STRING CONCATENATION:

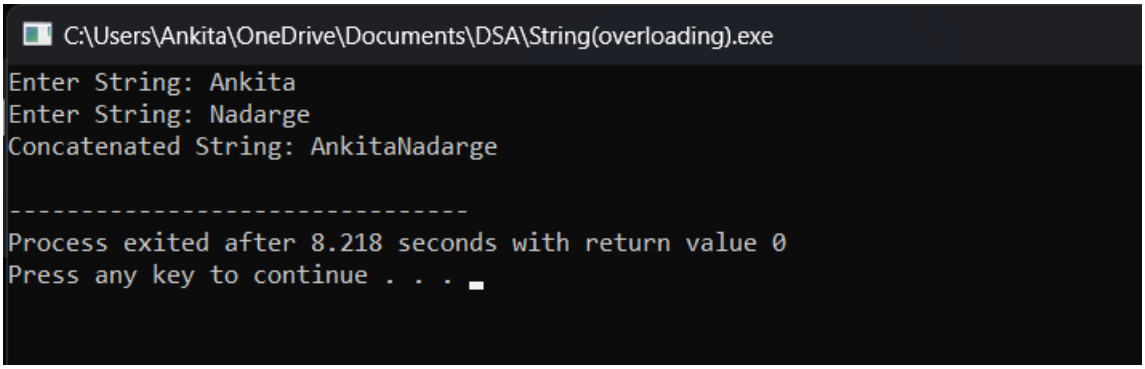
INPUT:

```
#include<iostream>
#include<string>
using namespace std;
class Mystring{
	string str;
	public:
		void input(){
			cout<<"Enter String: ";
			cin>>str;
		}
		void display()
		{
			cout<<"Concatenated String: "<<str<<endl;
		}
		Mystring operator+(Mystring obj)
		{
			Mystring temp;
			temp.str = str + obj.str;
			return temp;
		}
};
int main()
{
	Mystring s1,s2,s3;
	s1.input();
	s2.input();

	s3 = s1 + s2;
	s3.display();

	return 0;
}
```

OUTPUT:



MATRIX ADDITION

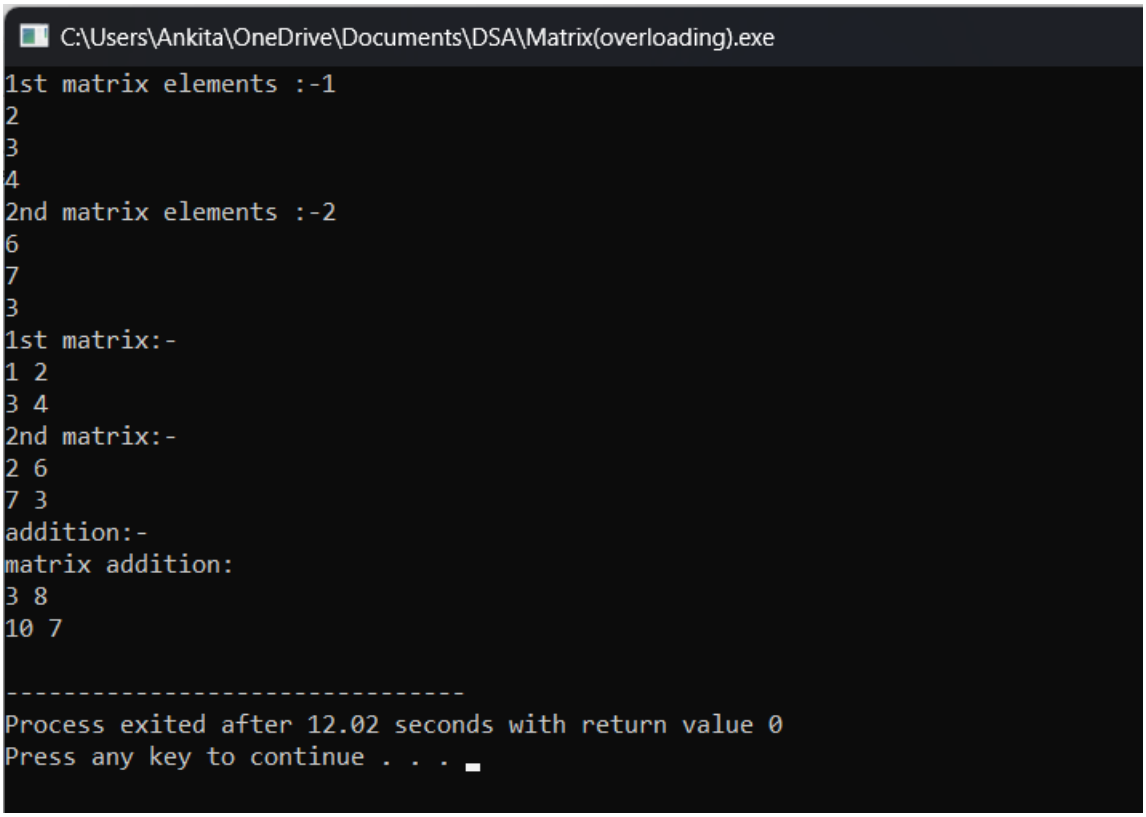
INPUT:

```
#include<iostream>
using namespace std;
class matrix12
{
   int a[2][2],i,j;
   public:
      void getdata();
      void showdata();
      void operator +(matrix12 obj);
};
void matrix12::getdata()
{
   for(i=0;i<2;i++)
   {
      for(j=0;j<2;j++)
```

```
      {
         cin>>a[i][j];
      }
   }
}
void matrix12::showdata()
{
   for(i=0;i<2;i++)
   {
      for(j=0;j<2;j++)
      {
         cout<<a[i][j]<<ends;
      }
      cout<<"\n";
   }
}
void matrix12::operator + (matrix12 obj)
{
   int b[2][2];
   for(i=0;i<2;i++)
   {
      for(j=0;j<2;j++)
      {
         b[i][j]=a[i][j]+obj.a[i][j];
      }
   }
   cout<<"matrix addition:"<<endl;
   for(i=0;i<2;i++)
   {
      for(j=0;j<2;j++)
      {
         cout<<b[i][j]<<ends;
      }
      cout<<"\n";
   }
}
int main()
{
   matrix12 ob1,ob2;
   cout<<"1st matrix elements :-";
   ob1.getdata();
   cout<<"2nd matrix elements :-";
   ob2.getdata();
   cout<<"1st matrix:-"<<endl;
```

```
    ob1.showdata();
    cout<<"2nd matrix:-"<<endl;
    ob2.showdata();
    cout<<"addition:-"<<endl;
    ob1+ob2;
    return 0;
}
```

