

```

/*
Programma : cppClassInt

Autore : Riontino Raffaele classe 4 informatici (serale)
        ITIS Ettore molinari - Milano    17/01/2011

Funzionalità : creazione di una classe per gestire un oggetto
               con le stesse caratteristiche di un intero.
*/

#include <iostream>
using namespace std;

class Int
{
public:
    Int (); //costruttore senza parametri
    Int (int temp); //costruttore con parametri
    void input(); //metodo per l'inserimento dei valori
    void out (); //metodo per la visualizzazione dell'oggetto
    int getValue(); //metodo che ritorna il contenuto dell'oggetto
    void setValue (Int); //metodo per modificare il valore dell'oggetto
    Int operator + (Int); //metodo per effettuare la somma tra gli oggetti
    Int operator - (Int); //metodo per la sottrazione
    Int operator * (Int); //metodo per la moltiplicazione
    Int operator / (Int); //metodo per la divisione
    void operator += (Int); //altro metodo per la somma
    void operator -= (Int); //altro metodo per la sottrazione
    void operator *= (Int); //altro metodo per la moltiplicazione
    void operator /= (Int); //altro metodo per la divisione
    bool operator > (Int); //metodo per confrontare se maggiore
    bool operator < (Int); //metodo per confrontare se minore
    bool operator >= (Int); //metodo per confrontare se maggiore-uguale
    bool operator <= (Int); //metodo per confrontare se minore-uguale
    bool operator == (Int); //metodo per confrontare se uguali
    int operator % (Int); //metodo che ritorna il resto della divisione
    void operator = (Int); //metodo di assegnazione di un valore all'oggetto
    void operator ++ (int); //metodo che incrementa di un valore
    void operator -- (int); //metodo che decrementa di un valore
    ~Int (); //distruttore dell'oggetto

private:
    int dato; //proprietà privata dell'oggetto
}; //fine class Int

Int::Int ()
{
    this->dato = 0; //assegno 0 all'oggetto
}

Int::Int (int temp)
{
    this->dato = temp; //assegno all'oggetto il valore ricevuto
}

void Int::input ()
{
    cin >> this->dato; //inserisco il dato
}

void Int::out()
{
    cout << this->dato; //visualizzo l'oggetto
}

int Int::getValue()

```

```

    {
        return this->dato; //ritorno il contenuto dell'oggetto
    }

void Int::setValue(Int temp)
{
    this->dato = temp.dato; //assegno all'oggetto il valore ricevuto
}

Int Int::operator + (Int temp)
{
    return (this->dato + temp.dato); //somma di due oggetti
}

Int Int::operator - (Int temp)
{
    return (this->dato - temp.dato); //sottrazione di due oggetti
}

Int Int::operator * (Int temp)
{
    return (this->dato * temp.dato); //moltiplicazione di due oggetti
}

Int Int::operator / (Int temp)
{
    return (this->dato / temp.dato); //divisione di due oggetti
}

void Int::operator += (Int temp)
{
    *this = *this + temp; //somma
}

void Int::operator -= (Int temp)
{
    *this = *this - temp; //sottrazione
}

void Int::operator *= (Int temp)
{
    *this = *this * temp; //moltiplicazione
}

void Int::operator /= (Int temp)
{
    *this = *this / temp; //divisione
}

bool Int::operator > (Int temp)
{
    if (this->dato > temp.dato) return true; //confronto se maggiore
    else return false;
}

bool Int::operator < (Int temp)
{
    if (this->dato < temp.dato) return true; //confronto se minore
    else return false;
}

bool Int::operator >= (Int temp)
{
    if (this->dato >= temp.dato) return true; //confronto se maggiore uguale
    else return false;
}

```

```

bool Int::operator <= (Int temp)
{
    if (this->dato <= temp.dato) return true; //confronto se minore uguale
    else return false;
}

bool Int::operator == (Int temp)
{
    if (this->dato == temp.dato) return true; //confronto se uguali
    else return false;
}

int Int::operator % (Int temp)
{
    return (this->dato % temp.dato); //ritorno il resto della divisione
}

void Int::operator = (Int temp)
{
    this->dato = temp.dato; //assegno il valore ricevuto
}

void Int::operator ++ (int)
{
    this->dato += 1; //incremento di un valore
}

void Int::operator -- (int)
{
    this->dato -= 1; //decremento di un valore
}

Int::~Int() //distruttore della classe
{
}

//fine class//

```

```

int main()
{
    cout << "\n\tEsempi di utilizzo della classe Int\n\n";
    Int a; //creo un oggetto di tipo Int, in questo modo a valore 0
    cout << "\noggetto senza valore : a = " << a.getValue() << endl;
    Int b(5); //creo un oggetto con valore 5
    cout << "\noggetto con valore inizializzato : b = " << b.getValue() << endl;
    Int ris; //nuovo oggetto
    ris = a + b; //sommo a e b e li assegno a ris
    cout << "\nris = a + b -> " << ris.getValue() << endl;
    ris = a - b; //sommo a e b e li assegno a ris
    cout << "\nris = a - b -> " << ris.getValue() << endl;
    ris = a * b; //sommo a e b e li assegno a ris
    cout << "\nris = a * b -> " << ris.getValue() << endl;
    ris = a / b; //sommo a e b e li assegno a ris
    cout << "\nris = a / b -> " << ris.getValue() << endl;
    ris += a; //sommo a e b e li assegno a ris
    cout << "\n\nris += a -> " << ris.getValue() << endl;
    ris ++;
    cout << "\nris ++ -> " << ris.getValue() << endl;
    ris += b; //sommo a e b e li assegno a ris
    cout << "\nris += b -> " << ris.getValue() << endl;
    ris --;
    cout << "\nris -- -> " << ris.getValue() << endl;
    system("pause");
}

```

```
system("cls");
cout << "\ninserisci il valore di a : ";
a.input();
cout << "\ninserisci il valore di b : ";
b.input();
cout << "\n\na > b -> ";
if (a > b) cout << " si" ;
else cout << " no";
cout << "\n\na < b -> ";
if (a < b) cout << " si" ;
else cout << " no";
cout << "\n\na >= b -> ";
if (a >= b) cout << " si" ;
else cout << " no";
cout << "\n\na <= b -> ";
if (a <= b) cout << " si" ;
else cout << " no";
cout << "\n\na == b -> ";
if (a == b) cout << " si" ;
else cout << " no";
cout << "\n\ninserisci un valore : ";
ris.input();
if (ris % 2 == 0) cout << "\nnumero pari!\n";
else cout << "\nnumero dispari!\n";
system("pause");
return 1;
} //fine main
```