



Processeurs de langage de programmation — Méthodes d'essai — Lignes directrices pour leur élaboration et acceptabilité

Programming language processors — Test methods — Guidelines for their development and acceptability

L'ISO (Organisation internationale de normalisation) est une fédération mondiale d'organismes nationaux de normalisation (comités membres de l'ISO). L'élaboration des Normes internationales est normalement confiée aux comités techniques de l'ISO. Chaque comité membre intéressé par une étude a le droit de faire partie du comité technique créé à cet effet. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec l'ISO participent également aux travaux.

La tâche principale des comités techniques de l'ISO est d'élaborer les Normes internationales. Exceptionnellement, un comité technique peut proposer la publication d'un rapport technique de l'un des types suivants:

- type 1: lorsque, en dépit de maints efforts au sein d'un comité technique, l'accord requis ne peut être réalisé en faveur de la publication d'une Norme internationale;
- type 2: lorsque le sujet en question est encore en cours de développement technique et requiert une plus grande expérience;
- type 3: lorsqu'un comité technique a réuni des données de nature différente de celles qui sont normalement publiées comme Normes internationales (ceci pouvant comprendre des informations sur l'état de la technique, par exemple).

La publication des rapports techniques dépend directement de l'acceptation du Conseil de l'ISO. Les rapports techniques des types 1 et 2 font l'objet d'un nouvel examen trois ans au plus tard après leur publication afin de décider éventuellement de leur transformation en Normes internationales. Les rapports techniques du type 3 ne doivent pas nécessairement être révisés avant que les données fournies ne soient plus jugées valables ou utiles.

L'ISO/TR 9547 a été préparé par le comité technique ISO/TC 97, *Systèmes de traitement de l'information*.

Le présent rapport technique du type 3 a été élaboré afin de fournir des lignes directrices aux concepteurs des méthodes d'essai pour les langages de programmation et afin d'aider également les concepteurs des langages de programmation proprement dits. Il devrait faciliter la préparation de méthodes d'essai spécifiques qui aideront les réalisateurs à fabriquer des processeurs de langage conformes à la norme, dont bénéficiera l'utilisateur final.

Table des matières

0	Introduction	3
1	Objet et domaine d'application	3
2	Définitions	3
2.1	Configuration	3
2.2	Extension	4
2.3	Défini par la réalisation	4
2.4	Processeur	4
2.5	Essai	4
2.6	Méthode d'essai	4
2.7	Programme de test	4
2.8	Rapport de test	4
2.9	Jeu de test	4
2.10	Outils de test	4
2.11	Documents requis	4
2.12	Sous-ensemble	4
3	Objet soumis au test	4
4	Description des méthodes d'essai	5
4.1	Jeu de test	5
4.2	Outils de test	6
4.3	Procédures techniques	6
4.3.1	Plan global de test	6
4.3.2	Documents requis	6
5	Lignes directrices pour l'élaboration des méthodes d'essai	7
5.1	Examen des méthodes d'essai existantes	7
5.2	Critères de conception des jeux de test	7
5.3	Faisabilité de l'essai	8
5.4	Caractéristiques du langage non couverts par le jeu de test	8
5.5	Test des sous-ensembles	9
5.6	Exigences de portabilité	9
5.7	Outils de test	9
6	Maintenance et révision des jeux de test	9
7	Disponibilité	9
8	Acceptabilité des méthodes d'essai	9
8.1	Critères d'acceptation de la méthode d'essai	9
8.2	Révision de la méthode d'essai	10

ITEH STANDARD PREVIEW
(standards.iteh.ai)

ISO/TR 9547:1988

<https://standards.iteh.ai/catalog/standards/sist/5454ead4-f2e3-4dde-89b7-17311239f667/iso-tr-9547-1988>

0 Introduction

Le domaine des langages de programmation est un sujet en évolution constante : on met au point de nouveaux langages, on révisé les anciens. Dans les deux cas, il ne faut pas oublier les exigences de conformité ; aussi faut-il étudier le problème des tests au moment de l'élaboration et de la révision des normes de langages.

