punktu startu. Ułatwia to analizę defektów i umożliwia przeprowadzenie audytu procesu [After Tmap].

**ścieżka przepływu sterowania:** Patrz *ścieżka*

**audit trail:** A path by which the original input to a process (e.g. data) can be tracked back through the process, taking the process output as a starting point. This facilitates defect analysis and allows a process audit to be carried out [After TMap]

**control flow path:** See *path*.

**ścieżka:** Sekwencja wydarzeń, np. wykonywalnych wyrażeń, w ramach modułu lub systemu począwszy od punktu wejścia do punktu wyjścia.

**śledzenie pionowe:** Śledzenie wymagań poprzez kolejne poziomy dokumentacji projektowej aż do modułów.

**śledzenie poziome:** Śledzenie odwzorowania wymagań testowych w dokumentacji testowej na kolejnych poziomach (np. plan testów, specyfikacja projektu testów, specyfikacja przypadku testowego i specyfikacja procedury testowej lub skryptu testowego).

**śledzenie:** Zdolność identyfikowania powiązanych bytów w dokumentacji i oprogramowaniu, np. wymagań i odpowiadających im testów. Patrz także: *Śledzenie poziome, śledzenie pionowe.*

**środowisko do testów modułowych:** Narzędzie, które dostarcza środowisko do testów modułowych. W czasie tych testów moduł jest testowany w izolacji lub z zaślepkami i sterownikami. Narzędzie to dostarcza także innych pomocy deweloperom, jak np. możliwość debugowania [Graham]

**środowisko produkcyjne:** Sprzęt i oprogramowanie zainstalowane w siedzibie użytkownika lub klienta, w którym moduł lub system będzie używany. W skład oprogramowania mogą wchodzić systemy operacyjne, bazy danych i inne aplikacje.

**środowisko testowe:** Środowisko, w skład którego wchodzi sprzęt, wyposażenie, symulatory, oprogramowanie oraz inne elementy wspierające, potrzebne do wykonania testu. [IEEE 610]

**path:** A sequence of events, e.g. executable statements, of a component or system from an entry point to an exit point.

**vertical traceability:** The tracing of requirements through the layers of development documentation to components.

**horizontal traceability:** The tracing of requirements for a test level through the layers of test documentation (e.g. test plan, test design specification, test case specification and test procedure specification or test script).

**traceability:** The ability to identify related items in documentation and software, such as requirements with associated tests. See also horizontal traceability, vertical traceability.

**unit test framework:** A tool that provides an environment for unit or component testing in which a component can be tested in isolation or with suitable stubs and drivers. It also provides other support for the developer, such as debugging capabilities. [Graham]

**operational environment:** Hardware and software products installed at users' or customers' sites where the component or system under test will be used. The software may include operating systems, database management systems, and other applications.

**test environment:** An environment containing hardware, instrumentation, simulators, software tools, and other support elements needed to conduct a test. [After IEEE 610]

## T

**tablica decyzyjna:** Tablica pokazująca kombinację wejść i/lub czynników (przyczyn) z odpowiadającymi im wyjściami i akcjami (skutkami), pomocna w projektowaniu przypadków testowych.

**tablica ortogonalna:** 2-wymiarowa tablica wybrana ze zbioru predefiniowanych tablic, opartych o kombinacje pewnej liczby zmiennych i zakresu wartości tych zmiennych. Każda zmienna reprezentuje

**decision table:** A table showing combinations of inputs and/or stimuli (causes) with their associated outputs and/or actions (effects), which can be used to design test cases.

**orthogonal array:** A 2-dimensional array selected from a set of predefined arrays based on the combination of the number of variables and the value ranges from those variables. Each variable represents a



kolumnę, a każda wartość tej zmiennej pojawia się w tabelicy wielokrotnie. Ilość wierszy odpowiada liczbie przypadków testowych potrzebnych do pokrycia każdej pary kombinacji wartości dwóch zmiennych.

**tablica stanów:** Tablica, która dla każdego stanu zestawia przejścia z tego stanu z każdym możliwym zdarzeniem. Obrazuje zarówno dozwolone, jak i niedozwolone przejścia.

**taksonomia błędów:** Patrz *taksonomia defektów*

**taksonomia defektów:** System (hierarchicznych) kategorii zaprojektowany w celu ułatwienia klasyfikacji defektów.

**technika czarno-skrzynkowa:** Patrz *czarno-skrzynkowa technika projektowania przypadków testowych*.

**technika oparta na defektach:** Patrz *technika projektowania testów oparta na defektach*

**technika oparta na doświadczeniu:** Patrz *technika projektowania testów oparta na doświadczeniu*

**technika oparta na specyfikacji:** Patrz *projektowanie przypadków testowych w oparciu o specyfikację*

**technika oparta o strukturę:** Patrz *biało-skrzynkowa technika projektowania przypadków testowych*

**technika projektowania przypadków testowych:** Patrz *technika projektowania testów*.

**technika projektowania testów niefunkcjonalnych:** Procedura otrzymywania i/lub wyboru przypadków testowych dla testów niefunkcjonalnych oparta na analizie specyfikacji modułu lub systemu bez odniesienia do jego wewnętrznej struktury. Patrz także *czarno-skrzynkowa technika projektowania przypadków testowych*

**technika projektowania testów oparta na defektach:** Procedura projektowania i/ lub wyboru przypadków testowych ukierunkowana na jeden lub więcej typów defektów, w której testy projektuje się na podstawie wiedzy o tych określonych typach defektów.

column, and each value for that variable occurs in the table multiple times. The number of rows represents the number of test cases required to cover each pairwise combination values of two variables

**state table:** A grid showing the resulting transitions for each state combined with each possible event, showing both valid and invalid transitions.

**bug taxonomy:** See *defect taxonomy*

**defect taxonomy:** A system of (hierarchical) categories design to be a useful aid for reproducibly classifying defects

**black-box technique:** See *black box test design technique*.

**defect based technique:** See *defect based test design technique*

**experienced-based technique:** See *experienced-based test design technique*.

**specification-based technique:** See *specification-based test design technique*.

**structure-based technique:** See *white box test design technique*

**test case design technique:** See *test design technique*.

**non-functional test design techniques:** Procedure to derive and/or select test cases for non-functional testing based on an analysis of the specification of a component or system without reference to its internal structure. See also *black box test design technique*

**defect based test design technique:** Procedure to derive and/or select test cases targeted at one or more types of defects, with test being developed from what is known about the specific defect types

**technika projektowania testów oparta na doświadczeniu:** Procedura projektowania i/lub wyboru przypadków testowych w oparciu o doświadczenie, wiedzę i intuicję testera.

**technika projektowania testów:** Procedura używana do wywodzenia i/lub wybierania przypadków testowych.

**technika specyfikacji testowej:** Patrz *technika projektowania testów*

**technika testowa:** Patrz *technika projektowania testów*

**technika wykonywania testu:** Metoda użyta do wykonania konkretnego testu, zarówno ręcznie, jak i automatycznie.

**test cyklu procesu:** Czarno-skrzynkowa technika projektowania przypadków testowych, w której testy są projektowane w celu wykonania procesu lub procedury biznesowej [TMap]. Patrz także *testowanie proceduralne*

**test dymny:** Podzbiór wszystkich zdefiniowanych/zaplanowanych przypadków testowych, które pokrywają główne funkcjonalności modułu lub systemu, mający na celu potwierdzenie, że kluczowe funkcjonalności programu działają, bez zagłębiania się w szczegóły. Codzienne budowanie i testy dymne stanowią dobre praktyki wytwarzania oprogramowania. Patrz także *test wstępny*

**test fałszywie obłany:** Test, w którym defekt został zaraportowany, jakkolwiek defekt ten wcale nie występuje

**test fałszywie zdany:** Test w którym nie zidentyfikowano obecności występującej w testowanym obiekcie usterki

**test kondycji:** Patrz *test dymny*

**test niezdany:** Patrz: *niezdany*

**test porównawczy:**

- (1) Standard, według którego mogą być przeprowadzane pomiary lub porównania
- (2) Test, który może być użyty w celu porównania systemów lub modułów ze sobą lub ze standardem jak w punkcie (1). [IEEE 610]

**test potwierdzający:** Patrz *test dymny*

**test startowy:** Patrz *test wstępny*

**experienced-based test design technique:** Procedure to derive and/or select test cases based on the tester's experience, knowledge and intuition.

**test design technique:** Procedure used to derive and/or select test cases.

**test specification technique:** See *test design technique*

**test technique:** See *test design technique*.

**test execution technique:** The method used to perform the actual test execution, either manually or automated.

**process cycle test:** A black box test design technique in which test cases are designed to execute business procedures and processes. [TMap] See also *procedure testing*

**smoke test:** A subset of all defined/planned test cases that cover the main functionality of a component or system, to ascertaining that the most crucial functions of a program work, but not bothering with finer details. A daily build and smoke test is among industry best practices. See also *intake test*.

**false-fail result:** A test result in which a defect is reported although no such defect actually

**false-pass result:** A test result which fails to identify the presence of a defect that is actually present in the test object.

