String Class - Problem on Copy Constructor

Write down a class named **String**. It will have the following skeleton.

```
class String
{
    char * pt;
    int len;
public:
    //Write appropriate Constructors and Destructor
    char get(int index);
        //return the character at the given index
        //Check if index is out of bound,
        //return 0 in case of error
    int put(int index, char c);
        //Assign character c at position index
        //Check if index is out of bound
        //return -1 in case of error
    int getLength();
        //return the allocation length of the string
    void print();
        //print the string upto allocation size
};
```

```
int main()
{
    String s1(4);
    String s2(4);

    s1.put(0, 'A');
    s1.put(1, 'B');
    s1.put(2, 'C');
    s1.put(3, 'D');
    s1.print();     //ABCD

    s2.put(0, '1');
    s2.put(1, '2');
    s2.put(2, '3');
    s2.put(3, '4');
    s2.print();     //1234

    String s3 = concat(s1, s2);
    s3.print();     //ABCD1234
    insert(s3, 2, 'Z');
    s3.print();     //ABZCD123

}
```

String Class - Problem on Copy Constructor

Write down the following non-member functions.

String concat(String s1, String s2)

Joins two strings s1 and s2 and returns the resultant string.

Bonus:

void insert(String &st, int index, char c)

Insert char c at index position of st. Shift other characters to the right.

Make sure the main function written on the right works correctly.

```
int main()
{
    String s1(4);
    String s2(4);

    s1.put(0, 'A');
    s1.put(1, 'B');
    s1.put(2, 'C');
    s1.put(3, 'D');
    s1.print();     //ABCD

    s2.put(0, '1');
    s2.put(1, '2');
    s2.put(2, '3');
    s2.put(3, '4');
    s2.print();     //1234

    String s3 = concat(s1, s2);
    s3.print();     //ABCD1234
    insert(s3, 2, 'Z');
    s3.print();     //ABZCD123

}
```