

Nạp chồng toán tử trong C#

(Operator Overloading)

Operator Overloading là Nạp chồng toán tử. Bạn có thể tái định nghĩa hoặc nạp chồng hầu hết các toán tử có sẵn trong C#. Vì thế, chúng ta có thể sử dụng các toán tử với các kiểu tự định nghĩa (user-defined). Các toán tử được nạp chồng trong C# là các hàm với các tên đặc biệt: từ khóa operator được theo sau bởi biểu tượng cho toán tử đang được định nghĩa. Tương tự như bất kỳ hàm nào khác, một toán tử được nạp chồng có một kiểu trả về và một danh sách tham số.

Cú pháp:

```
<phạm vi truy cập> static <kiểu dữ liệu>  
operator+ (danh sách các đối số)  
{  
    //code  
}
```

Ví dụ, bạn xét hàm sau:

```
public static Box operator+ (Box b, Box c)
{
    Box box = new Box();
    box.chieu_dai = b.chieu_dai + c.chieu_dai;
    box.chieu_rong = b.chieu_rong + c.chieu_rong;
    box.chieu_cao = b.chieu_cao + c.chieu_cao;
    return box;
}
```

Hàm trên triển khai toán tử cộng (+) cho một lớp Box tự định nghĩa (user-defined). Nó cộng các thuộc tính của hai đối tượng Box và trả về đối tượng kết quả Box.

Triển khai Nạp chồng toán tử trong C#

Ví dụ dưới đây minh họa cách triển khai nạp chồng toán tử trong C#: tạo hai lớp có tên lần lượt là **Box**, **TestCsharp** như sau:

Lớp **Box**: chứa các thuộc tính và phương thức

```
using System;

namespace NopChongToanTu
{
    class Box
    {
        private double chieu_dai;
        private double chieu_rong;
        private double chieu_cao;

        public double tinhTheTich()
        {
            return chieu_dai * chieu_rong * chieu_cao;
        }

        public void setChieuDai(double len)
        {
            chieu_dai = len;
        }
    }
}
```

```

        public void setChieuRong(double bre)
        {
            chieu_rong = bre;
        }

        public void setChieuCao(double hei)
        {
            chieu_cao = hei;
        }

        // nap chong toan tu + de cong hai doi tuong Box.
        public static Box operator +(Box b, Box c)
        {
            Box box = new Box();
            box.chieu_dai = b.chieu_dai + c.chieu_dai;
            box.chieu_rong = b.chieu_rong + c.chieu_rong;
            box.chieu_cao = b.chieu_cao + c.chieu_cao;
            return box;
        }
    }
}

```

Lớp **TestCsharp**: chứa phương thức main() để thao tác trên đối tượng Box

```
using System;
namespace NopChongToanTu
{
    public class TestCsharp
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Nap chong toan tu trong C#");
            Console.WriteLine("Minh hoa nap chong toan tu");
            Console.WriteLine("-----");
            //tao cac doi tuong Box1, Box2 va Box3
            Box Box1 = new Box();
            Box Box2 = new Box();
            Box Box3 = new Box();
            double the_tich = 0.0;

            // nhap thong tin Box1
            Box1.setChieuDai(6.0);
            Box1.setChieuRong(7.0);
            Box1.setChieuCao(5.0);

            // nhap thong tin Box2
            Box2.setChieuDai(12.0);
            Box2.setChieuRong(13.0);
            Box2.setChieuCao(10.0);

            // tinh va hien thi the tich Box1
            the_tich = Box1.tinhTheTich();
            Console.WriteLine("The tich cua Box1 la
: {0}", the_tich);
```

```

        // tính và hiển thị thể tích Box2
        the_tich = Box2.tinhTheTich();
        Console.WriteLine("The tích của Box
2 là: {0}", the_tich);

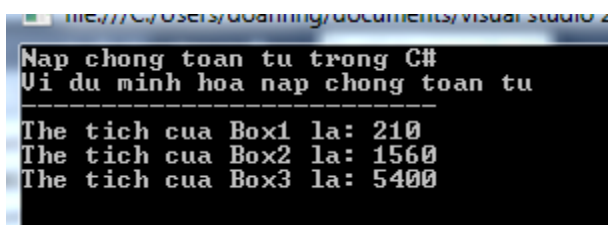
        // con hai đối tượng
        Box3 = Box1 + Box2;

        // tính và hiển thị thể tích Box3
        the_tich = Box3.tinhTheTich();
        Console.WriteLine("The tích của Box
3 là: {0}", the_tich);

        Console.ReadKey();
    }
}
}