**sanity test:** See *smoke test*.

**test fail:** See *fail*.

**benchmark test:** (1) A standard against which measurements or comparisons can be made. (2) A test that is be used to compare components or systems to each other or to a standard as in (1). [After IEEE 610]

**confidence test:** See *smoke test*.

**pretest:** See *intake test*.

**test wstępny:** Szczególny rodzaj testu dymnego mający na celu podjęcie decyzji czy moduł lub system jest gotowy do dalszego szczegółowego testowania. Najczęściej jest wykonywany na początku fazy wykonywania testów. Patrz również:

*test dymny*

**test** zestaw jednego lub więcej przypadków testowych. [IEEE 829]

**testalia do testów automatycznych:** Testalia używane w testowaniu automatycznym, np. skrypty testowe.

**testalia:** Wszystkie dokumenty i narzędzia (artefakty) wytworzone i używane podczas procesu testowania niezbędne do planowania, projektowania i wykonywania testów, takie jak dokumentacja, skrypty, wejścia, oczekiwane rezultaty, procedury, pliki, bazy danych, środowiska oraz każde dodatkowe oprogramowanie i narzędzia użyte podczas testowania. [Fewster i Graham]

**tester:** Wykwalifikowany profesjonalista, zaangażowany w testowanie modułu lub systemu.

**testowalność wymagań:** Stopień, w jakim wymagania są sformułowane w sposób pozwalający na zaprojektowanie testu (a w konsekwencji przypadków testowych) i wykonanie testów w celu określenia czy te wymagania są spełnione. [wg IEEE 610]

**testowalność:** Właściwość oprogramowania umożliwiająca testowanie go po zmianach [ISO 9126] Patrz *pielęgnalność*

**testowanie ad hoc:** Testy wykonywane nieformalnie; nie ma miejsca żadne formalne przygotowanie testu, nie jest użyta żadna rozpoznana technika projektowania przypadków testowych, brak jest oczekiwań co do rezultatów, wykonaniem testu kieruje dowolność.

**testowanie akceptacyjne przez użytkownika:** Patrz *testowanie akceptacyjne*

**testowanie akceptacyjne:** Testowanie formalne przeprowadzane w celu umożliwienia użytkownikowi, klientowi lub innemu uprawnionemu podmiotowi ustalenia, czy zaakceptować system lub moduł. [IDEE 610]

**intake test:** A special instance of a smoke test to decide if the component or system is ready for detailed and further testing. An intake test is typically carried out at the start of the test execution phase. See also *smoke test*.

**test:** A set of one or more test cases. [IEEE 829]

**automated testware:** Testware used in automated testing, such a tool scripts

**testware:** Artifacts produced during the test process required to plan, design, and execute tests, such as documentation, scripts, inputs, expected results, set-up and clear-up procedures, files, databases, environment, and any additional software or utilities used in testing. [After Fewster and Graham]

**tester:** A skilled professional who is involved in the testing of a component or system.

**testable requirements:** The degree to which a requirement is stated in terms that permit establishment of test designs (and subsequently test cases) and execution of tests to determine whether the requirements have been met. [After IEEE 610]

**testability:** The capability of the software product to enable modified software to be tested. [ISO 9126] See also *maintainability*.

**ad hoc testing:** Testing carried out informally: no formal test preparation takes place, no recognized design technique is used, there are no expectations for results and arbitrariness guides the test execution activity

**user acceptance testing:** See *acceptance testing*.

**acceptance testing:** Formal testing with respect to user needs, requirements and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customer or other authorized entity to

**testowanie alfa** Symulowane lub rzeczywiste testy produkcyjne przeprowadzane przez potencjalnych użytkowników lub niezależny zespół testowy, przeprowadzane u producenta, ale bez udziału wytwórców oprogramowania. Testowanie alfa jest często wykorzystywane jako forma wewnętrznych testów akceptacyjnych dla oprogramowania z półki.

**testowanie algorytmu [TMap]:** Patrz *testowanie rozgałęzień*

**testowanie na podstawie ryzyka:** Testowanie nastawione na wykrycie i dostarczenie informacji o ryzykach produktowych. [Gerrard]

**testowanie beta:** Produkcyjne testowanie przez potencjalnego i / lub istniejącego użytkownika / klienta w zewnętrznym miejscu niezwiązanym z programistami/twórcami - poza organizacją wytwórczą, w celu podjęcia decyzji, czy moduł albo system zaspokajają potrzeby użytkownika / klienta i współgra z procesami biznesowymi. Testowanie beta często jest traktowane jako forma zewnętrznych testów akceptacyjnych oprogramowania "z półki", w celu uzyskania informacji zwrotnej z rynku.

**testowanie bezpieczeństwa:** Testowanie mające na celu określenie bezpieczeństwa oprogramowania.

**testowanie biało-skrzynkowe:** Testowanie oparte na analizie wewnętrznej struktury modułu lub systemu.

**testowanie czarno-skrzynkowe:** Testowanie funkcjonalne lub niefunkcjonalne, bez odniesienia do wewnętrznej struktury modułu lub systemu.

**testowanie decyzji:** Biało-skrzynkowa technika projektowania przypadków testowych polegająca na testowaniu wyników decyzji.

**testowanie dokumentacji:** Kontrola jakości (dokładności, prawidłowości, kompletności itp.) dokumentacji, np. podręcznika użytkownika lub opisu instalacji.

determine whether or not to accept the system [after IEEE 610]

**alpha testing:** Simulated or actual operational testing by potential users/customers or an independent test team at the developers' site, but outside the development organization. Alpha testing is often employed for off-the shell software as a form of internal acceptance testing

**algorithm test [TMap]:** See *branch testing*

**risk-based testing:** Testing oriented towards exploring and providing information about product risks. [After Gerrard]

**beta testing:** Operational testing by potential and/or existing users/customers at an external site not otherwise involved with the developers, to determine whether or not a component or system satisfies the user/customer needs and fits within the business processes. Beta testing is often employed as a form of external acceptance testing for off-the-shelf software in order to acquire feedback from the market.

**safety testing:** Testing to determine the safety of a software product.

**white-box testing:** Testing based on an analysis of the internal structure of the component or system.

**black-box testing:** Testing, either functional or non-functional, without reference to the internal structure of the component or system.

**decision testing:** A white box test design technique in which test cases are designed to execute decision outcomes.

**documentation testing:** Testing the quality of the documentation, e.g. user guide or installation guide.

**testowanie dostępności:** Testowanie mające na celu określenie czy użytkownik będący osobą niepełnosprawną może używać modułu lub systemu [Gerrard]

**testowanie dynamiczne:** Testowanie, podczas którego wykonywany jest kod modułu lub systemu.

**testowanie efektywności:** Testowanie mające na celu określenie efektywności oprogramowania.

**testowanie eksploracyjne:** Nieformalna technika projektowania testów, w której tester projektuje testy w czasie, gdy są one wykonywane i wykorzystuje się informacje zdobyte podczas testowania do projektowania nowych i lepszych testów. [Bach]

**testowanie funkcjonalne:** Testowanie oparte na analizie specyfikacji funkcjonalnej modułu lub systemu. Patrz również: *Testowanie czarno-skrzynkowe*

**testowanie funkcjonalności:** Testowanie mające na celu określenie funkcjonalności oprogramowania.

**testowanie gruntowne:** Podejście do testów, w którym zestaw testowy obejmuje wszystkie kombinacje wartości wejściowych i warunków wstępnych.

**testowanie instalowalności:** Proces testowania instalowalności oprogramowania. Patrz też: *testowanie przenaszalności*.

**testowanie instrukcji:** Biało-skrzynkowa technika projektowania przypadków testowych, w której przypadki są zaprojektowane do wykonania instrukcji.

**testowanie integracji modułów:** Testy wykonywane w celu wykrycia usterek w interfejsach i interakcjach pomiędzy integrowanymi modułami

**testowanie integracji systemów:** Testowanie integracji systemów i pakietów; testowanie interfejsów z organizacjami zewnętrznymi, (np. Elektroniczna Wymiana Danych, Internet.)

**testowanie integracyjne:** Testowanie wykonywane w celu wykrycia defektów w interfejsach i interakcjach pomiędzy modułami lub systemami. Patrz również: *testowanie integracji modułów, testowanie integracji systemów*

**accessibility testing:** testing to determine the ease by which users with disabilities can use a component or system [Gerrard]

**dynamic testing:** Testing that involves the execution of the software of a component or system.

**efficiency testing:** The process of testing to determine the efficiency of a software product.

**exploratory testing:** An informal test design technique where the tester actively controls the design of the tests as those tests are performed and uses information gained while testing to design new and better tests. [After Bach]

**functional testing:** Testing based on an analysis of the specification of the functionality of a component or system. See also *black box testing*.

**functionality testing:** The process of testing to determine the functionality of a software product.

**exhaustive testing:** A test approach in which the test suite comprises all combinations of input values and preconditions.

**installability testing:** The process of testing the installability of a software product. See also *portability testing*.

**statement testing:** A white box test design technique in which test cases are designed to execute statements.

**component integration testing:** Testing performed to expose defects in the interfaces and interaction between integrated components.

**system integration testing:** Testing the integration of systems and packages; testing interfaces to external organizations (e.g. Electronic Data Interchange, Internet).

**integration testing:** Testing performed to expose defects in the interfaces and in the interactions between integrated components or systems. See also *component integration testing, system integration testing*.

**testowanie interfejsu:** Testowanie wykonywane w celu wykrycia błędów w interfejsach pomiędzy modułami.

**testowanie jednostkowe:** Patrz *testowanie modułów*.

**testowanie kombinacji warunków w decyzjach:** Patrz *testowanie wielokrotnych warunków*.

**testowanie kombinacji warunków:** Patrz *testowanie wielokrotnych warunków*

**testowanie konfiguracji:** Patrz *testowanie przenaszalności*

**testowanie konwersji:** Testowanie programów używanych do przenoszenia danych z istniejących systemów do systemów je zastępujących.

**testowanie krawędzi:** Patrz *testowanie rozgałęzień*.

**testowanie krytycznych elementów systemu (wątkowe):** Odmiana testowania integracyjnego modułów, w którym stopniowa integracja modułów następuje po zaimplementowaniu podzbioru wymagań, w przeciwieństwie do integrowania modułów według poziomów hierarchii.

