

Structure Date

Date.h

```

/*****
                        Date.h - description
-----
debut      : 27/11/2015 22:30:09
copyright  : (C) 2015 par Quentin SCHROTER, Nicolas GRIPONT
e-mail     : quentin.schroter@insa-lyon.fr , nicolas.gripont@insa-lyon.fr
*****/

//----- Interface de la classe <Date> (fichier Date.h) -----
#ifndef ! defined ( DATE_H )
#define DATE_H

//----- Interfaces utilisees

//----- Constantes

//----- Types

//-----
// Role de la structure <Date>
// Structure permettant de créer un objet de type date
//
//-----

struct Date
{
//----- PUBLIC

public:
//----- Methodes publiques

//----- Surcharge d'operateurs
    Date & operator = ( const Date & uneDate );
    // Mode d'emploi :
    // Redefinition de l'operateur = pour une Date
    //
    // Contrat : aucun.
    //

    friend ostream & operator << (ostream & os, Date &d);
    // Mode d'emploi :
    // Redefinition de l'operateur << pour un ostream
    //
    // Contrat : aucun.
    //

//----- Constructeurs - destructeur
    Date ( const Date & uneDate );
    // Mode d'emploi : Constructeur de copie
    //

```

```
// Contrat : aucun.
//

Date ( int yyyy, int mm, int dd, int h, int m, int s = 0, int g = 0 );
// Mode d'emploi : Constructeur.
//
// Contrat :
//

virtual ~Date ( );
// Mode d'emploi : Destructeur.
//
// Contrat : aucun.
//

//----- Attributs publics

int annee;    // Valeur de l'annee.

int mois;     // Valeur du mois.

int jour;     // Valeur du jour.

int heure;    // Valeur de l'heure.

int minute;   // Valeur de la minute.

int seconde;  // Valeur de la seconde.

int gmt;      // Valeur de la difference de l'heure avec celle
              // de Greenwich.

};

//----- Types dependants de <Date>

#endif // DATE_H
```

Date.cpp

```
/*
Date - fichier d'en-tête de la classe <Date>
-----
debut      : 27/11/2015 22:30:09
copyright  : (C) 2015 par Quentin SCHROTER, Nicolas GRIPONT
e-mail     : quentin.schroter@insa-lyon.fr , nicolas.gripont@insa-lyon.fr
*/

//----- Realisation de la classe <Date> (fichier Date.cpp) --

//----- INCLUDE

//----- Include systeme
#include <iostream>
using namespace std;
```

```
//----- Include personnel
#include "Date.h"

//----- Constantes

//----- Variables de classe

//----- Types privés

//----- PUBLIC
//----- Fonctions amies

//----- Methodes publiques

//----- Surcharge d'opérateurs
Date & Date::operator = ( const Date & uneDate )
// Algorithme :
//
{
    if (this != &uneDate)
    {
        annee = uneDate.annee;
        mois = uneDate.mois;
        jour = uneDate.jour;
        heure = uneDate.heure;
        minute = uneDate.minute;
        seconde = uneDate.seconde;
        gmt = uneDate.gmt;
    }

    return * this;
} //----- Fin de operator =

ostream & operator << (ostream & os, Date &d)
// Algorithme :
//
{
    os << d.jour << "/" << d.mois << "/" << d.annee << " " << d.heure<<":"
    << d.minute << ":" << d.seconde << " ";
    if ( d.gmt > 0 )
    {
        os << "+" ;
    }
    else
    {
        os << "-";
    }
    if ( d.gmt / 100 < 10 && d.gmt / 100 > 0 )
    {
        os << "0";
    }
    else if ( d.gmt / 100 == 0 )
    {
        os << "00";
    }
}
```

```

    }
    os << d.gmt;

    return os;
} //----- Fin de operator <<

//----- Constructeurs - destructeur
Date::Date ( const Date & uneDate ) : annee(uneDate.annee),
    mois(uneDate.mois), jour(uneDate.jour), heure(uneDate.heure),
    minute(uneDate.minute), seconde(uneDate.seconde), gmt(uneDate.gmt)
// Algorithme :
//
{
#ifdef MAP
    cout << "Appel au constructeur de copie de <Date>" << endl;
#endif
} //----- Fin de Date (constructeur de copie)

Date::Date ( int yyyy, int mm, int dd, int h, int m, int s, int g ) :
    annee(yyyy), mois(mm), jour(dd), heure(h), minute(m), seconde(s),
    gmt(g)
// Algorithme :
//
{
#ifdef MAP
    cout << "Appel au constructeur de <Date>" << endl;
#endif
} //----- Fin de Date

Date::~Date ( )
// Algorithme :
//
{
#ifdef MAP
    cout << "Appel au destructeur de <Date>" << endl;
#endif
} //----- Fin de ~Date

//----- PRIVE

//----- Methodes protegees

//----- Methodes privees

```