

Glossaire – labo Interactif

A

Abstraction fonctionnelle

L'utilisation de fonctions pour construire de nouvelles fonctions. Une nouvelle fonction est définie avec d'autres fonctions.

Adjonction

Une fonction qui définit un nouvel environnement en ajoutant une paire (identificateur, variable) à un environnement existant.

Affectation compatible

Une tentative de seconde affectation à une variable à affectation unique, dont la valeur est compatible avec la valeur déjà en mémoire. Par exemple, `declare X Y in X=foo(Y) X=foo(1)`, l'affectation `X=foo(1)` est compatible et fera `Y=1`.

Affectation incompatible

Une tentative de seconde affectation à une variable à affectation unique, dont la valeur n'est pas compatible avec la valeur déjà en mémoire. Par exemple, `declare X=42 X=43`, l'affectation `X=43` est une affectation incompatible. Une affectation incompatible signale une erreur avec une exception.

Affectation multiple

Une cellule peut être affectée plusieurs fois.

Affectation unique

Une variable ne peut être affectée qu'à une seule valeur.

Analyseur de jetons

Un programme qui prend une séquence de caractères et renvoie une séquence de jetons.

Analyseur lexical

Voir Analyseur de jetons.

Atome

Une constante qui commence avec une lettre minuscule et qui est suivie de lettres (majuscules ou minuscules), de chiffres ou du caractère de soulignement ('_').

C

Calcul

Une séquence d'états d'exécution qui commence par un état initial.

Capacité objet

Une référence infalsifiable à un objet. Un système basé sur les capacités objet donne une sécurité en profondeur qui permet d'éviter la plupart des problèmes des virus et autres programmes malicieux.

Cellule

Une paire composée d'une constante, qui est un nom, et d'une référence dans la mémoire à affectation unique. Il existe une opération d'affectation qui permet de changer cette référence. La paire elle-même fait partie de la mémoire à affectation multiple. Les cellules implémentent un état explicite. On peut ajouter les cellules à un modèle de calcul pour donner un modèle avec état.

Classe

Une fonction qui renvoie un objet avec un nouvel état quand elle est appelée. Un langage de programmation peut donner un soutien syntaxique aux classes.

Combinaison

Une fonction de deux entiers n et k qui donne le nombre de sous-ensembles de taille k possibles d'un ensemble

Glossaire – labo Interactif

de taille n .

Complexité temporelle

Le temps d'exécution d'un programme en fonction de la taille de l'entrée, à un facteur constant près.

Concurrence

La propriété d'un programme qui contient plusieurs activités indépendantes, chacune s'exécutant à sa propre vitesse.

Cons

Voir Paire de liste.

D

Dataflow

Le comportement d'une exécution d'attendre les informations quand elles ne sont pas disponibles pendant l'exécution et de continuer quand les opérations deviennent disponibles. Par exemple, quand une opération essaie d'utiliser une variable qui n'est pas encore liée. Si l'opération attend qu'une autre opération lie la variable et continue ensuite, nous disons que l'exécution est dataflow.

Développement descendant

Une méthodologie de construction de programmes qui consiste d'abord à écrire le programme principal en ensuite de le compléter avec les fonctions manquantes. On l'appelle *top-down development* en anglais.

E

Entier

Un nombre qui peut être zéro, négatif ou positif et dont la partie fractionnelle est zéro. Un entier est un type dans le langage Oz.

Environnement

Une fonction qui prend un identificateur et qui renvoie une variable en mémoire. Cette fonction existe pour comprendre l'exécution d'un programme mais ne fait pas partie des fonctions définies par le programmeur.

État

Une séquence de valeurs dans le temps qui contient les résultats intermédiaires d'un calcul souhaité.

État d'exécution

Une paire qui contient (1) une pile d'instructions sémantiques et (2) une mémoire à affectation unique.

État explicite

Dans une procédure, un état explicite est un état dont l'existence s'étend au-delà d'un appel de la procédure sans être présent dans les arguments de la procédure. Dans un langage noyau, un état explicite est un concept qui permet d'accéder à une mémoire à affectation multiple. Différentes possibilités pour ce concept sont la cellule ou le port.