**testowanie logiki:** Patrz *testowanie biało-skrzynkowe*

**testowanie losowe:** Czarno-skrzynkowa technika projektowania przypadków testowych, w której przypadki testowe są dobierane na podstawie pseudolosowego algorytmu udającego środowisko produkcyjne. Technika ta jest wykorzystywana do testów wymagań niefunkcjonalnych takich jak wydajność lub niezawodność.

**testowanie LSKiS:** Biało-skrzynkowa technika projektowania przypadków testowych, w której przypadki testowe są projektowane tak, aby wykonywały LSKiSy.

**testowanie łatwości serwisowania:** Patrz *testowanie pielęgnowalności*

**testowanie metodą "Wielkiego Wybuchu" (skokowe):** Rodzaj testowania integracyjnego, w którym elementy oprogramowania i/lub elementy sprzętu są łączone w moduł lub cały system jednocześnie, zamiast stopniowo [IEEE 610]. Patrz też: *testowanie integracyjne*

**interface testing:** An integration test type that is concerned with testing the interfaces between components or systems.

**unit testing:** See *component testing*.

**branch condition combination testing:** See *multiple condition testing*.

**condition combination testing:** See *multiple condition testing*

**configuration testing:** See *portability testing*.

**conversion testing:** Testing of software used to convert data from existing systems for use in replacement systems.

**arc testing:** See *branch testing*.

**thread testing:** A version of component integration testing where the progressive integration of components follows the implementation of subsets of the requirements, as opposed to the integration of components by levels of a hierarchy.

**logic-driven testing:** See *white box testing*.

**random testing:** A black box test design technique where test cases are selected, possibly using a pseudo-random generation algorithm, to match an operational profile. This technique can be used for testing non-functional attributes such as reliability and performance.

**LCSAJ testing:** A white box test design technique in which test cases are designed to execute LCSAJs.

**serviceability testing:** See *maintainability testing*.

**big-bang testing:** A type of integration testing in which software elements, hardware elements, or both are combined all at once into a component or an overall system, rather than in stages. [After IEEE 610] See also *integration testing*.

**testowanie międzyoperacyjności:** Proces testowania mający na celu określenie współpracy oprogramowania. Patrz także *testowanie funkcjonalności*

**testowanie na podstawie kodu:** Patrz *testowanie biało-skrzynkowe*.

**testowanie negatywne:** Testowanie, którego celem jest pokazanie, że oprogramowanie nie działa. Testowanie negatywne jest bardziej związane z postawą testerów, niż ze specyficznym podejściem czy techniką projektowania testów, np. testowanie z błędnymi wartościami wejściowymi lub wyjątkami [Beizer].

**testowanie niefunkcjonalne:** Testowanie atrybutów modułu lub systemu, które nie odnoszą się do jego funkcjonalności, np. niezawodności, efektywności, pielęgnowalności i przenaszalności.

**testowanie niezawodności:** Proces testowania mający na celu określenie niezawodności oprogramowania.

**testowanie N-przejsć:** Forma testowania przejść pomiędzy stanami, gdzie przypadki testowe są zaprojektowane tak, aby wykonać wszystkie poprawne sekwencje N+1-przejsć. [Chow] Patrz także *testowanie przejść pomiędzy stanami*

**testowanie obciążenia:** Testowanie, podczas którego system pracuje przy dużej ilości danych. Patrz także *testowanie zużycia zasobów*

**testowanie obciążeniowe:** Rodzaj testów, których zadaniem jest pomiar zachowania modułu lub systemu przy zwiększającym się obciążeniu, np. liczbie równoległe pracujących użytkowników i/lub liczbie transakcji, w celu określenia, jakie obciążenie moduł lub system jest w stanie obsłużyć. Patrz także *testowanie przeciążające*

**testowanie odporności:** Testowanie mające na celu określenie odporności oprogramowania.

**testowanie odtwarzalności:** Proces testowania mający na celu określenie odtwarzalności oprogramowania. Patrz także *testowanie niezawodności*

**testowanie odzyskiwania:** Patrz *testowanie odtwarzalności*

**interoperability testing:** The process of testing to determine the interoperability of a software product. See also *functionality testing*.

**code-based testing:** See *white box testing*.

**negative testing:** Tests aimed at showing that a component or system does not work. Negative testing is related to the testers' attitude rather than a specific test approach or test design technique, e.g. testing with invalid input values or exceptions. [After Beizer].

**non-functional testing:** Testing the attributes of a component or system that does not relate to functionality, eg. reliability, efficiency, usability, maintainability and portability

**reliability testing:** The process of testing to determine the reliability of a software product.

**N-switch testing:** A form of state transition testing in which test cases are designed to execute all valid sequences of N+1 transitions. [Chow] See also *state transition testing*.

**volume testing:** Testing where the system is subjected to large volumes of data. See also *resource-utilization testing*.

**load testing:** A test type concerned with measuring the behavior of a component or system with increasing load, e.g. number of parallel users and/or numbers of transactions to determine what load can be handled by the component or system. See also *stress testing*.

**robustness testing:** Testing to determine the robustness of the software product.

**recoverability testing:** The process of testing to determine the recoverability of a software product. See also *reliability testing*.

**recovery testing:** See *recoverability testing*.

**testowanie oparte na projekcie:** Technika projektowania przypadków testowych na podstawie architektury lub szczegółowego projektu modułu lub systemu (np. testy wymiany danych pomiędzy modułami lub systemami).

**testowanie oparte na specyfikacji:** Patrz *projektowanie przypadków testowych metodą czarnej skrzynki*.

**testowanie oparte na specyfikacji:** Patrz *testowanie czarno-skrzynkowe*.

**testowanie oparte na wymaganiach:** Podejście do testów, w którym przypadki testowe są projektowane w oparciu o cele i warunki zawarte w wymaganiach (np. testy sprawdzające konkretne funkcje lub badające niefunkcjonalne wymagania systemu takie jak niezawodność lub użyteczność).

**testowanie oparte o słowa akcji:** Patrz *testowanie oparte o słowa kluczowe*

**testowanie oparte o słowa kluczowe:** Technika skryptowa wykorzystująca dane zapisane w plikach, które zawierają nie tylko dane testowe i oczekiwane rezultaty, ale także słowa kluczowe związane z aplikacją poddawaną testowaniu. Słowa kluczowe są interpretowane przez specjalne skrypty, które są wywoływane przez skrypt kontrolujący test. Patrz także *testowanie sterowane danymi*.

**testowanie par:** Czarno-skrzynkowa technika projektowania przypadków testowych w której przypadki testowe są projektowane tak by wykonać wszystkie możliwe dyskretne kombinacje każdej pary parametrów wejściowych

**testowanie parami:** Testowanie, w którym dwie osoby, np. dwóch testerów, programista i tester lub użytkownik końcowy i tester, pracują w celu znalezienia błędów. Zwykle podczas testowania osoby te korzystają z jednego komputera.

**testowanie pielęgnacyjne:** Testowanie zmian wdrożonego systemu lub wpływu zmienionego środowiska na wdrożony system.

**testowanie pielęgnowalności:** Proces testowania mający na celu sprawdzenie czy oprogramowanie jest pielęgnowalne.

**design-based testing:** An approach to testing in which test cases are designed based on the architecture and/or detailed design of a component or system (e.g. tests of interfaces between components or systems).

**specification-based testing:** See *black box testing*.

**specification-based testing:** See *black box testing*.

**requirements-based testing:** An approach to testing in which test cases are designed based on test objectives and test conditions derived from requirements, e.g. tests that exercise specific functions or probe non-functional attributes such as reliability or usability.

**action word driven testing:** See *keyword driven testing*

**keyword driven testing:** A scripting technique that uses data files to contain not only test data and expected results, but also keywords related to the application being tested. The keywords are interpreted by special supporting scripts that are called by the control script for the test. See also *data driven testing*.

**pairwise testing:** A black box test design technique in which test cases are design to execute all possible discrete combination of each pair of input parameters

**pair testing:** Two persons, e.g. two testers, a developer and a tester, or an end-user and a tester, working together to find defects. Typically, they share one computer and trade control of it while testing.

**maintenance testing:** Testing the changes to an operational system or the impact of a changed environment to an operational system.

**maintainability testing:** The process of testing to determine the maintainability of a software product.



**testowanie podzbiorów:** Patrz *podział na podzbiory równoważności*

**testowanie pokrycia logiki:** Patrz *testowanie biało-skrzynkowe* [Myers]

**testowanie połączenia:** Patrz *testy integracji modułów*.

**testowanie potwierdzające:** Patrz *retesty*

**testowanie proceduralne:** Testowanie oparte o procedury biznesowe użytkownika lub procedury produkcyjne by upewnić się, że moduł lub system dostarcza oczekiwaną usługę.

**testowanie produkcyjne:** Patrz *testowanie beta*

**testowanie produkcyjne:** Testowanie mające na celu ocenę modułu lub systemu w jego środowisku produkcyjnym. [IEEE 610]

**testowanie profilu operacyjnego:** Testowanie statystyczne używające modelu działania systemu tzn. zadań o krótkich czasach wykonania i prawdopodobieństwach ich użycia. [Musa]

**testowanie programu:** Patrz *testowanie modułów*

**testowanie przeciążające:** Testowanie prowadzone w celu oceny zachowania systemu lub modułu na granicy lub poza granicami wyspecyfikowanych wymagań. [IEEE 610]. Patrz też *testowanie obciążeniowe*.

