1. Operator Overloading And Copy Constructor:

   Complete the class definition with appropriate overloaded operators so that the statements in the provided main function can be executed.

   You should write any additional functions as needed.

   The output of your program should match the sample output.

```
class String{
        private:

        char *s;
        int len;

        public:
        String(){};
        String(char *p){};

        …….

};

int main(){
        String s("BUET");
        String s1,s2;

        cout <<s.getString()<<endl;
        // Should concat "CSE" at the end of "BUET"
        s = s+"CSE";
        cout<<s.getString()<<endl;
        // Should concat "108" at the end of "BUETCSE"
        s1 = s = s+108;
        cout<<s1.getString()<<endl;
        // Should reverse "BUETCSE108" and make it “801ESCTUEB”
        s= - s;
        cout<<s.getString()<<endl;
        // s1 should be concatenated 3 times and stored in s2. If s1 contains “abc”, s2
        should now contain “abcabcabc”
        s2=s1*3;

        cout<<s2.getString()<<endl;

}
```

### Sample Output

```
BUET
BUETCSE
BUETCSE108
801ESCTEUB
BUETCSE108BUETCSE108BUETCSE108
```