État implicite

Un état qui existe dans l'esprit du programmeur mais pas en tant que concept dans le langage de programmation. Par exemple, la séquence des valeurs des arguments lors des appels successifs d'une fonction forment un état implicite.

F

Factorielle

Une fonction d'un entier n qui donne le nombre de permutations de n objets, c'est-à-dire, le nombre de manières

Glossaire – labo Interactif

différentes de les placer dans une séquence où l'ordre est important.

Forme Étendue de Backus-Naur (EBNF)

Un formalisme pour définir des grammaires formelles, appelée d'après ses inventeurs John Backus et Peter Naur. Le formalisme EBNF distingue des symboles terminaux et des symboles non terminaux. Un symbole terminal est simplement un jeton. Un symbole non terminal est défini avec une règle de grammaire, qui montre comment le convertir en une séquence de jetons.

G

Grammaire

Un ensemble de règles qui définit comment construire des « phrases » à partir des « mots ». Les grammaires peuvent être utilisées pour les langages naturels, comme le français ou le suédois, tout comme les langages artificiels, comme les langages de programmation. Pour les langages de programmation, les « phrases » sont appelées des « instructions » et les « mots » sont appelés des « jetons ».

Grammaire hors-contexte

Une grammaire dans laquelle l'expansion d'un symbole non terminal donne toujours la même séquence de jetons indépendamment de la position du non terminal dans une règle. Voir aussi Grammaire sensible au contexte.

Grammaire sensible au contexte

Une grammaire qui contient une règle dont l'utilisation dépend du contexte dans lequel il est utilisé. Voir aussi Grammaire hors-contexte.

I

Identificateur

Un nom (la chaîne de caractères) qui est dans le texte d'un programme et qui représente une variable en mémoire pendant l'exécution d'un programme.

Identificateur libre

Un identificateur dans une instruction qui n'est pas déclaré dans cette instruction.

Induction mathématique

Une technique de preuve mathématique qui permet de prouver qu'une propriété est vraie pour tous les éléments d'un ensemble bien fondé de valeurs. La technique se fait en deux parties. D'abord, on prouve que la propriété est satisfaite pour l'élément le plus petit de l'ensemble. Ensuite, on prouve que la propriété est satisfaite pour un élément donné si elle est satisfaite pour un élément plus petit. Alors, l'induction mathématique dit que la propriété est satisfaite pour tous les éléments de l'ensemble.

Instruction

Dans une grammaire, une séquence de jetons. Dans un langage de programmation, une instruction définit un ensemble d'actions qui seront faites lors de l'exécution de l'instruction.

Instruction sémantique

Une paire d'une instruction et d'un environnement, ce qui définit la relation entre l'instruction et ce qu'elle référence en mémoire.

J

Jeton

Une séquence de caractères, considérée comme une entité dans une grammaire.

L

Langage formel

Un ensemble de mots. Certains langages formels peuvent être définis par des grammaires.

Glossaire – labo Interactif

Langage noyau

Langage simple qui contient tous les concepts d'un langage pratique sans les ajouts syntaxiques.

Liste

Une séquence d'éléments, avec une syntaxe délimitée à gauche et à droite par des crochets, comme [5 6 7 8]. Dans la mémoire, une liste est, soit une liste vide (écrite nil), soit un élément suivi par une autre liste.

M

Mémoire à affectation unique

Un ensemble de variables, qui sont éventuellement liées. Ces variables sont partitionnées en deux groupes : (1) les ensembles de variables qui sont égales mais non liées à une valeur et (2) les variables qui sont liées à un nombre, un enregistrement ou une procédure.

N

Non-déterminisme

La propriété de l'exécution d'un programme qui permet le système qui l'exécute de choisir une parmi plusieurs instructions pour faire un pas d'exécution. Un programme non-déterministe peut (mais ne doit pas forcément) donner des résultats différents lors des exécutions différentes, même avec les mêmes entrées.

O

Objet

Une fonction avec une mémoire interne (un état explicite) est généralement appelée un objet. Un objet peut avoir d'autres propriétés comme une syntaxe appropriée et une définition par classe.