**testowanie przejść pomiędzy stanami:** Czarno-skrzynkowa technika projektowania przypadków testowych, w której przypadki są projektowane tak, aby sprawdzały dozwolone oraz niedozwolone przejścia między stanami. Patrz też *testowanie N-przejść*

**testowanie przenaszalności:** Proces testowania mający na celu określenie przenaszalności oprogramowania.

**testowanie przepływu danych:** Biało-skrzynkowa metoda projektowania przypadków testowych, w której testy projektowane są w oparciu o analizę par definicja - użycie zmiennej.

**testowanie przez użytkownika:** Testowanie, w którym rzeczywisti użytkownicy są zaangażowani w ocenę użyteczności modułu lub systemu.

**partition testing:** See *equivalence partitioning*. [Beizer]

**logic-coverage testing:** See *white box testing*. [Myers]

**link testing:** See *component integration testing*.

**confirmation testing:** See *re-testing*.

**procedure testing:** Testing of user's business procedure or operational procedure to ensure that the component or system provide the expected service.

**field testing:** See *beta testing*.

**operational testing:** Testing conducted to evaluate a component or system in its operational environment. [IEEE 610]

**operational profile testing:** Statistical testing using a model of system operations (short duration tasks) and their probability of typical use. [Musa]

**program testing:** See *component testing*.

**stress testing:** Testing conducted to evaluate a system or component at or beyond the limits of its specified requirements. [IEEE 610] See also *load testing*.

**state transition testing:** A black box test design technique in which test cases are designed to execute valid and invalid state transitions. See also *N-switch testing*.

**portability testing:** The process of testing to determine the portability of a software product.

**data flow testing:** A white box test design technique in which test cases are designed to execute definition and use pairs of variables.

**user test:** A test whereby real-life users are involved to evaluate the usability of a component or system.

**testowanie przyrostowe:** Testowanie, podczas którego moduły lub systemy są integrowane i testowane po jednym lub kilka jednocześnie, dopóki wszystkie elementy nie zostaną zintegrowane i przetestowane.

**testowanie regresywne:** Ponowne przetestowanie uprzednio testowanego programu po dokonaniu w nim modyfikacji, w celu upewnienia się, że w wyniku zmian nie powstały nowe defekty lub nie ujawniły się wcześniej nie wykryte w nie zmienionej części oprogramowania. Testy takie są przeprowadzane po zmianach oprogramowania lub jego środowiska pracy.

**testowanie rozgałęzień:** Biało-skrzynkowa technika projektowania przypadków testowych, w której przypadki te projektowane są w celu wykonania rozgałęzień.

**testowanie równoległości przetwarzania:** Testy wykonywane w celu sprawdzenia jak wykonanie dwu lub więcej czynności w tym samym czasie (przez przeplatanie ich wykonania lub równoległe wykonanie) jest obsługiwane przez moduł lub system. [IEEE 610]

**testowanie różnicowe:** Testowanie, podczas którego dwie lub więcej wersji modułu albo systemu uruchamianych jest z tymi samymi danymi wejściowymi. Wyjścia są porównywane i rozbieżności są analizowane. [IEEE 610]

**testowanie skalowalności:** Testowanie mające na celu określenie skalowalności oprogramowania.

**testowanie składniowe:** Czarno-skrzynkowa technika projektowania przypadków testowych, w której przypadki testowe są zaprojektowane na podstawie definicji dziedziny danych wejściowych i/ lub dziedziny danych wyjściowych.

**testowanie składowania danych:** Patrz *testowanie zużycia zasobów*

**testowanie skrypcowe:** Wykonanie testu przeprowadzone w oparciu o uprzednio udokumentowany ciąg testów

**incremental testing:** Testing where components or systems are integrated and tested one or some at a time, until all the components or systems are integrated and tested.

**regression testing:** Testing of a previously tested program following modification to ensure that defects have not been introduced or uncovered in unchanged areas of the software, as a result of the changes made. It is performed when the software or its environment is changed.

**branch testing:** A white box test design technique in which test cases are designed to execute branches.

**concurrency testing:** Testing to determine how the occurrence of two or more activities within the same interval of time, achieved either by interleaving the activities or by simultaneous execution, is handled by the component or system. [After IEEE 610]

**back-to-back testing:** Testing in which two or more variants of a component or system are executed with the same inputs, the outputs compared, and analyzed in cases of discrepancies. [IEEE 610]

**scalability testing:** Testing to determine the scalability of the software product.

**syntax testing:** A black box test design technique in which test cases are designed based upon the definition of the input domain and/or output domain.

**storage testing:** See *resource utilization testing*.

**scripted testing:** Test execution carried out by following a previously documented sequence of tests.

**testowanie spójności baz danych:** **database integrity testing:** Testing the methods and processes used to access and manage the data(base), to ensure access methods, processes and data rules function as expected and that during access to the database, data is not corrupted or unexpectedly deleted, updated or created. Testowanie metod i procesów wykorzystywanych do czytania i zarządzania danymi oraz bazą danych w celu sprawdzenia: metod dostępu do nich, sprawdzenia procesów i poprawności funkcji i upewnienia się, że podczas dostępu do danych w bazie, dane nie ulegają zniszczeniu lub nie są tworzone, modyfikowane bądź usuwane w nieoczekiwany sposób.

**testowanie spójności danych:** Patrz *testowanie spójności baz danych* **data integrity testing:** See *database integrity testing*.

**testowanie stanów:** Patrz: *testowanie przejść pomiędzy stanami* **finite state testing:** See *state transition testing*.

**testowanie statyczne:** Testowanie modułu lub systemu na poziomie specyfikacji lub implementacji bez wykonywania tego oprogramowania, np. przeglądy lub analiza statyczna kodu. **static testing:** Testing of a component or system at specification or implementation level without execution of that software, e.g. reviews or static code analysis.

**testowanie statystyczne:** Technika projektowania przypadków testowych, w której używany jest model statystycznego rozkładu danych wejściowych do zbudowania reprezentatywnych przypadków testowych. Patrz też *testowanie profilu operacyjnego* **statistical testing:** A test design technique in which a model of the statistical distribution of the input is used to construct representative test cases. See also *operational profile testing*.

**testowanie sterowane danymi:** Technika automatyzacji testów, która polega na umieszczeniu danych testowych i oczekiwanych wyników w tabeli lub arkuszu kalkulacyjnym, tak aby jeden skrypt mógł wykonać wszystkie testy z tabeli. Testowanie sterowane danymi jest często używane jako uzupełnienie narzędzi wykonywania testów takich jak narzędzia rejestrująco-odtwarzające [Fewster i Graham]. Patrz też *testowanie oparte o słowa kluczowe* **data driven testing:** A scripting technique that stores test input and expected results in a table or spreadsheet, so that a single control script can execute all of the tests in the table. Data driven testing is often used to support the application of test execution tools such as capture/playback tools. [Fewster and Graham] See also *keyword driven testing*.

**testowanie strukturalne:** Patrz: *testowanie biało-skrzynkowe* **structural testing:** See *white box testing*.

**testowanie systemowe:** Proces testowania zintegrowanego systemu w celu sprawdzenia jego zgodności z wyspecyfikowanymi wymaganiami. [Hetzel] **system testing:** The process of testing an integrated system to verify that it meets specified requirements. [Hetzel]

**testowanie szklanej skrzynki:** Patrz: *testowanie biało-skrzynkowe* **glass box testing:** See *white box testing*.

**testowanie ścieżek:** Biało-skrzynkowa technika projektowania przypadków testowych, w której przypadki testowe **path testing:** A white box test design technique in which test cases are designed to execute paths.

projektowane są w celu wykonania ścieżek.

**testowanie uregulowań:** Patrz *testowanie zgodności* **regulation testing:** See *compliance testing*.

**testowanie użyteczności:** Testowanie mające na celu określenie, w jakim stopniu oprogramowanie jest zrozumiałe, łatwe do nauczenia, łatwe w użyciu oraz atrakcyjne dla użytkowników w określonych warunkach. [ISO 9126] **usability testing:** Testing to determine the extent to which the software product is understood, easy to learn, easy to operate and attractive to the users under specified conditions. [After ISO 9126]

**testowanie w oparciu o proces biznesowy:** Podejście w testowaniu, w którym przypadki testowe projektowane są w oparciu o opis i/lub wiedzę o procesie biznesowym. **business process-based testing:** An approach to testing in which test cases are designed based on descriptions and/or knowledge of business processes.

**testowanie w oparciu o przypadki użycia:** Czarno-skrzynkowa technika projektowania przypadków testowych, w której przypadki testowe są projektowane tak, by wykonywane były scenariusze użytkownika. **use case testing:** A black box test design technique in which test cases are designed to execute user scenarios.

**testowanie w oparciu o scenariusze użytkownika:** Patrz *testowanie w oparciu o przypadki użycia* **user scenario testing:** See *use case testing*.

**testowanie w oparciu o scenariusze:** Patrz *testowanie w oparciu o przypadki użycia* **scenario testing:** See *use case testing*.

**testowanie w oparciu o standardy:** Patrz *testowanie zgodności*. **standards testing:** See *compliance testing*.

**testowanie w oparciu o strukturę:** Patrz: *testowanie biało-skrzynkowe* **structure-based testing:** See *white-box testing*.

**testowanie w oparciu o tablicę decyzyjną:** Czarno-skrzynkowa technika projektowania przypadków testowych polegająca na sprawdzeniu działania modułu lub systemu w reakcji na kombinacje warunków wejściowych podanych w tablicy decyzyjnej [Veenendaal]. Patrz też *tablica decyzyjna* **decision table testing:** A black box test design technique in which test cases are designed to execute the combinations of inputs and/or stimuli (causes) shown in a decision table. [Veenendaal] See also *decision table*.

**testowanie wartości brzegowych:** Patrz *analiza wartości brzegowych*. **boundary value testing:** See *boundary value analysis*.

