Type abstrait ‘Graphe’

 *Constructeur*  : creerGraphe() : Graphe

*Observateur*  : estVide() : Booleen

*Observateur*  : aSommet(Sommet s) : Booleen

*Observateur*  : aArête(Arête a) : Booleen

*Observateur*  : NbSommet() : Entier

*Observateur*  : NbArête() : Entier

*Transformateur* : ajouterSommet(Sommet s) : Graphe

*Transformateur* : ajouterArête(Sommet s) : Graphe

*Transformateur* : supprimerSommet(Sommet s) : Graphe

*Transformateur* : supprimerArête(Sommet s) : Graphe

Type Abstrait ‘Commande’

*Constructeur*  : creerCommande() : Commande

*Constructeur*  : creerCommande(Chaine NomClient) : Commande

*Constructeur*  : creerCommande(Chaine NomClient,

Chaine Destination) : Commande

*Observateur*  : estVide() : Booleen

*Observateur*  : aProduit(Produit p) : Booleen

*Observateur* : aIncomp (Incomp i) : Booleen

*Observateur*  : NbProduit() : Entier

*Observateur*  : NbIncomp() : Entier

*Observateur*  : getClient() : Chaine

*Observateur* : getDestination() : Chaine

*Transformateur* : setNomClient(Chaine NomClient) : Commande

*Transformateur* : setDestination(Chaine Ville) : Commande

*Transformateur* : ajouterProduit(Produit p) : Commande

*Transformateur* : ajouterIncomp(Incomp i) :

*Transformateur* : supprimerProduit(Produit p) : Commande

*Transformateur* : supprimerIncomp(Incomp i) : Commande

Pré-conditions

. Définie( supprimerProduit( p ) ) => aProduit( p ) = Vrai

. Définie( supprimerIncomp( i ) ) => aIncomp( i ) = Vrai

Axiomes

. creerCommande().estVide() = Vrai

. creerCommande(s).estVide() = Vrai

. creerCommande(s1,s2).estVide()

. ajouterProduit(p).estVide () = Faux

. p = q => ajouterProduit(q).aProduit(p) = Vrai

. p <> q => ajouterProduit(q).aProduit(p) = aProduit(p)

. creerCommande().aProduit(p) = Faux

. creerCommande(s).aProduit(p) = Faux

. creerCommande(s1,s2).aProduit(p) = Faux

. i = j => ajouterIncomp(j).aIncomp(i) = Vrai

. i <> j => ajouterIncomp(j).aIncomp(i) = aIncomp(i)

. creerCommande().aIncomp() = Faux

. creerCommande(s).aIncomp() = Faux

. creerCommande(s1,s2).aIncomp() = Faux

. ajouterProduit(p).NbProduit() = 1 + NbProduit()

. creerCommande().NbProduit() = 0

. creerCommande(s).NbProduit() = 0

. creerCommande(s1,s2).NbProduit() = 0

. ajouterIncomp(i).NnIncomp() = 1 + NbIncomp()

. creerCommande().NbIncomp() = 0

. creerCommande(s).NbIncomp() = 0

. creerCommande(s1,s2).NbIncomp() = 0

. setClient(s).getClient() = s

. creerCommande().getClient() = « »

. creerCommande(s).getClient() = s

. creerCommande(s1,s2).getClient() = s1

. setDestination(s).getDestination() = s

. creerCommande().getDestination() = « »

. creerCommande(s).getDestination() = « »

. creerCommande(s1,s2).getDestination() = s2