P

Paire de liste

Une paire d'un élément suivi par une liste, qui définit une nouvelle liste. On l'appelle *cons* en anglais.

Parseur

Un programme qui prend une séquence de jetons et renvoie un arbre syntaxique qui représente une instruction.

Pas d'exécution

Une transition d'un état vers l'état suivant dans un calcul.

Pile sémantique

Une pile d'instructions sémantiques, utilisées pour l'exécution d'une instruction dans la sémantique. Une pile sémantique a trois états possibles pendant l'exécution : exécutable, terminée et suspendue.

Port

Une paire composée d'une constante, qui est un nom, et d'une référence dans la mémoire à affectation unique. Il existe un flot dont la queue non liée est cette référence. Il existe une opération d'envoi qui permet d'ajouter un élément à ce flot en liant la référence. En même temps, la paire est modifiée pour référencer la nouvelle queue du flot. Un port est donc un canal de communication. Les ports implémentent un état explicite. On peut ajouter les ports à un modèle de calcul pour donner un modèle avec état.

Portée

La portée d'une occurrence d'un identificateur est la partie d'un programme pour laquelle cet identificateur correspond à la même variable en mémoire. La portée peut être lexicale (aussi appelée portée statique) ou dynamique.

Glossaire – labo Interactif

Programmation déclarative

Un programme est une fonction ou une relation au sens mathématique. L'exécution d'un programme est l'évaluation de cette fonction ou relation sur des arguments qui sont des structures de données.

Portée dynamique

La portée est déterminée pendant l'exécution du programme. La variable qui correspond à une occurrence d'un identificateur est celle définie dans la déclaration qui contient l'occurrence et qui est la plus récente pendant l'exécution qui mène jusqu'à l'instruction qui contient l'occurrence.

Portée lexicale

Voir Portée statique.

Portée statique

La portée est déterminée par une inspection du code d'un programme. La variable qui correspond à une occurrence d'un identificateur est celle définie dans la déclaration qui contient l'occurrence et qui est la plus proche de l'occurrence dans le texte du programme.

Programmation basée objet

La programmation avec les classes et les objets.

Programmation déclarative

La programmation avec des fonctions. Un programme est une série de définitions de fonction et son résultat est donné par l'évaluation d'une fonction.

Programmation d'ordre supérieur

La capacité d'utiliser des fonctions comme des valeurs : de les créer et exécuter à volonté et de les passer comme arguments et résultats à d'autres fonctions.

Programmation orientée objet

La programmation avec les classes et les objets, où l'on peut définir de nouvelles classes à partir de classes existantes avec l'héritage.

Programmation sans état

Un paradigme appelé aussi programmation déclarative.

R

Restriction

Une fonction qui définit un nouvel environnement dont le domaine est un sous-ensemble du domaine d'un environnement existant.

S

Sémantique

Un modèle mathématique d'un langage de programmation qui définit son comportement lors de l'exécution. On distingue différentes manières de définir la sémantique, dont les plus importantes sont la sémantique opérationnelle, la sémantique axiomatique et la sémantique dénotationnelle.

Sémantique axiomatique

Un modèle mathématique qui définit le sens d'une instruction comme une relation entre l'état de l'entrée (avant l'exécution de l'instruction) et l'état de la sortie (après l'exécution de l'instruction). Cette relation est donnée comme une assertion logique.

Sémantique dénotationnelle

Un modèle mathématique qui définit le sens d'une instruction comme une fonction sur un domaine abstrait.

Sémantique opérationnelle

Glossaire – labo Interactif

Un modèle mathématique qui montre comment une instruction s'exécute sur une machine abstraite.

Spécification d'un programme

Une définition mathématique des entrées dont le programme a besoin et des résultats qu'il calcule.

Syntaxe

Pour un langage de programmation, la définition des instructions légales, c'est-à-dire, qui peuvent être exécutées. La syntaxe d'un langage est définie par une grammaire.

V

Variable

Une entité dans la mémoire du programme qui représente une valeur. Une variable au sens mathématique. Une variable dans ce sens ne peut pas être affectée plus qu'une fois.

Variable en mémoire

Voir Variable