**testowanie wartości negatywnych:** Testowanie polegające na próbach podawania wartości, które powinny być odrzucane przez moduł lub system. Patrz także *tolerowanie błędów* **invalid testing:** Testing using input values that should be rejected by the component or system. See also *error tolerance*.

**testowanie warunków decyzji:** Biało-skrzynkowa technika projektowania przypadków testowych polegająca na sprawdzaniu wartości warunków i wyników decyzji **decision condition testing:** A white box test design technique in which test cases are designed to execute condition outcomes and decision outcomes.

**testowanie warunków:** Biało-skrzynkowa technika projektowania przypadków testowych polegająca na testowaniu wyniku dla różnych warunków, **condition testing:** A white box test design technique in which test cases are designed to execute condition outcomes.

**testowanie wielokrotnych warunków:** **multiple condition testing:** A white box Biało-skrzynkowa technika projektowania testów, w której przypadki testowe są projektowane tak, aby wykonane zostały kombinacje wyjść warunków prostych (w ramach instrukcji).

**testowanie współpracy:** patrz *testowanie międzyoperacyjności* **compatibility testing:** See *interoperability testing*.

**testowanie wstępujące:** Podejście przyrostowe do testowania integracyjnego, które polega na testowaniu modułów najniższego poziomu jako pierwszych, co ułatwia testowanie modułów wyższych poziomów. Proces ten jest powtarzany dopóki moduł na szczycie hierarchii nie zostanie przetestowany. Patrz *testowanie integracyjne*. **bottom-up testing:** An incremental approach to integration testing where the lowest level components are tested first, and then used to facilitate the testing of higher level components. This process is repeated until the component at the top of the hierarchy is tested. See also *integration testing*.

**testowanie wydajnościowe:** Proces testowania mający na celu określenie wydajności oprogramowania. Patrz także testowanie efektywności **performance testing:** The process of testing to determine the performance of a software product. See also *efficiency testing*.

**testowanie wyizolowane:** Testowanie poszczególnych modułów w wyizolowanym środowisku, gdzie inne moduły symulowane są przez zaślepki i sterowniki testowe. **isolation testing:** Testing of individual components in isolation from surrounding components, with surrounding components being simulated by stubs and drivers, if needed.

**testowanie wytwórcze:** Formalne lub nieformalne testowanie przeprowadzone podczas tworzenia systemu lub modułu przez jego twórcę, zwykle w środowisku wytwórczym [IEEE 610] **development testing:** Formal or informal testing conducted during the implementation of a component or system, usually in the development environment by developers. [After IEEE 610]

**testowanie zabezpieczeń:** Testowanie mające na celu określenie zabezpieczeń oprogramowania. Patrz także *testowanie funkcjonalności* **security testing:** Testing to determine the security of the software product. See also *functionality testing*.

**testowanie zgodności:** Proces testowania określający zgodność modułu albo systemu. **compliance testing:** The process of testing to determine the compliance of the component or system.

**testowanie zstępujące:** Podejście przyrostowe do testowania integracyjnego, w którym moduł na górze hierarchii jest testowany jako pierwszy, a moduły niższych rzędów są symulowane przez zaślepki. Przetestowane moduły są używane później do testowania modułów niższych rzędów. Taki proces jest powtarzany aż zostaną przetestowane moduły leżące najniżej w hierarchii. Patrz także *testowanie integracyjne* **top-down testing:** An incremental approach to integration testing where the component at the top of the component hierarchy is tested first, with lower level components being simulated by stubs. Tested components are then used to test lower level components. The process is repeated until the lowest level components have been tested. See also *integration testing*.

**testowanie zużycia zasobów:** Proces testowania mający na celu określenie poziomu zużycia zasobów wykorzystywanych przez oprogramowanie. Patrz także *testowanie efektywności*

**testowanie zwinne:** metoda testowania stosowana w projektach korzystających z metodologii zwinnych takich jak programowanie ekstremalne (XP), traktujące wytwarzanie jako klienta testowania i kładąca nacisk na metodę „najpierw przygotuj testy”. Zobacz także *wytwarzanie sterowane testami*

**testowanie:** Proces składający się z wszystkich czynności cyklu życia, zarówno statycznych jak i dynamicznych; skoncentrowany na planowaniu, przygotowaniu i ewaluacji oprogramowania oraz powiązanych produktów w celu określenia czy spełniają one wyspecyfikowane wymagania, wykazania, że są dopasowane do swoich celów oraz wykrywania usterek.

**testy akceptacyjne w środowisku użytkownika:** Testowanie akceptacyjne wykonywane przez użytkowników/ klientów w ich środowisku pracy w celu określenia czy moduł lub system spełnia potrzeby użytkownika/ klienta oraz realizuje procesy biznesowe. Najczęściej zawierają zarówno testy sprzętu, jak i oprogramowania.

**testy integracyjne wewnętrzne:** Patrz *testowanie integracji modułów*

**testy integracyjne zewnętrzne:** Patrz *testowanie integracji systemów*

**testy komponentów:** Patrz *testy modułowe*

**testy migracji:** Patrz *testy konwersji*.

**testy modułowe:** Testowanie pojedynczych modułów oprogramowania.[IEEE 610]

**testy mutacji:** Patrz *testy różnicowe*

**testy wielokrotnych zmodyfikowanych warunków w decyzji:** Patrz *pokrycie zmodyfikowanych warunków*

**testy zgodności:** Patrz *testowanie zgodności*

**testy zmodyfikowanych warunków w decyzji:** Patrz *pokrycie zmodyfikowanych warunków*

**resource utilization testing:** The process of testing to determine the resource-utilization of a software product. See also *efficiency testing*.

**agile testing:** Testing practice for a project using agile methodology, such as extreme programming (XP), treating development as the customer of testing and emphasizing the test-first design paradigm. See also *test driven development*

**testing:** The process consisting of all life cycle activities, both static and dynamic, concerned with planning, preparation and evaluation of software products and related work products to determine that they satisfy specified requirements, to demonstrate that they are fit for purpose and to detect defects.

**site acceptance testing:** Acceptance testing by users/customers at their site, to determine whether or not a component or system satisfies the user/customer needs and fits within the business processes, normally including hardware as well as software.

**integration testing in the small:** See *component integration testing*.

**integration testing in the large:** See *system integration testing*.

**component testing:** The testing of individual software components. [After IEEE 610]

**migration testing:** See *conversion testing*.

**module testing:** See *component testing*.

**mutation testing:** See *back-to-back testing*

**modified multiple condition testing:** See *condition determination coverage testing*.

**conformance testing:** See *compliance testing*.

**modified condition decision testing:** See *condition determination testing*.

**tolerowanie błędów:** Zdolność systemu lub modułu do kontynuacji prawidłowego działania mimo podania błędnych wejść. [IEEE 610]

**tolerowanie usterek:** Zdolność oprogramowania do utrzymania określonego poziomu wydajności w przypadku występowania usterek (defektów) lub naruszenia jego interfejsów. [ISO 9126] Patrz również *niezawodność, odporność*.

**tworzenie grafów przyczynowo-skutkowych:** Czarno-skrzynkowa technika projektowania przypadków testowych, w której przypadki te są projektowane na podstawie grafów przyczynowo-skutkowych. [BS 7925/2]

**typ ryzyka:** Specyficzna kategoria ryzyka, związana z typem testów, które należy przeprowadzić, jako szczególne kategorie ryzyk i świadomości jak je ograniczać i łagodzić, np. ryzyko związane z interfejsem użytkownika może być łagodzone poprzez przeprowadzenie testów użyteczności.

**typ testów:** Grupa czynności testowych nakierowanych na testowanie modułu lub systemu, skupionych na specyficznych celach, takich jak test funkcjonalny, test użyteczności, test regresyjny itp. Typ testów może być użyty na jednym lub kilku poziomach testowania. [TMap]

**error tolerance:** The ability of a system or component to continue normal operation despite the presence of erroneous inputs. [After IEEE 610].

**fault tolerance:** The capability of the software product to maintain a specified level of performance in cases of software faults (defects) or of infringement of its specified interface. [ISO 9126] See also *reliability, robustness*.

**cause-effect graphing:** A black box test design technique in which test cases are designed from cause-effect graphs. [BS 7925/2]

**risk type:** A specific category of risk that is related to the type of testing to be performed, as certain categories of risks and known to be amenable to being mitigated by certain types of testing e.g. a risk associated with the user interface may be mitigated by performing usability testing.

**test type:** A group of test activities aimed at testing a component or system focused on a specific test objective, i.e. functional test, usability test, regression test etc. A test type may take place on one or more test levels or test phases. [After TMap]

## U

**uczulanie na ścieżkę:** Wybieranie zbioru wartości wejściowych w celu wymuszenia realizacji określonej ścieżki.

**unikanie ryzyka:** Patrz *kontrola ryzyka*

**Usprawnianie Procesu Testowego (TPI - akronim od ang. Test Process Improvement):** Podstawa działań na rzecz ulepszenia procesu testowego, opisująca kluczowe elementy efektywnego procesu testowego, ze szczególnym uwzględnieniem testów systemowych i akceptacyjnych.

**usterka:** Patrz: *defekt*

**utrzymanie:** Zmiany oprogramowania po wdrożeniu produkcyjnym dokonywane w celu naprawy błędów, poprawy wydajności lub innych jego atrybutów. Zmiany mogą dotyczyć także przystosowania produktu do zmienionego środowiska [IEEE 1219]

**path sensitizing:** Choosing a set of input values to force the execution of a given path.

**risk mitigation:** See *risk control*.

**Test Process Improvement (TPI):** A continuous framework for test process improvement that describes the key elements of an effective test process, especially targeted at system testing and acceptance testing.

