Programmation Objet

Le concept d’objet et de la conception orientée objet

1. Introduction

La programmation structurée (procédurale) préconise deux types de découpage aux modules. Une décomposition descendante (général -> détail) et une décomposition ascendante (détail->général).

La programmation procédurale permet :

* une compréhension plus aisée des programmes
* une fiabilité des programmes
* la construction de type dérivé

Inconvénients de la programmation structurée :

* Le défaut de la réutilisabilité
* Le défaut d’extensibilité
* Le défaut de contrôle

1. La représentation par objets

Exemple : Développer une application dont le rôle est d’analyser si un numéro de carte de crédit ne fait pas l’objet d’une interdiction. Chaque matin, l’application se connecte à un serveur qui lui communique dans un ordre quelconque les numéros de cartes de crédits interdits. Tout au long de la journée, on lui fournit des numéros de carte et il vérifie l’autorisation. Le numéro de carte est supposé représentable comme un entier.

a/ Conception Orientée Action (procédurale)

Tableau d’entier : Tab[1 :N] (Tri à bulle)

Entier : Numerocourant

Booléen

Actions : Rechercher => Trier => Initialiser

b/ Conception Orientée Objet

Objet : Ensemble des numéro de carte

Attributs :

* Numéros de cartes
* Arbres de cartes

Méthode : Vérifier l’existence des numéros de cartes, trier et initialiser seront dans la partie privée du logiciel

1. La conception par objet

a/ Objet

Définition : Un objet informatique est un module regroupant des outils de description et de manipulation d’un objet concret ou abstrait, du monde réel qui le représente.

L’approche orienté objet raisonne en terme d’utilisateur (pas du concepteur). Les étapes de l’approche d’objet sont :

* Identification
* Fonctionnalités
* Interaction entre les objets

Amélioration apporté par l’approche objet sont :

* Facilité de conception
* Localisation
* Localisation (données et méthodes sont dans le même module)
* Intégrité (Un objet est une boite noire)
* Flexibilité (plusieurs représentations sont possible pour le même objet)
* Réutilisabilité

Définition classe : Entité génératrice d’une famille d’objet (Ses instances). Instancier une classe signifie donner naissance à un objet.

Une classe peut être définit à partir d’une ou plusieurs autres classes (ancêtre), par le mécanisme d’héritage (Java, C++, PHPv3, Ruby, Python, Eiffel).

Définition attribut : Propriété relationnelle associé à une classe (Nom, Prénom, Age).

Définition méthode : Propriété fonctionnelle associé à une classe.

Exemple :

Classe Ensentier

{ privee://membre privé

Tab(1 :N) :tableau de type entier

Public://membre publiques

Procédure trie (E/S : Tab(1,N))tableau d’entier)

Fonction rechercher (E/Tab(1 :N)tableau d’entier, Nb :entier) : type booléen.

Définition d’une instance : Une instance est un objet. L’état d’une instance est caractérisé par la valeur de ses attributs

Définition d’héritage : C’est le mécanisme qui permet de composer des interfaces d’une ou plusieurs classes dont une nouvelle classe.

B hérite de A (A : classe de base mère, B : classe dérivé, fille).

Définition de polymorphisme :

Si i € C => f() de C

Si i € C’=> f() de C’

C’est la capacité d’obtenir plusieurs réactions, plus de réaction a une sollicitation donnée (possibilité de donner le même nom a plusieurs fonctions dans des classes différentes).

Définition de l’encapsulation : C’est un regroupement de structures de données et d’un ensemble de fonctions, seul habilité à les manipuler.

Fonctions

Méthodes

Données

A partir de la description informelle du problème, identifier les objets (noms) et les actions (verbes).