Rajvaibhav Rahane

17u283 223045

SE-C Comp,Viit,Pune

***CODE:***

/\*

\*

\*@Rajvaibhav Rahane

\*/

/\*

\* Program to replicate String class

\* implements default and copy constructors

\* implements functions length(),concat(),getReversedString(),indexOf(),isPalindrome()

\* overloads operators >>,<< for standard i/o operations

\*/

#include<bits/stdc++.h>

#include<stdio.h>

using namespace std;

class Strings{

private:

char \*str;

int strLength;

void calculateLength();

public :

Strings(){

str=NULL;

strLength=0;

}

Strings(Strings\* s){

str=new char[s->length()];

strLength=s->length();

int i;

for(i=0;i<s->length();i++){

str[i]=s->str[i];

}

str[i]='\0';

cout<<"Copy Constructor called\n";

}

//void createString(int length);

int length();

Strings concat(Strings);

bool isPalindrome();

Strings copy();

int compareTo(Strings);

friend ostream & operator<<(ostream &,const Strings s);

friend istream & operator>>(istream &in,Strings &s);

Strings getReversedString();

int indexOf(Strings);

};

bool Strings::isPalindrome(){ //returns whether string is palindrome or not

if(str!=NULL){

for(int i=0,j=strLength-1;i<j;i++,j--){

if(str[i]!=str[j])

return false;

}

}

return true;

}

void Strings::calculateLength(){

if(str!=NULL)

for(strLength=0;str[strLength]!='\0';strLength++);

}

int Strings::length(){

return strLength;

}

/\*void Strings::createString(int length){

if(length!=0){

cin>>str;

//cout<<"strlen"<<strlen(str)<<"called";

//str[length]='\0';

calculateLength();

}

}\*/

istream & operator>>(istream &in,Strings &s){

s.str=new char[0];

scanf("%[^\n]%\*c",s.str);

//in>>s.str;

s.calculateLength();

s.str[s.length()]='\0';

return in;

}

ostream & operator<<(ostream &out,Strings s){

if(s.str!=NULL)

out<<s.str;

else

out<<"NULL String";

return out;

}

Strings Strings::concat(Strings str){

Strings resultant;

int i;

resultant.str=new char[this->length()+str.length()+1];

for(i=0;i<this->length();i++){

resultant.str[i]=this->str[i];

}

for(int j=0;i<this->length()+str.length();i++,j++){

resultant.str[i]=str.str[j];

}

resultant.str[i]='\0';

resultant.calculateLength();

return resultant;

}

Strings Strings::getReversedString(){

Strings str(this);

char temp;

for(int i=0,j=str.strLength-1;i<j;i++,j--){

temp=str.str[i];

str.str[i]=str.str[j];

str.str[j]=temp;

}

return str;

}

Strings Strings::copy(){

return Strings(this);

}

int Strings::compareTo(Strings s){

if(this->str==NULL){

if(s.str==NULL) return 0; //both strings equal

else return -2; //main string lexographically lower

}else if(s.str==NULL) return 2; //s is lower

else{

int i;

for(i=0;i<s.length()&& i<strLength;i++){

if(s.str[i]!=str[i]){

return str[i]-s.str[i];

}

}

if(i!=s.length())return -1;

else if(i!=strLength) return 1;

return 0;

}

}

int Strings::indexOf(Strings substring){

int index=-1,k,j;

for(int i=0;i<=this->length()-substring.length();i++){

if(this->str[i]==substring.str[0]){

index=i;

for(j=i+1,k=1;k<substring.length();j++,k++){

if(this->str[j]!=substring.str[k]){

index=-1;

break;

}

}

if(index!=-1)

return index;

}

}

return index;

}

void printMenu(){

cout<<"1)Print Length\t";

cout<<"2)Check is string Palindrome\t";

cout<<"3)Compare 2 Strings\t";

cout<<"4)Copy a String\t";

cout<<"5)Reverse a String\t";

cout<<"6)Find substring in String\n";

cout<<"7)Exit\t Choice : ";

}

int main(){

int choice;

do{

printMenu();scanf("%d%\*c",&choice);

switch(choice){

case 1:{

Strings s;cout<<"Enter String:";cin>>s;

cout<<s<<" "<<s.length()<<endl;break;

}

case 2:{

Strings s;cout<<"Enter String:";cin>>s;

cout<<s<<" is "<<(s.isPalindrome()?"":"not ")<<"palindrome"<<endl;break;

}

case 3:{

Strings s1;cout<<"Enter String:";cin>>s1;

Strings s2;cout<<"Enter String:";cin>>s2;

cout<<s1.compareTo(s2)<<endl;

break;

}

case 4:{

Strings s;cout<<"Enter String:";cin>>s;

Strings copy=s.copy();cout<<copy<<endl;break;

}

case 5:{

Strings s;cout<<"Enter String:";cin>>s;

cout<<"Reversed String : "<<s.getReversedString()<<endl;break;

}

case 6:{

Strings s;cout<<"Enter String:";cin>>s;

Strings sub;cout<<"Enter Substring:";cin>>sub;

int index=s.indexOf(sub);

cout<<"Substring ";

if(index==-1) cout<<"absent";

else cout<<"present at index(0 Based) "<<index;

cout<<endl;break;

}

case 7:{break;}

}

}while(choice!=7);

/\*Strings s,s2;string raj;

/\*int length;

cout<<"Enter Length";

cin>>length;

s.createString(length);

cin>>s;

cout<<s;

cout<<" "<<s.length()<<endl;

cin>>s2;

cout<<s2;

cout<<" "<<s2.length()<<endl;

Strings concat=s.concat(s2);

cout<<concat;

cout<<" "<<concat.length()<<endl;

cout<<s<<" "<<s2<<endl;

Strings reversed=s.getReversedString();

cout<<reversed<<" "<<reversed.length()<<"\t"<<s<<endl;

cout<<s2.indexOf(s)<<endl;

cout<<s<<"\t"<<s2<<endl;

cin>>raj;

cout<<raj;

cout<<" "<<raj.length();\*/

return 0;

}

***Output:***