**fault:** See *defect*.

**maintenance:** Modification of a software product after delivery to correct defects, to improve performance or other attributes, or to adapt the product to a modified environment. [IEEE 1219]

**uwaga do wydania:** Dokument opisujący elementy testowe, ich konfiguracje, status oraz inne ważne informacje przekazywane na początku fazy testowania od programis [IEEE 829]

**użyteczność:** Zdolność oprogramowania do bycia zrozumiałym, łatwym w nauce, używanym i atrakcyjnym dla użytkownika, gdy oprogramowanie to jest używane w określonych warunkach. [ISO 9126]

**release note:** A document identifying test items, their configuration, current status and other delivery information delivered by development to testing, and possibly other stakeholders, at the start of a test execution phase. [After IEEE 829]

**usability:** The capability of the software to be understood, learned, used and attractive to the user when used under specified conditions. [ISO 9126]

## W

**walidacja:** Określenie poprawności produktów procesu tworzenia oprogramowania pod względem spełnienia potrzeb i wymagań użytkownika. [ISO 9000]

**wartość brzegowa:** Wartość wejścia lub wyjścia, która jest na granicy klas równoważności lub jest w najbliższym przyrostowym sąsiedztwie tej granicy. Na przykład wartość minimalna i maksymalna zakresu.

**wartość warunku** Wyliczenie wartości warunku: Prawda albo Fałsz.

**wartość wejścia:** Wartość danej wejściowej. Patrz również *wejście*

**wartość wyjścia:** Wartość danej wyjściowej. Patrz *wyjście*.

**warunek rozgałęzienia:** Patrz *warunek*

**warunek testowy:** Element lub zdarzenie modułu lub systemu, który może być zweryfikowany przez jeden lub więcej przypadków testowych, np. funkcja, transakcja, cecha, atrybut jakości lub element struktury.

**warunek wielokrotny:** Patrz *warunek złożony*

**warunek wstępny:** Uwarunkowania środowiska i stanu, jakie muszą być spełnione zanim moduł lub system będzie mógł być uruchomiony przez określony test lub procedurę testową.

**warunek wyjściowy:** Uwarunkowania stanu lub środowiska, które muszą być spełnione po wykonaniu testu lub procedury testowej.

**warunek złożony:** Dwa lub więcej prostych warunków połączonych spójnikami logicznymi (AND, OR lub XOR), np. "a>b AND c>1000"

**validation:** Confirmation by examination and through provision of objective evidence that the requirements for a specific intended use or application have been fulfilled. [ISO 9000]

**boundary value:** An input value or output value which is on the edge of an equivalence partition or at the smallest incremental distance on either side of an edge, for example the minimum or maximum value of a range.

**condition outcome:** The evaluation of a condition to True or False.

**input value:** An instance of an input. See also *input*.

**output value:** An instance of an output. See also *output*.

**branch condition:** See *condition*.

**test condition:** An item or event of a component or system that could be verified by one or more test cases, e.g. a function, transaction, feature, quality attribute, or structural element.

**multiple condition:** See *compound condition*.

**precondition:** Environmental and state conditions that must be fulfilled before the component or system can be executed with a particular test or test procedure.

**postcondition:** Environmental and state conditions that must be fulfilled after the execution of a test or test procedure.

**compound condition:** Two or more single conditions joined by means of a logical operator (AND, OR or XOR), e.g. 'A>B AND C>1000'.



**warunek:** Wyrażenie logiczne, którego wartością może być Prawda albo Fałsz, na przykład  $A > B$ . Patrz także *warunek testowy*.

**ważność:** Stopień wpływu defektu na rozwój lub działanie modułu lub systemu. [IEEE 610]

**wejście do testów:** Dane otrzymywane z zewnętrznego źródła przez przedmiot testu podczas wykonywania testu. Źródłem zewnętrznym może być sprzęt, oprogramowanie lub człowiek.

**wejście:** Zmienna (przechowywana wewnątrz modułu albo poza nim), która jest odczytywana przez moduł.

**weryfikacja:** Proces kontroli polegający na sprawdzeniu, czy produkty danego etapu produkcji spełniają zdefiniowane warunki. [ISO 9000]

**Wide Band Delphi:** Technika estymacji pracochłonności (wariant metody delfickiej), bazująca na wiedzy eksperckiej, polegająca na dokładnym szacowaniu w oparciu o zbiorową wiedzę członków zespołu eksperckiego.

**właściwość jakościowa:** Patrz *atribut jakościowy*

**wskaźnik wydajności:** Metryka wysokiego poziomu określająca poziom skuteczności i/lub efektywności wykorzystywana do śledzenia i kontroli wytwarzania, np. czas realizacji zamówionego oprogramowania [CMMI]

**wskaźnik:** Dana która określa położenie innej zmiennej, np. zmienna, która określa adres następnego rekordu pracownika, który to rekord ma być przetwarzany [IEEE 610]

**współczynnik awarii:** Stosunek liczby awarii danej kategorii do określonej jednostki miary, np. awarie na jednostkę czasu, awarie na liczbę transakcji, awarie na liczbę uruchomień komputera. [IEEE 610]

**współczynnik wydajności testu:** Metryka wysokiego poziomu mierząca efektywność i/lub wydajność używana do pomiaru i kontrolowania postępu testów. Przykładem może być np. Odsetek Wykrytych Błędów (OWB)

**condition:** A logical expression that can be evaluated as True or False, e.g.  $A > B$ . See also *test condition*.

**severity:** The degree of impact that a defect has on the development or operation of a component or system. [After IEEE 610]

**test input:** The data received from an external source by the test object during test execution. The external source can be hardware, software or human.

**input:** A variable (whether stored within a component or outside) that is read by a component.

**verification:** Confirmation by examination and through provision of objective evidence that specified requirements have been fulfilled. [ISO 9000]

**Wide Band Delphi:** An expert based test estimation technique that aims at making an accurate estimation using the collective wisdom of the team members.

**quality characteristic:** See *quality attribute*.

**performance indicator:** A high level metric of effectiveness and/or efficiency used to guide and control progressive development, e.g. lead-time slip for software development. [CMMI]

**pointer:** A data item that specifies the location of another data item; for example a data item that specifies the address of the next employee record to be processed [IEEE610]

**failure rate:** The ratio of the number of failures of a given category to a given unit of measure, e.g. failures per unit of time, failures per number of transactions, failures per number of computer runs. [IEEE 610]

**test performance indicator:** A high level metric of effectiveness and/or efficiency used to guide and control progressive test development, e.g. Defect Detection Percentage (DDP).

**wyciek pamięci:** Błąd mechanizmu dynamicznej alokacji pamięci dla programu polegający na nie odzyskiwaniu pamięci po zaprzestaniu jej używania. Może on prowadzić do awarii spowodowanej brakiem pamięci

**wydajność:** Stopień, w jaki system lub moduł, przy zadanych ograniczeniach, spełnia zamierzoną funkcjonalność odnośnie szybkości przetwarzania i przepustowości. [IEEE 610] Patrz także *efektywność*

**wyjście:** Zmienna (przechowywana wewnątrz modułu lub poza nim), której wartość ten moduł zapisuje.

**wykonalna ścieżka:** Ścieżka, dla której istnieje zestaw danych wejściowych i warunków wstępnych, przy których przejście tej ścieżki jest możliwe.

**wykonanie testów:** Proces projektowania i nadawania priorytetów procedurom testowym, tworzenie danych testowych i, opcjonalnie, przygotowywania jarzma testowego, pisanie automatycznych skryptów testowych; infrastruktury testowej: organizacyjnych artefaktów potrzebnych do wykonania testów składających się ze środowisk testowych narzędzi testowych, wyposażenia biurowego i procedur postępowania; danych testowych: danych pozyskiwanych z zewnętrznych źródeł przez obiekty testowe podczas wykonywania testów. Zewnętrznym źródłem może być sprzęt, oprogramowanie lub człowiek.

**wykonanie testu:** Proces przeprowadzenia testu na testowanym module lub systemie, produkujący rzeczywiste rezultaty.

**wymaganie funkcjonalne:** Wymaganie specyfikujące funkcję, którą moduł lub system musi realizować. [IEEE 610]

**wymaganie niefunkcjonalne:** Wymaganie, które nie dotyczy funkcjonalności, ale cech oprogramowania takich jak niezawodność, efektywność, użyteczność, pielęgnowalność i przenaszalność.

**wymaganie testowe:** Patrz *warunek testowy*

**wymaganie:** Warunek lub umiejętność potrzebna użytkownikowi do rozwiązania problemu lub osiągnięcia celu, które moduł lub system musi spełniać lub posiadać, aby wypełnić założenia umowy, standardu,

**memory leak:** A defect in a program's dynamic store allocation logic that causes it to fail to reclaim memory after it has finished using it, eventually causing the program to fail due to lack of memory.

**performance:** The degree to which a system or component accomplishes its designated functions within given constraints regarding processing time and throughput rate. [After IEEE 610] See also *efficiency*.

**output:** A variable (whether stored within a component or outside) that is written by a component.

**feasible path:** A path for which a set of input values and preconditions exists which causes it to be executed.

**test implementation:** The process of developing and prioritizing test procedures, creating test data and, optionally, preparing test harnesses and writing automated test scripts, test infrastructure: the organizational artifacts needed to perform testing, consisting of test environments, test tools, office environment and procedures; test input: the data received from an external source by the test object during test execution. The external source can be hardware, software or human.

**test execution:** The process of running a test on the component or system under test, producing actual result(s).

**functional requirement:** A requirement that specifies a function that a component or system must perform. [IEEE 610]

**non-functional requirement:** A requirement that does not relate to functionality, but to attributes such as reliability, efficiency, usability, maintainability and portability.