L'objectif du présent document est de donner des lignes directrices pour l'élaboration d'une méthode d'essai basée sur un jeu de programmes de test et d'étapes précises devant être suivies pour qu'une méthode d'essai soit formellement approuvée par l'ISO. Il convient de lire ces lignes directrices conjointement à la norme d'un langage donné afin d'élaborer une méthode spécifique pour des processeurs de langages de programmation.

Ces lignes directrices sont écrites à l'usage des concepteurs de méthodes d'essai et de ceux qui les approuveront au sein de l'ISO. La disponibilité de méthodes d'essai spécifiques aidera les réalisateurs à fabriquer des processeurs de langage conformes à la norme, ce qui profitera de ce fait à l'utilisateur final.

En lisant ces lignes directrices, il ne faut surtout pas oublier que les normes des langages de programmation n'ont pas encore atteint un niveau de précision et de complétude permettant l'élaboration de tests de conformité pour chacun des aspects et caractéristiques d'un langage décrit dans une norme de langage de programmation.

Les questions relatives aux essais sont rarement l'objectif principal des comités de normalisation, ce qui peut poser des problèmes pour l'application rigoureuse de ces lignes directrices à une quelconque norme donnée de langage de programmation.

1 Objet et domaine d'application

Les présentes lignes directrices décrivent une méthodologie permettant de déterminer si un processeur conçu pour un langage de programmation particulier possède les caractéristiques requises telles qu'elles sont indiquées dans la Norme internationale de ce langage de programmation.

L'évaluation de la conformité d'un processeur de langage peut être faite au moyen de la "Méthodologie de Jeu de test". On n'exclut pas l'utilisation d'autres méthodologies, mais celles-ci ne sont pas traitées dans le présent document.

2 Définitions

Pour les besoins des présentes lignes directrices, les termes suivants s'appliquent :

2.1 Configuration : Ordinateurs hôte et cible, avec tout (ou tous) système(s) d'exploitation et logiciels utilisés pour faire fonctionner un processeur.

2.2 Extension : Possibilité du langage qui ne figure pas dans la norme du langage, mais qui n'engendre ni d'ambiguïté ni de contradiction lorsqu'elle est ajoutée à la norme du langage (bien que, dans quelques langages, elle puisse servir à lever une restriction).

2.3 Défini par la réalisation : Dépendant du processeur, mais devant selon la norme du langage, être défini et documenté par les réalisateurs.

2.4 Processeur : Compilateur, traducteur ou interprète travaillant en association avec une configuration.

2.5 Essai : Opération technique qui consiste à déterminer une ou plusieurs caractéristiques d'un produit, processus ou service donné, selon un mode opératoire spécifié. (Guide ISO/CEI N° 2 - 1986).

2.6 Méthode d'essai : Procédure technique définie pour la réalisation d'un essai. (Guide ISO/CEI N° 2 - 1986).

2.7 Programme de test : Suite de caractères destinée à être soumise à un processeur afin de déterminer si oui ou non ce processeur présente un cas spécifique d'une propriété déterminée.

2.8 Rapport d'essai : Document qui présente les résultats d'un essai et d'autres informations s'y rapportant. (Guide ISO/CEI N° 2 - 1986).

2.9 Jeu de test : Un ensemble de programmes de tests de référence destiné à évaluer la conformité d'un processeur à la norme d'un langage.

2.10 Outils de test : Tout moyen supplémentaire permettant d'améliorer l'efficacité, la fiabilité et la facilité d'emploi des différents stades des essais (par exemple, mise en oeuvre du jeu de test, garantie de l'intégrité, du traitement du jeu de test, collecte des résultats des essais, analyse des résultats des essais, édition du rapport d'essai).