```

Biên dịch và chạy chương trình C# trên sẽ cho kết quả sau:



```

Nap chong toan tu trong C#
Vi du minh hoa nap chong toan tu
-----
The tích của Box1 là: 210
The tích của Box2 là: 1560
The tích của Box3 là: 5400

```

Toán tử có thể nạp chồng và không thể nạp chồng trong C#

Bảng dưới miêu tả các toán tử có thể nạp chồng trong C#:

Toán tử	Miêu tả
+, -, !, ~, ++, --	Những toán tử một ngôi này nhận một toán hạng và có thể được nạp chồng
+, -, *, /, %	Những toán tử nhị phân này nhận một toán hạng và có thể được nạp chồng
=, !=, <, >, <=, >=	Các toán tử so sánh có thể được nạp chồng
&&,	Các toán tử logic điều kiện không thể được nạp chồng một cách trực tiếp

<code>+=, -=, *=, /=, %=</code>	Các toán tử gán không thể được nạp chồng
<code>=, ., ?:, ->, new, is, sizeof, typeof</code>	Các toán tử này không thể được nạp chồng

Ví dụ

Từ các khái niệm trên, chúng ta kế thừa ví dụ trên và nạp chồng thêm một số toán tử trong C#: ở trên chúng ta đã tạo hai lớp **Box**, **TestCsharp**, bây giờ chúng ta sửa lại code của mỗi lớp như sau:

Lớp **Box**: chứa các thuộc tính và phương thức

```
using System;

namespace NopChongToanTu
{
```

```
class Box
{
    private double chieu_dai;
    private double chieu_rong;
    private double chieu_cao;

    public double tinhTheTich()
    {
        return chieu_dai*chieu_rong*chieu_cao;
    }

    public void setChieuDai(double len)
    {
        chieu_dai = len;
    }

    public void setChieuRong(double bre)
    {
        chieu_rong = bre;
    }

    public void setChieuCao(double hei)
```

```

    {
        chieu_cao = hei;
    }

    // nap chong toan tu +
    public static Box operator +(Box b, Box c)
    {
        Box box = new Box();
        box.chieu_dai=b.chieu_dai+c.chieu_dai;
        box.chieu_rong=b.chieu_rong+c.chieu_rong;
        box.chieu_cao = b.chieu_cao +c.chieu_cao;
        return box;
    }

    //nap chong toan tu ==
    public static bool operator ==(Box lhs, Box rhs)
    {
        bool status = false;
        if (lhs.chieu_dai == rhs.chieu_dai
        && lhs.chieu_cao == rhs.chieu_cao && lhs.chieu_
        rong == rhs.chieu_rong)
        {
            status = true;
        }
        return status;
    }

```

```

    }

    //nap chong toan tu !=
    public static bool operator !=(Box lhs, Box rhs)
    {
        bool status = false;
        if (lhs.chieu_dai != rhs.chieu_dai
            || lhs.chieu_cao != rhs.chieu_cao || lhs.chieu_
            rong != rhs.chieu_rong)
        {
            status = true;
        }
        return status;
    }

    //nap chong toan tu <
    public static bool operator <(Box lhs, Box rhs)
    {
        bool status = false;
        if (lhs.chieu_dai < rhs.chieu_dai &
            & lhs.chieu_cao < rhs.chieu_cao && lhs.chieu_ro
            ng < rhs.chieu_rong)
        {
            status = true;
        }
        return status;
    }

```

```

    }

    //nap chong toan tu >
    public static bool operator >(Box lhs, Box rhs)
    {
        bool status = false;
        if (lhs.chieu_dai > rhs.chieu_dai &
& lhs.chieu_cao > rhs.chieu_cao && lhs.chieu_ro
ng > rhs.chieu_rong)
        {
            status = true;
        }
        return status;
    }

    //nap chong toan tu <=
    public static bool operator <=(Box lhs, Box rhs)
    {
        bool status = false;
        if (lhs.chieu_dai <= rhs.chieu_dai
&& lhs.chieu_cao <= rhs.chieu_cao && lhs.chieu_
rong <= rhs.chieu_rong)
        {
            status = true;
        }
        return status;
    }

```

```

    }

    //nap chong toan tu >=
    public static bool operator >=(Box lhs, Box rhs)
    {
        bool status = false;
        if (lhs.chieu_dai >= rhs.chieu_dai
        && lhs.chieu_cao >= rhs.chieu_cao && lhs.chieu_
        rong >= rhs.chieu_rong)
        {
            status = true;
        }
        return status;
    }

    //nap chong phuong thuc ToString()
    public override string ToString()
    {
        return String.Format("({0}, {1}, {2
        })", chieu_dai, chieu_rong, chieu_cao);
    }
}
}