**test requirement:** See *test condition*.

**requirement:** A condition or capability needed by a user to solve a problem or achieve an objective that must be met or possessed by a system or system component to satisfy a contract, standard, specification,

specyfikacji lub innego formalnego dokumentu. [IEEE 610] or other formally imposed document. [After IEEE 610]

**wynik decyzji:** Rezultat decyzji (określający wybraną ścieżkę przetwarzania) **decision outcome:** The result of a decision (which therefore determines the branches to be taken).

**wynik fałszywie – negatywny:** patrz *test fałszywie obłany* **false-negative result:** See *false-fail result*

**wynik fałszywie - pozytywny:** patrz *test fałszywie zdany* **false-positive result:** See *false-pass result*

**wynik testu:** Patrz *rezultat*

**test outcome:** See *result*.

**wynik:** Patrz *rezultat*

**outcome:** See *result*.

**wyroczenia testowa:** Źródło dostarczające oczekiwanych rezultatów umożliwiające porównanie ich z otrzymanymi rezultatami. Wyroczenia może być istniejący system (np. dla benchmarków), podręcznik użytkownika, wiedza testera, ale nie może być nią kod. [Arion] **test oracle:** A source to determine expected results to compare with the actual result of the software under test. An oracle may be the existing system (for a benchmark), a user manual, or an individual's specialized knowledge, but should not be the code. [After Adrion]

**wyroczenia:** Patrz *wyroczenia testowa*

**oracle:** See *test oracle*.

**wyspecyfikowane wejście:** Dana wejściowa, dla której specyfikacja określa rezultat. **specified input:** An input for which the specification predicts a result.

**wytwarzanie sterowane testami:** Sposób wytwarzania oprogramowania, w którym przypadki testowe są przygotowywane i często automatyzowane zanim powstanie oprogramowanie, które będzie testowane za ich pomocą. **test driven development:** A way of developing software where the test cases are developed, and often automated, before the software is developed to run those test cases.

## Z

**zabezpieczenie:** Atrybuty oprogramowania określające jego zdolność do zapobiegania nieautoryzowanym przypadkowym i umyślnym dostępowi zarówno do programu, jak i do danych. [ISO 9126] Patrz także *funkcjonalność* **security:** Attributes of software products that bear on its ability to prevent unauthorized access, whether accidental or deliberate, to programs and data. [ISO 9126] See also *functionality*.

**zablokowany przypadek testowy:** Przypadek testowy, który nie może zostać wykonany, ponieważ jego warunki wstępne nie mogą zostać osiągnięte. **blocked test case:** A test case that cannot be executed because the preconditions for its execution are not fulfilled.

**zabłędnianie:** Patrz *posiew usterek*. [Abbott] **bebugging:** See *fault seeding*. [Abbott]

**zachowanie:** Odpowiedź modułu lub systemu na zestaw wartości wejściowych i warunków wstępnych. **behavior:** The response of a component or system to a set of input values and preconditions.

**zaliczenie testu:** Patrz *zaliczenie*

**test pass:** See *pass*.

**zaliczenie:** Test jest uważany za zaliczony, jeśli jego rezultat pasuje do rezultatu **pass:** A test is deemed to pass if its actual result matches its expected result.

oczekiwanego.

**zamknięcie testu:** Podczas fazy zamknięcia testów zbierane są dane z zakończonych aktywności w celu konsolidacji doświadczenia, testaliów, faktów i liczb. Faza zamknięcia testu składa się z finalizowania i archiwizacji testaliów i oceny procesu testowego, włączając przygotowanie raportu oceny testu. Patrz również *proces testowy*

**zamówione oprogramowanie:** Patrz *oprogramowanie na zamówienie*

**zamrożona podstawa testu:** Dokument podstawy testu, który może być zmieniony jedynie przez formalny proces kontroli zmiany. Patrz też: *podstawa*

**zapewnienie jakości:** Część zarządzania jakością zorientowane na zapewnienie że wymagania jakościowe będą spełnione [ISO-9000]

**zapis testów:** Patrz *log testowy*

**zapisywanie testów:** Patrz *logowanie testów*

**zarządzanie błędami:** proces składający się z rozpoznania, analizy, prowadzenia działań i likwidacji usterek. Polega on na rejestracji usterek, ich klasyfikacji oraz określaniu ich działania [IEEE 1044]

**zarządzanie incydentami:** Proces rozpoznawania, badania i dysponowania incydentami oraz podejmowania działań w związku z nimi. Proces ten obejmuje rejestrację incydentów, ich klasyfikację oraz ocenę wpływu. [IEEE 1044]

**zarządzanie jakością:** Ogół skoordynowanych czynności mających na celu kierowanie organizacją i kontrolowanie jej pod kątem jakości. Zwykle obejmuje czynności takie jak: zdefiniowanie polityki jakościowej i celów jakościowych, planowanie jakości, sterowanie jakością, zapewnienie jakości i poprawa jakości.

**zarządzanie konfiguracją:** Dyscyplina używająca technicznych i administracyjnych metod kierowania i nadzoru aby: określić i udokumentować charakterystyki funkcjonalne i fizyczne elementów konfiguracji, kontrolować zmiany tych charakterystyk, zapisywać i raportować o wykonywaniu zmian i statusie

**test closure:** During the test closure phase of a test process data is collected from completed activities to consolidate experience, testware, facts and numbers. The test closure phase consists of finalizing and archiving the testware and evaluating the test process, including preparation of a test evaluation report. See also *test process*.

**custom software:** See *bespoke software*

**frozen test basis:** A test basis document that can only be amended by a formal change control process. See also *baseline*.

**quality assurance:** Part of quality management focused on providing confidence that quality requirement will be fulfilled [ISO 9000]

**test record:** See *test log*.

**test recording:** See *test logging*.

**defect management:** The process of recognizing, investigating, taking action and disposing of defects. It involves recording defects, classifying them and identifying the impact. [After IEEE 1044]

**incident management:** The process of recognizing, investigating, taking action and disposing of incidents. It involves logging incidents, classifying them and identifying the impact. [After IEEE 1044]

**quality management:** Coordinated activities to direct and control an organization with regard to quality. Direction and control with regard to quality generally includes the establishment of the quality policy and quality objectives, quality planning, quality control, quality assurance and quality improvement. [ISO 9000]

**configuration management:** A discipline applying technical and administrative direction and surveillance to: identify and document the functional and physical characteristics of a configuration item, control changes to those characteristics, record and report change processing and implementation status, and verify

implementacji oraz weryfikować zgodność z compliance with specified requirements. wyspecyfikowanymi wymaganiami. [IEEE [IEEE 610] 610]

**zarządzanie problemami:** Patrz *zarządzanie incydentami* **problem management:** See *defect management*.

**zarządzanie ryzykiem:** Systematyczne wdrażanie procedur i praktyk dla zadań identyfikacji, analizowania, ustalania priorytetów i kontrolowania ryzyka. **risk management:** Systematic application of procedures and practices to the tasks of identifying, analyzing, prioritizing, and controlling risk.

**zarządzanie testami:** Planowanie, szacowanie, monitorowanie oraz kontrola przebiegu testów, na ogół prowadzone przez kierownika testów. **test management:** The planning, estimating, monitoring and control of test activities, typically carried out by a test manager.

**zastępowalność:** Zdolność oprogramowania do wykorzystania w miejsce innego oprogramowania z takim samym przeznaczeniem i w takim samym środowisku. [ISO 9126] Patrz także *przenaszalność* **replaceability:** The capability of the software product to be used in place of another specified software product for the same purpose in the same environment. [ISO 9126] See also *portability*.

**zaśleпка:** Szkieletowa albo specjalna implementacja modułu używana podczas produkcji lub testów innego modułu, który tę zaślepkę wywołuje albo jest w inny sposób od niej zależny. Zastępuje wywoływany moduł. [IEEE 610] **stub:** A skeletal or special-purpose implementation of a software component, used to develop or test a component that calls or is otherwise dependent on it. It replaces a called component. [After IEEE 610]

**zbiór testów:** Patrz *zestaw testowy* **test set:** See *test suite*.

**zdarzenie testowe:** Patrz *incydent* **test incident:** See *incident*.

**zdolność adaptacyjna:** Zdolność oprogramowania do dostosowania się do różnych środowisk, bez konieczności stosowania działań lub środków innych niż te, które dostarczono do tego celu [ISO 9126]. Patrz także *przenaszalność* **adaptability:** the capability of the software product to be adapted for different specified environments without applying actions or means other than those provided for this purpose for the software considered [ISO 9126]. See also *portability*

**zdolność analizy:** Zdolność wytwarzanego produktu do bycia zdiagnozowanym pod kątem braków lub przyczyn awarii lub pod kątem rozpoznania części do modyfikacji. [ISO 9126] Patrz też *pielęgnowalność* **analyzability:** The capability of the software product to be diagnosed for deficiencies or causes of failures in the software, or for the parts to be modified to be identified [ISO 9126]. See also *maintainability*

**zestaw przypadków testowych:** Patrz *zestaw testowy* **test case suite:** See *test suite*.

**zestaw testowy:** Ciąg przypadków testowych, w którym warunków wyjściowych z jednego testu używa się jako warunki wejściowe do następnego testu. **test suite:** A set of several test cases for a component or system under test, where the post condition of one test is often used as the precondition for the next one.

**zgadywanie błędów:** Technika projektowania testów gdzie bazując na doświadczeniu testera przewiduje się, jakie defekty, będące efektem wykonanych pomyłek, mogą być obecne w testowanym module lub systemie i projektuje się testy tak, aby te defekty ujawnić.

