

# Comparable Class Documentation

Source File: 'Object.h'  
Namespace: dse  
Class Header: class Object

## Overview

The *Comparable* class is an interface class designed to allow objects to be comparable.

## Member Functions

- `hash() const [private pure virtual]`
  - **Return:** A nonnegative integer.

## Non-Member Functions

- `operator==(const Comparable& lhs,const Comparable& rhs)`
  - **Purpose:** Checks if the hash values of the parameter are the same.
  - **Parameters:**
    - \* *lhs*: Constant reference of an *Comparable* object.
    - \* *rhs*: Constant reference of an *Comparable* object.
  - **Return:** A Boolean.
- `operator!=(const Comparable& lhs,const Comparable& rhs)`
  - **Purpose:** Checks if the hash values of the parameter are different.
  - **Parameters:**
    - \* *lhs*: Constant reference of an *Comparable* object.
    - \* *rhs*: Constant reference of an *Comparable* object.
  - **Return:** A Boolean.
- `operator<(const Comparable& lhs,const Comparable& rhs)`
  - **Purpose:** Checks if the hash value of the first parameter is less than the hash value of the second parameter.
  - **Parameters:**
    - \* *lhs*: Constant reference of an *Comparable* object.
    - \* *rhs*: Constant reference of an *Comparable* object.
  - **Return:** A Boolean.
- `operator<=(const Comparable& lhs,const Comparable& rhs)`
  - **Purpose:** Checks if the hash value of the first parameter is less than or equal to the hash value of the second parameter.
  - **Parameters:**
    - \* *lhs*: Constant reference of an *Comparable* object.
    - \* *rhs*: Constant reference of an *Comparable* object.
  - **Return:** A Boolean.
- `operator>(const Comparable& lhs,const Comparable& rhs)`
  - **Purpose:** Checks if the hash value of the first parameter is greater than the hash value of the second parameter.
  - **Parameters:**
    - \* *lhs*: Constant reference of an *Comparable* object.
    - \* *rhs*: Constant reference of an *Comparable* object.
  - **Return:** A Boolean.
- `operator>=(const Comparable& lhs,const Comparable& rhs)`
  - **Purpose:** Checks if the hash value of the first parameter is greater than or equal to the hash value of the second parameter.
  - **Parameters:**
    - \* *lhs*: Constant reference of an *Comparable* object.
    - \* *rhs*: Constant reference of an *Comparable* object.

- **Return:** A Boolean.
- `operator-(const Comparable& lhs, const Comparable& rhs)`
  - **Purpose:** Retrieves the distance between the hash values of the parameters.
  - **Parameters:**
    - \* *lhs*: Constant reference of an *Comparable* object.
    - \* *rhs*: Constant reference of an *Comparable* object.
  - **Return:** A nonnegative integer.