```

Lớp **TestCsharp**: chứa phương thức main() để thao tác trên đối tượng Box

```

using System;
namespace NopChongToanTu
{
    public class TestCsharp
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Nap chong toan t
u trong C#");
            Console.WriteLine("Vi du minh hoa n
ap chong toan tu");
            Console.WriteLine("-----
-----");
            //tao cac doi tuong Box1, Box2, Box
2, Box4

            Box Box1 = new Box();
            Box Box2 = new Box();
            Box Box3 = new Box();
            Box Box4 = new Box();
            double the_tich = 0.0;

            // nhap thong tin Box1
            Box1.setChieuDai(6.0);
            Box1.setChieuRong(7.0);
            Box1.setChieuCao(5.0);

            // nhap thong tin Box2
            Box2.setChieuDai(12.0);
            Box2.setChieuRong(13.0);
            Box2.setChieuCao(10.0);

```

```
//hien thi thong tin cac Box boi su
dung phuong thuc nap chong ToString():
    Console.WriteLine("Thong tin Box 1:
{0}", Box1.ToString());
    Console.WriteLine("Thong tin Box 2:
{0}", Box2.ToString());

    // tinh the tich Box1
    the_tich = Box1.tinhTheTich();
    Console.WriteLine("The tich cua Box
1 la: {0}", the_tich);

    // tinh the tich Box2
    the_tich = Box2.tinhTheTich();
    Console.WriteLine("The tich cua Box
2 la: {0}", the_tich);

    // cong hai doi tuong
    Box3 = Box1 + Box2;
    Console.WriteLine("Thong tin Box 3:
{0}", Box3.ToString());

    // tinh the tich cua Box3
    the_tich = Box3.tinhTheTich();
    Console.WriteLine("The tich cua Box
3 la: {0}", the_tich);

    //so sanh cac Box
    if (Box1 > Box2)
        Console.WriteLine("Box1 la lon
hon Box2");
```



```
        else
            Console.WriteLine("Box1 la khon
g lon hon Box2");

        if (Box1 < Box2)
            Console.WriteLine("Box1 la nho
hon Box2");
        else
            Console.WriteLine("Box1 la khon
g nho hon Box2");

        if (Box1 >= Box2)
            Console.WriteLine("Box1 la lon
hon hoac bang Box2");
        else
            Console.WriteLine("Box1 la khon
g lon hon hoac bang Box2");

        if (Box1 <= Box2)
            Console.WriteLine("Box1 la nho
hon hoac bang Box2");
        else
            Console.WriteLine("Box1 la khon
g nho hon hoac bang Box2");

        if (Box1 != Box2)
            Console.WriteLine("Box1 la khon
g bang Box2");
        else
            Console.WriteLine("Box1 bang Bo
x2");
```

```

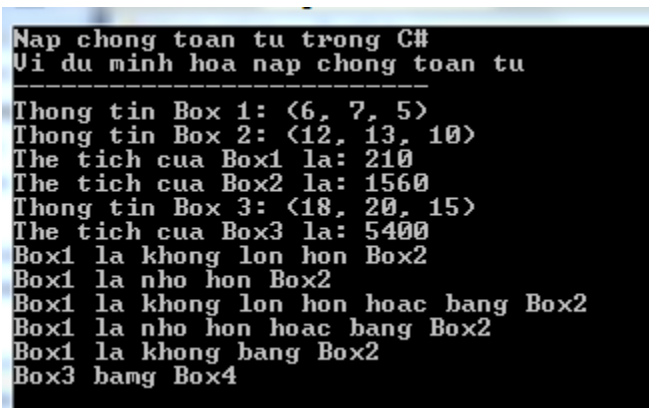
        Box4 = Box3;

        if (Box3 == Box4)
            Console.WriteLine("Box3 bang Bo
x4");
        else
            Console.WriteLine("Box3 la khon
g bang Box4");

        Console.ReadKey();
    }
}
}

```

Biên dịch và chạy chương trình C# trên sẽ cho kết quả sau:



```

Nap chong toan tu trong C#
Vi du minh hoa nap chong toan tu
-----
Thong tin Box 1: <6, 7, 5>
Thong tin Box 2: <12, 13, 10>
The tich cua Box1 la: 210
The tich cua Box2 la: 1560
Thong tin Box 3: <18, 20, 15>
The tich cua Box3 la: 5400
Box1 la khong lon hon Box2
Box1 la nho hon Box2
Box1 la khong lon hon hoac bang Box2
Box1 la nho hon hoac bang Box2
Box1 la khong bang Box2
Box3 bang Box4

```

Bài tập

Viết chương trình C# sử dụng khái niệm nạp chồng toán tử để cộng, trừ, nhân, chia, so sánh hai phân số.