2.11 Documents requis : Ensemble de documents exigés par la norme du langage de programmation.

2.12 Sous-ensemble : Un sous-ensemble S du langage de programmation L est un langage de programmation tel que tout programme en S

- est également un programme de L, et
- a le même sens en S qu'en L.

3 Objet soumis au test

Etant donné qu'un processeur ne fonctionne qu'en association avec une configuration, quelques conditions doivent être respectées :

3.1 Spécification intégrale du processeur soumis au test, y compris les options sélectionnées, et de la configuration adoptée pour l'essai.

3.2 Test d'une seule et même copie d'un processeur sur une configuration déterminée.

4 Description des méthodes d'essai

Dans le présent contexte, la procédure technique utilise un jeu de test, des outils de test et, éventuellement des données.

4.1 Jeu de test

Un jeu de test doit être conçu pour contrôler la conformité des processeurs en les soumettant aux programmes de test. Chaque essai porte sur quelques lignes directrices de la norme du langage et leur interaction. Un jeu de test doit être conçu pour tester la conformité et non pour évaluer d'autres caractéristiques d'un processeur.

Le jeu de test doit, si possible, couvrir tous les aspects de la norme du langage et examiner les problèmes de mise en oeuvre touchant à la conformité.

Le jeu de test ne doit pas être trop volumineux ; on doit tenir compte des aspects économiques (par exemple, coûts relatifs du contrôle de la conformité par rapport aux coûts de développement d'une mise en oeuvre).

Le jeu de test doit être écrit de manière à pouvoir être révisé facilement sous un système de gestion de version et soumis à une procédure de révision.

Le jeu de test doit être conçu de façon à prendre en compte tous les niveaux ou options possibles spécifiés dans la norme du langage. Ainsi, le jeu de test doit-il être modulaire.

Les programmes de test doivent être, dans la mesure du possible, indépendants les uns des autres et l'ordre d'exécution des programmes de test ne doit pas influencer sur les résultats des uns ou des autres. Là où cela n'est pas possible ou pas souhaitable, en raison de certains aspects dus à la conception du langage, il convient de spécifier et documenter les relations entre tous les tests (ou les modules du code source) qui dépendent les uns des autres.

Chaque programme de test doit avoir un seul objectif en rapport avec les exigences de la norme. Lorsque cela n'est pas pratique, il faut pouvoir identifier facilement les résultats des objectifs particuliers. Les programmes de test doivent fournir un résultat conformément aux objectifs stipulés.

Le jeu de test doit comprendre des programmes de test conformes aux règles de la norme du langage. Il doit également comprendre des programmes de test non conformes aux règles de ladite norme, dans les situations où la norme du langage spécifie les propriétés syntaxiques ou sémantiques que l'on doit rejeter. Les programmes de test doivent pouvoir, dans la mesure du possible, opérer un auto-contrôle et de ce fait doivent signaler le succès ou l'échec par un message (ce qui devrait faciliter l'édition automatique d'un rapport d'essai).

Le jeu de test doit permettre l'utilisation de valeurs paramétrées qui prennent en compte la configuration utilisée avec le processeur.

4.2 Outils de test

On doit prendre les dispositions nécessaires pour mettre au point des outils de test. L'automatisation à la fois du contrôle et de l'analyse des résultats est essentielle, afin de faciliter des essais rigoureux et économiques.

4.3 Procédures techniques

4.3.1 Plan global de test

La méthode d'essai doit définir le plan global de test tel que décrit ci-après.

Un jeu de test doit être exécuté sur une configuration désignée en utilisant le processeur à tester. Les résultats (succès ou échec) de l'essai d'un processeur spécifique sur une configuration spécifique (tous les composants étant entièrement définis) seront reproductibles si aucune modification n'a eu lieu sur aucun composant de la configuration.

iTeh STANDARD PREVIEW
(standardsiteh.ai)

Les renseignements donnés dans les documents requis doivent être examinés pour voir s'ils sont conformes à la norme du langage et doivent être comparés aux résultats des essais.

