Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology
CSE210: Object-Oriented Programming Language
Class Test-2, Marks-20, Time-25 Minutes

Q. Implement the constructors and the destructor and overload the appropriate functions and operators for the following C++ class so that the objects of the class can perform as shown in the main() function. You are allowed to add more member variables and functions in the class if you think necessary.

```
class MyString {
        char *str;
public:
        MyString() ; //Constructor sets string str pointing null
        MyString(char *s) ; //Constructor safely copies string s into string str
        ~MyString() ; /// Destructor releases the allocated memory of str
        void show(){cout<<str<<endl;}
        ……
        ……

};

int main(){
        char str[100]="I love C++";
        MyString s1(str), s2;
        s2++;
        s2.show(); //Prints
        ++s2;
        s2.show(); //Prints
        s2--;
        s2.show(); //Prints
        --s2;
        s2.show(); //Prints
        --s1;
        s1.show(); //Prints "I love C++ "
        s1--;
        s1.show(); //Prints "I love C++"
        ++s1;
        s1.show(); //Prints "I love C++I love C++"
        s1++;
        s1.show(); //Prints "I love C++I love C++I love C++"
        --s1;
        s1.show(); //Prints "I love C++I love C++"
        s1--;
        s1.show(); //Prints "I love C++"
        s2=++s1;
        s1.show(); //Prints "I love C++I love C++"
        s2.show(); //Prints "I love C++I love C++"
        s2=s1++;
        s1.show(); //Prints "I love C++I love C++I love C++"
        s2.show(); //Prints "I love C++I love C++"
        s2=--s1;
        s1.show(); //Prints "I love C++I love C++"
        s2.show(); //Prints "I love C++I love C++"
        s2=s1--;
        s1.show(); //Prints "I love C++"
        s2.show(); //Prints "I love C++I love C++"
        return 0;
}
```

Answer:

```
#include <iostream>
#include <cstring>

using namespace std;

class String{

	char *str;
	int length;

public:
	String(){str=0; length=0;}
	String(char*);
	String(const String &);
	~String();
	String operator+(String s);
	String &operator++();
	String &operator--();
	String operator+(char*);
	String operator-(String s);
	String operator*(int n);
	String &operator=(String s);
	char* get_str(){return str;}
	friend ostream& operator<<(ostream &out, String ob);
	friend istream& operator>>(istream &in, String &ob);

};

String::String(char *s){
	int len=strlen(s);
	str=new char[len+1];
	strcpy(str,s);
	length=len;
}

String::String(const String &s){
	int len=strlen(s.str);
	str=new char[len+1];
	strcpy(str, s.str);
	length=len;
}

String::~String(){delete [] str;}

String String::operator + (String s){
	String temp;
	temp.str=new char[strlen(str)+strlen(s.str)+2];
	strcpy(temp.str, str);
	strcat(temp.str, " ");
	strcat(temp.str, s.str);
	return temp;
}

String String::operator + (char *s){
	String temp;
	temp.str=new char[strlen(str)+strlen(s)+2];
	strcpy(temp.str, str);
	strcat(temp.str, " ");
	strcat(temp.str, s);
	return temp;
```

```
}

String &String::operator++(){
	int len=strlen(str);
	if(len==0)
		return *this;
	char *inc=new char[length+1];
	int i;
	for(i=0; i<length; i++){
		inc[i]=str[i];
	}
	inc[i]=0;
	char *temp=new char[len+length+1];
	strcpy(temp, str);
	strcat(temp, inc);
	delete []inc;
	delete []str;
	str=temp;
	return *this;
}

String &String::operator --(){
	int len=strlen(str);
	if(len==0||len==length)
		return *this;
	char *temp=new char[len-length+1];
	int i;
	for(i=0; i<len-length; i++){
		temp[i]=str[i];
	}
	temp[i]=0;
	delete []str;
	str=temp;
	return *this;
}

String &String::operator = (String s){
	str=new char[strlen(s.str)+1];
	strcpy(str, s.str);
	return *this;
}

String String::operator *(int n){
	String temp;
	temp.str=new char[strlen(str)*n+n];
	strcpy(temp.str, str);
	for(int i=1; i<n; i++){
		strcat(temp.str, " ");
		strcat(temp.str, str);
	}
	return temp;
}

ostream& operator<<(ostream &out, String ob){
	out<<"String: ";
	out<<ob.str;
	out<<endl;
	return out;
}

istream& operator>>(istream &in, String &ob){
	cout<<"Enter a string: ";
```

```
	ob.str=new char[100];
	in>>ob.str;
	ob.length=strlen(ob.str);
	return in;
}


int main(){
	char str1[100]="I love";
	char str2[100]="Programming";
	String s1(str1), s2(str2), s3, s4;
	cout<<"s1.str: "<<s1.get_str()<<endl;
	cout<<"s2.str: "<<s2.get_str()<<endl;
	s3=s1+s2;
	s3=s3+"I feel great";
	cout<<"s3.str: "<<s3.get_str()<<endl;
	s1=s1*3;
	cout<<"s1.str*3: "<<s1.get_str()<<endl;
	cin>>s4;
	cout<<s4;
	++s4;
	cout<<s4;
	++s4;
	cout<<s4;
	--s4;
	cout<<s4;
	--s4;
	cout<<s4;
	return 0;
}
```