**zgłoszenie błędu:** Patrz: *zgłoszenie defektu*

**zgłoszenie defektu:** Dokument opisujący usterkę w module lub systemie, która może spowodować nieprawidłowe działanie jego wymaganych funkcji [IEEE 829]

**zgłoszenie incydentu testowego:** Patrz: *zgłoszenie defektu*

**zgłoszenie incydentu:** Patrz *zgłoszenie defektu*

**zgłoszenie odchylenia:** Patrz *zgłoszenie defektu*

**zgłoszenie problemu:** Patrz *zgłoszenie defektu*

**zgodność:** Zdolność oprogramowania, do podlegania standardom, konwencjom albo regulacjom prawnym i podobnym rozporządzeniom [ISO 9126]

**zintegrowany model dojrzałości organizacyjnej (CMMI):** Struktura, która opisuje kluczowe elementy efektywnego rozwoju produktu i procesu jego utrzymania. Model dojrzałości organizacyjnej składa się z najlepszych praktyk w planowaniu, inżynierii i zarządzaniu rozwojem i utrzymaniem produktu. CMMI jest wyznaczonym następcą CMM. [CMMI] Patrz też: *model dojrzałości organizacyjnej (CMM)*

**Zintegrowany Model Dojrzałości Testów (TMMi - akronim od angielskiego Test Maturity Model Integrated):** Pięciostopniowa podstawa działań na rzecz ulepszenia procesu testowego, zgodna ze Zintegrowanym Modelem Dojrzałości Organizacyjnej (Capability Maturity Model Integrated CMMI), który opisuje kluczowe elementy efektywnego procesu testowego.

**złożoność cyklomatyczna:** Liczba niezależnych ścieżek w programie. Złożoność cyklomatyczna jest określona wzorem:  $L - N + 2P$ , gdzie L to liczba krawędzi/połączeń w grafie, N to liczba

**error guessing:** A test design technique where the experience of the tester is used to anticipate what defects might be present in the component or system under test as a result of errors made, and to design tests specifically to expose them.

**bug report:** See *defect report*.

**defect report:** A document reporting on any flaw in a component or system that can cause the component or system to fail to perform its required function. [After IEEE 829]

**test incident report:** See *incident report*.

**software test incident report:** See *incident report*.

**deviation report:** See *incident report*.

**problem report:** See *defect report*.

**compliance:** The capability of the software product to adhere to standards, conventions or regulations in laws and similar prescriptions. [ISO 9126]

**Capability Maturity Model Integration (CMMI):** A framework that describes the key elements of an effective product development and maintenance process. The Capability Maturity Model Integration covers best-practices for planning, engineering and managing product development and maintenance. CMMI is the designated successor of the CMM. [CMMI] See also *Capability Maturity Model (CMM)*.

**Test Maturity Model Integrated (TMMi):** A five level staged framework for test process improvement, related to the Capability Maturity Model Integration (CMMI) that describes the key elements of an effective test process.

**cyclomatic complexity:** The number of independent paths through a program. Cyclomatic complexity is defined as:  $L - N + 2P$ , where - L = the number of edges/links in a graph, - N = the number of nodes in a

węzłów grafu, P to liczba rozłącznych części grafu (np.. wywoływany graf lub podprocedura) [McCabe]

**złożoność:** Stopień trudności do nauczenia, utrzymania i weryfikacji. według którego moduł lub system i/lub jego wewnętrzna struktura został (została) zaprojektowany(a). Patrz *złożoność cyklomatyczna*

**zmiana stanu:** Przejście pomiędzy dwoma stanami systemu lub modułu.

**zmienna:** Element pamięci komputera, który jest dostępny w programie poprzez swoją nazwę.

**zmiennosc:** Zdolność produktu oprogramowania do wprowadzania wyspecyfikowanych zmian. [ISO 9126]. Patrz: *pielęgnowalność*

**zrozumiałość:** Zdolność oprogramowania do umożliwienia użytkownikowi zrozumienia czy jest ono odpowiednie i jak może być użyte do realizacji określonych zadań. Patrz także *użyteczność*

**zużycie zasobów:** Zdolność oprogramowania do wykorzystania odpowiedniej ilości i typu zasobów np. ilości pamięci głównej i dodatkowej wykorzystywanej przez program oraz rozmiaru plików tymczasowych podczas działania oprogramowania w ustalonych warunkach. [za ISO 9126] Patrz także *efektywność*

graph - P = the number of disconnected parts of the graph (e.g. a called graph and a subroutine) [After McCabe]

**complexity:** The degree to which a component or system has a design and/or internal structure that is difficult to understand, maintain and verify. See also *cyclomatic complexity*.

**state transition:** A transition between two states of a component or system.

**variable:** An element of storage in a computer that is accessible by a software program by referring to it by a name.

**changeability:** The capability of the software product to enable specified modifications to be implemented. [ISO 9126] See also *maintainability*.

**understandability:** The capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use. [ISO 9126] See also *usability*.

**resource utilization:** The capability of the software product to use appropriate amounts and types of resources, for example the amounts of main and secondary memory used by the program and the sizes of required temporary or overflow files, when the software performs its function under stated conditions. [After ISO 9126] See also *efficiency*.

#### Odwołania do norm

1. BS 7925-2:1998. Software Component Testing.
2. DO-178B:1992. Software Considerations in Airborne Systems and Equipment Certification, Requirements and Technical Concepts for Aviation (RTCA SC167).
3. IEEE 610.12:1990. Standard Glossary of Software Engineering Terminology.
4. IEEE 829:1998. Standard for Software Test Documentation.
5. IEEE 1008:1993. Standard for Software Unit Testing.
6. IEEE 1012:2004. Standard for Verification and Validation Plans
7. IEEE 1028:1997. Standard for Software Reviews and Audits.
8. IEEE 1044:1993. Standard Classification for Software Anomalies.
9. IEEE 1219:1998. Software Maintenance.
10. ISO/IEC 2382-1:1993. Data processing - Vocabulary - Part 1: Fundamental terms.
11. ISO 9000:2005. Quality Management Systems – Fundamentals and Vocabulary.
12. ISO/IEC 9126-1:2001. Software Engineering – Software Product Quality – Part 1:Quality characteristics and sub-characteristics.
13. ISO/IEC 12207:1995. Information Technology – Software Life Cycle Processes.

14. ISO/IEC 14598-1:1999. Information Technology – Software Product Evaluation – Part 1

Bibliografia:

- [*Abbott*] J. Abbot (1986), *Software Testing Techniques*, NCC Publications.
- [*Adrion*] W. Adrion, M. Branstad and J. Cherniabsky (1982), *Validation, Verification and Testing of Computer Software*, in: *Computing Surveys*, Vol. 14, No 2, June 1982.
- [*Bach*] J. Bach (2004), *Exploratory Testing*, in: E. van Veenendaal, *The Testing Practitioner – 2nd edition*, UTN Publishing, ISBN 90-72194-65-9.
- [*Beizer*] B. Beizer (1990), *Software Testing Techniques*, van Nostrand Reinhold, ISBN 0-442-20672-0
- [*Chow*] T. Chow (1978), *Testing Software Design Modelled by Finite-State Machines*, in: *IEEE Transactions on Software Engineering*, Vol. 4, No 3, May 1978.
- [*CMM*] M. Paulk, C. Weber, B. Curtis and M.B. Chrissis (1995), *The Capability Maturity Model, Guidelines for Improving the Software Process*, Addison-Wesley, ISBN 0-201-54664-7
- [*CMMI*] M.B. Chrissis, M. Konrad and S. Shrum (2004), *CMMI, Guidelines for Process Integration and Product Improvement*, Addison Wesley, ISBN 0-321-15496-7
- [*Fenton*] N. Fenton (1991), *Software Metrics: a Rigorous Approach*, Chapman & Hall, ISBN 0-3249-425-1
- [*Fewster and Graham*] M. Fewster and D. Graham (1999), *Software Test Automation, Effective use of test execution tools*, Addison-Wesley, ISBN 0-201-33140-3.
- [*Freedman and Weinberg*] D. Freedman and G. Weinberg (1990), *Walkthroughs, Inspections, and Technical Reviews*, Dorset House Publishing, ISBN 0-932633-19-6.
- [*Gerrard*] P. Gerrard and N. Thompson (2002), *Risk-Based E-Business Testing*, Artech House Publishers, ISBN 1-58053-314-0.
- [*Gilb and Graham*] T. Gilb and D. Graham (1993), *Software Inspection*, Addison-Wesley, ISBN 0-201-63181-4.
- [*Graham*] D. Graham, E. van Veenendaal, I. Evans and R. Black (2007), *Foundations of Software Testing*, Thomson Learning, ISBN 978-1-84480-355-2
- [*Grochtmann*] M. Grochtmann (1994), *Test Case Design Using Classification Trees*, in: *Conference Proceedings STAR 1994*.
- [*Hetzel*] W. Hetzel (1988), *The complete guide to software testing – 2nd edition*, QED Information Sciences, ISBN 0-89435-242-3.
- [*McCabe*] T. McCabe (1976), *A complexity measure*, in: *IEEE Transactions on Software Engineering*, Vol. 2, pp. 308-320
- [*Musa*] J. Musa (1998), *Software Reliability Engineering Testing*, McGraw-Hill Education, ISBN 0-07913-271-5
- [*Myers*] G. Myers (1979), *The Art of Software Testing*, Wiley, ISBN 0-471-04328-1.
- [*TMap*] M. Pol, R. Teunissen, E. van Veenendaal (2002), *Software Testing, A guide to the TMap Approach*, Addison Wesley, ISBN 0-201-745712.
- [*Veenendaal*] E. van Veenendaal (2004), *The Testing Practitioner – 2nd edition*, UTN Publishing, ISBN 90-72194-65-9.