ISO/TR 9547:1988

La sortie d'un processeur qui a été contrôlé avec le jeu de test doit être analysée suivant des règles clairement définies. Ces règles doivent donner des critères d'évaluation objective de toutes les sorties possibles, y compris l'absence de sortie, pour chaque programme de test faisant partie du jeu de test.

Les résultats du contrôle de conformité par rapport à la norme doivent être analysés et consignés dans un rapport. Celui-ci doit fournir et résumer tous les renseignements relatifs au contrôle de conformité par rapport à la norme du langage (mise en place du contrôle, contrôle effectif, événements majeurs au cours du contrôle). Le rapport d'essai doit également énumérer les caractéristiques du langage impossibles à tester, en donnant une raison, là où il convient, pour la réalisation soumise au test. L'analyse et l'édition du rapport d'essai doivent être aussi automatisées que possible. Le niveau d'automatisation doit dépendre du type de test et du format de présentation des résultats.

4.3.2 Documents connexes

La méthode d'essai doit spécifier la documentation nécessaire, comprenant :

- la description de la classification du test avec les objectifs de chaque classe ;
- l'ordre d'exécution des tests, s'il est demandé ;
- une liste des caractéristiques du langage non couvertes par le jeu de test.

5 Lignes directrices pour l'élaboration des méthodes d'essai

Lors de la mise au point d'une méthode d'essai, on doit prendre en compte un certain nombre de points. Ces points figurent ci-après.

5.1 Examen des méthodes d'essai existantes

Les méthodes d'essai utilisées pour d'autres processeurs doivent être examinées pour déterminer avec quel degré de succès elles ont répondu à leurs objectifs de conception. Lorsque les objectifs de conception d'une nouvelle méthode d'essai ressemble à ceux de méthodes d'essai existantes, il convient d'envisager la possibilité d'adapter les principes de conception correspondants à la nouvelle méthode d'essai.

5.2 Critères de conception des jeux de test

Chaque programme d'essai doit être écrit de manière à faciliter sa compréhension par l'utilisation d'une documentation claire et explicite, comprenant :

- des commentaires clairs, concis, et non redondant d'un programme de test à l'autre ;
- une référence à des clauses de la norme du langage ;
- un énoncé clair des hypothèses faites dans la conception du jeu de test.

<https://standards.iteh.ai/catalog/standards/sist/5454ead4-f2e3-4dde-89b7-67311329fc67/iso-tr-9547-1988>

Le jeu de test doit être écrit de manière à faciliter sa compréhension par l'utilisation d'une codification explicite, telle que :

- conventions d'appellation pour les programmes de test ;
- identificateurs mnémoniques ;
- cohérence de la codification dans tout le jeu de test ;
- majuscules et minuscules.

L'écriture du jeu de test doit tenir compte des aspects techniques spécifiés dans la norme du langage et qui sont définis par la mise en oeuvre, par exemple :

- paramètres de mise en oeuvre (par exemple, utilisation du plus grand entier supporté) ;
- valeurs minimales et maximales des paramètres avec un choix raisonnable entre les deux (par exemple, mesure de la profondeur d'imbrication) ;
- utilisation de caractéristiques qui n'existent pas dans les sous-ensembles ;
- précision de la virgule flottante ;

- utilisation des fichiers et de données extérieures ;
- dispositifs d'entrée-sortie.

Le jeu de test doit être écrit de manière à réduire au minimum l'utilisation des fonctions qui pourraient être restreintes par des caractéristiques de la configuration ou de la mise en oeuvre. Lorsqu'on utilise de telles fonctions dans le jeu de test pour évaluer d'autres aspects de conformité, cette dépendance doit être contenue dans les limites raisonnables (par exemple, valeurs numériques, dimension des tableaux).

