TriangulatedCategoriesForCAP

framework for triangulated categories

0.1

21/07/2016

Kamal Saleh

Kamal Saleh
Email: kamal.saleh@rwth-aachen.de
Homepage: Kamal.saleh@rwth-aachen.de
Address: Siegen

Contents

Chapter 1

Introduction

1.1 What is this package

bla bla

1.2 Definition of triangulated categories

A triangulated category consists of the following data.

Chapter 2

Creating triangulated categories and their operations

2.1 Creating triangulated category.

bla bla.

2.2 Primitive operations

In the following T is the shift fuctor of the triangulated category $\mathscr{C}$. Its inverse will be denoted by D.

2.2.1 ShiftOfObject (for IsCapCategoryObject)

▷ `ShiftOfObject(a)` (operation)
Returns: object
The argument is an object a. The output is $T(a)$.

2.2.2 ShiftOfMorphism (for IsCapCategoryMorphism)

▷ `ShiftOfMorphism(alpha)` (operation)
Returns: morphism
The argument is a morphism α. The output is $T(\alpha)$.

2.2.3 ReverseShiftOfObject (for IsCapCategoryObject)

▷ `ReverseShiftOfObject(a)` (operation)
Returns: object
The argument is an object a. The output is $D(a)$.

2.2.4 ReverseShiftOfMorphism (for IsCapCategoryMorphism)

▷ `ReverseShiftOfMorphism(alpha)` (operation)
Returns: morphism
The argument is a morphism α. The output is $D(\alpha)$.

2.2.5 IsomorphismFromObjectToShiftAfterReverseShiftOfTheObject (for IsCapCategoryObject)

▷ `IsomorphismFromObjectToShiftAfterReverseShiftOfTheObject(a)` (operation)

Returns: morphism

The argument is an object a. The output is an isomorphism $\alpha : a \to TD(a)$.

2.2.6 IsomorphismFromObjectToReverseShiftAfterShiftOfTheObject (for IsCapCategoryObject)

▷ `IsomorphismFromObjectToReverseShiftAfterShiftOfTheObject(a)` (operation)

Returns: morphism

The argument is an object a. The output is an isomorphism $\alpha : a \to DT(a)$.

2.2.7 IsExactForTriangles (for IsCapCategoryTriangle)

▷ `IsExactForTriangles(tr)` (operation)

Returns: a boolian

The argument is a triangle in the category $\mathscr{C}$. The output is `true` if tr is exact. Otherwise the output is `false`.

2.2.8 CompleteMorphismToExactTriangle (for IsCapCategoryMorphism)

▷ `CompleteMorphismToExactTriangle(alpha)` (operation)

Returns: an exact triangle.

The input is a morphism $\alpha : a \to b$. The output is an exact triangle $a \to b \to c \to T(a)$, in which the morphism from a to b is α.

2.2.9 CompleteToMorphismOfExactTriangles (for IsCapCategoryExactTriangle, IsCapCategoryExactTriangle, IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `CompleteToMorphismOfExactTriangles(tr1, tr2, alpha, beta)` (operation)

Returns: a morphism.

The input is two triangles tr_1, tr_2 and two morphisms α, β as in axiom TR?. The output is a morphism γ that complete the diagram in TR? into a morphism of exact triangles.

2.2.10 OctohedralAxiom (for IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `OctohedralAxiom(alpha, beta)` (operation)

Returns: list.

The input is two morphisms $\alpha : a \to b, \beta : b \to c$. The output is a list of 4 exact triangle satisfying the octohedral axiom.

2.3 General operations

2.3.1 ApplyShift (for IsCapCategoryObject, IsInt)

▷ `ApplyShift(a, n)` (operation)

Returns: object

The argument is an object a. The output is $T^n(a)$.

2.3.2 ApplyShift (for IsCapCategoryMorphism, IsInt)

▷ `ApplyShift(alpha, n)` (operation)

Returns: morphism

The argument is a morphism α. The output is $T^n(\alpha)$.

2.3.3 CreateTriangle (for IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `CreateTriangle(alpha, beta, gamma)` (operation)

Returns: a triangle

The arguments are morphism $\alpha : a \to b, \beta : b \to c, \gamma : c \to T(a)$. The output is the triangle defined by α, β, γ.

2.3.4 ConeObject (for IsCapCategoryMorphism)

▷ `ConeObject(alpha)` (operation)

Returns: object

The input is a morphism $\alpha : a \to b$. The output is an object c such that there exsits an exact triangle $a \to b \to c \to T(a)$, in which the morphism from a to b is α.

2.3.5 CreateExactTriangle (for IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `CreateExactTriangle(alpha, beta, gamma)` (operation)

Returns: a triangle

The arguments are morphism $\alpha : a \to b, \beta : b \to c, \gamma : c \to T(a)$. The output is exact triangle defined by α, β, γ.

2.3.6 CreateExactTriangle (for IsCapCategoryTriangle)

▷ `CreateExactTriangle(tr)` (operation)

Returns: exact triangle

The argument is a triangle tr. The output is an exact triangle that equals to tr as triangles.

2.3.7 IsEqualForTriangles (for IsCapCategoryTriangle, IsCapCategoryTriangle)

▷ `IsEqualForTriangles(tr1, tr2)` (operation)

Returns: a boolian

The output is `true` if $tr_1 = tr_2$. Otherwise it returns `false`.

2.3.8 CreateMorphismOfTriangles (for IsCapCategoryTriangle, IsCapCategoryTriangle, IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `CreateMorphismOfTriangles(tr1, tr2, f, g, h)` (operation)

Returns: morphism of triangles

The output is a morphism of triangles defined by the input data.

2.3.9 PreCompose (for IsCapCategoryTrianglesMorphism, IsCapCategoryTrianglesMorphism)

▷ `PreCompose(mor1, mor2)` (operation)

Returns: morphism of triangles

The input is two morphisms of triangles $mor_1 : tr_1 \to tr_2, mor_2 : tr_2 \to tr_3$. The output is their pre-composition.

2.3.10 PostCompose (for IsCapCategoryTrianglesMorphism, IsCapCategoryTrianglesMorphism)

▷ `PostCompose(mor2, mor1)` (operation)

Returns: morphism of triangles

The input is two morphisms of triangles $mor_1 : tr_1 \to tr_2, mor_2 : tr_2 \to tr_3$. The output is their post-composition.

2.3.11 CompleteToMorphismOfExactTriangles (for IsCapCategoryExactTriangle, IsCapCategoryExactTriangle, IsCapCategoryMorphism, IsCapCategoryMorphism, IsList)

▷ `CompleteToMorphismOfExactTriangles(tr1, tr2, alpha, beta, list)` (operation)

Returns: morphism

The input is two exact triangles and two morphisms α, β and a `list`. The list is allowed to be $[1,2],[1,3]$ or $[2,3]$. It discribes between which objects in tr_1, tr_2 the morphisms α, β are. The output is a morphism that complete the diagram to a morphism of exact triangles.

2.4 Attributes

2.4.1 ShiftFunctor (for IsCapCategory)

▷ `ShiftFunctor(C)` (attribute)

Returns: a functor

The input is finalised triangulated category $\mathscr{C}$. The output is its shift functor T.

2.4.2 ReverseShiftFunctor (for IsCapCategory)

▷ `ReverseShiftFunctor(C)` (attribute)

Returns: a functor

The input is finalised triangulated category $\mathscr{C}$. The output is the inverse of its shift functor, i.e., D.

2.4.3 NaturalIsomorphismFromIdentityToReverseShiftAfterShiftFunctor (for IsCapCategory)

▷ `NaturalIsomorphismFromIdentityToReverseShiftAfterShiftFunctor(C)` (attribute)

Returns: a natural transformation

The input is finalised triangulated category $\mathscr{C}$. The output is the natural isomorphism between the identity functor and $D \circ T$.

2.4.4 NaturalIsomorphismFromIdentityToShiftAfterReverseShiftFunctor (for IsCapCategory)

▷ `NaturalIsomorphismFromIdentityToShiftAfterReverseShiftFunctor(C)` (attribute)

Returns: a natural transformation

The input is finalised triangulated category $\mathscr{C}$. The output is the natural isomorphism between the identity functor and $T \circ D$.

2.4.5 CapCategory (for IsCapCategoryTriangle)

▷ `CapCategory(tr)` (attribute)

Returns: Cap category

The input is a triangle. The output is the category of tr.

2.5 Properties

2.5.1 IsTriangulatedCategory (for IsCapCategory)

▷ `IsTriangulatedCategory(C)` (property)

Returns: a boolian

The output is `true` if the category is triangulated. Otherwise `false`.

2.5.2 IsIsomorphism (for IsCapCategoryTrianglesMorphism)

▷ `IsIsomorphism(mor)` (property)

Returns: a boolian

The output is `true` if *mor* is isomorphism. Otherwise `false`.

Index