Le jeu de test doit être également conçu pour utiliser un ensemble minimal d'instructions simples pour tester les fonctions complexes. Les structures complexes ne doivent être utilisées qu'en cas de besoin spécifique pour un test particulier. Les programmes non conformes à la norme du langage ne doivent s'écarter de celle-ci que dans la mesure où cela est nécessaire pour atteindre l'objectif d'un test spécifique.

5.3 Faisabilité de l'essai

La méthode d'essai doit être conçue de façon à ne pas être trop difficile à mettre en oeuvre. La méthode d'essai doit être un outil utile pour les responsables de la mise en oeuvre ; par conséquent, le fait d'incorporer l'utilisation d'un jeu de test dans le contrôle d'un compilateur au cours de son développement et de sa maintenance ne doit pas être trop coûteux. <https://standards.iteh.ai/catalog/standards/sist/5454ead4-f2e3-4dde-89b7-67311329fc67/iso-tr-9547-1988>

Le volume de sortie au cours de l'exécution de la méthode d'essai doit être réduit au minimum.

5.4 Caractéristiques du langage non couvertes par le jeu de test

Il se peut qu'il soit impossible de tester certaines caractéristiques du langage.

D'autres éléments de la norme du langage peuvent avoir été exclus de l'objectif de la méthode d'essai pour des raisons pratiques ou pour d'autres raisons (par exemple, pour éviter d'avoir un jeu de test trop grand ou trop complexe).

Lorsque le jeu ne couvre pas certaines caractéristiques du langage, la méthode d'essai doit en faire mention (ou les énumérer).

Les responsables de la mise en oeuvre des jeux de test peuvent parfois rencontrer des problèmes particuliers relatifs aux extensions du langage. En général, lorsque la norme du langage permet les extensions, mais n'en donne pas les spécifications, le contrôle des extensions doit être considéré comme en dehors de l'objet de la méthode d'essai. Là où les extensions ne sont pas permises, il se peut qu'il soit particulièrement difficile de vérifier minutieusement leur absence. On doit apporter un soin particulier à décrire la stratégie de contrôle des extensions (s'il y en a effectivement une) dans la méthode d'essai.

5.5 Test des sous-ensembles

La méthode d'essai doit être conçue de manière à permettre de tester les sous-ensembles (lorsque la norme du langage les autorise) à l'aide du même programme de test ou d'un sous-ensemble de celui-ci.

5.6 Exigences de portabilité

Le jeu de test doit être portable sur différentes configurations. Les petites ne doivent pas être pénalisées. Ainsi, chaque programme doit-il pouvoir s'exécuter aussi bien sur une petite configuration que sur une grande. Le résultat du test (c'est-à-dire succès ou échec) ne doit pas dépendre de la taille du programme.

5.7 Outils de test

Les outils de test automatiques doivent, si possible, être écrits dans le même langage que celui du processeur soumis au test, et doivent être portables sur différentes configurations.

6 Maintenance et révision des jeux de test

Des dispositions doivent être prises concernant la maintenance et la révision des jeux de test à travers une enquête publique et selon un calendrier publié.

7 Disponibilité

Les jeux de test et les outils doivent être disponibles sous un ou plusieurs des formats généralement utilisés dans la profession. A cet effet, il convient de prendre en considération l'utilisation des normes ISO.

8 Acceptabilité des méthodes d'essai

Pour qu'une méthode d'essai donnée soit acceptée, il faut qu'elle respecte les critères suivants.

8.1 Critères d'acceptation de la méthode d'essai

Plusieurs critères existent :

- 1) Le langage en question doit faire l'objet d'une norme ISO.
- 2) L'objectif de l'essai et les programmes de test doivent se rapporter aux exigences de conformité de la norme.
- 3) Le jeu de test doit être complet au sens qu'il teste, dans la mesure du possible, tous les aspects du langage.
- 4) L'exécution des programmes de test doit se faire de façon pratique. L'exécution des jeux de test ne doit pas prendre trop de temps, ne doit pas utiliser des ressources excessives du système, et ne doit pas être trop coûteux.