OUTPUT:



STRING CONCATENATION:

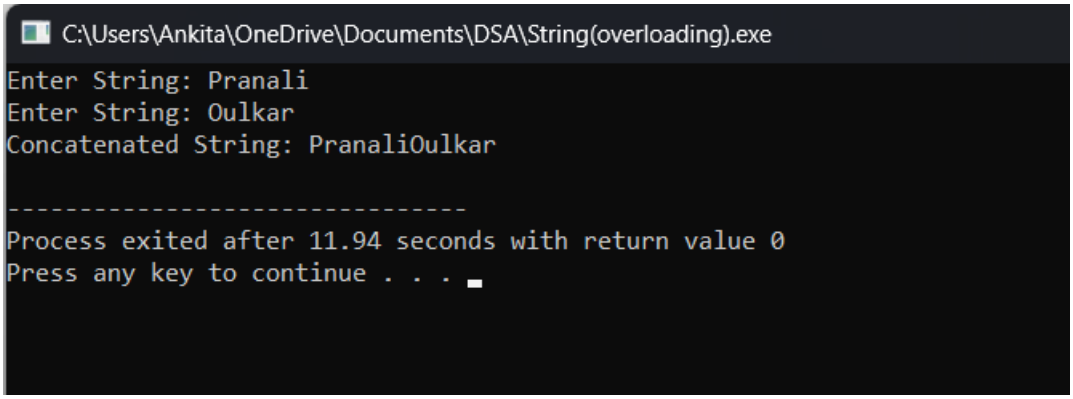
INPUT:

```
#include<iostream>
#include<string>
using namespace std;
class Mystring{
	string str;
	public:
		void input(){
			cout<<"Enter String: ";
			cin>>str;
		}
		void display()
		{
			cout<<"Concatenated String: "<<str<<endl;
		}
		Mystring operator+(Mystring obj)
		{
			Mystring temp;
			temp.str = str + obj.str;
			return temp;
		}
};
int main()
{
	Mystring s1,s2,s3;
	s1.input();
	s2.input();

	s3 = s1 + s2;
	s3.display();

	return 0;
}
```

OUTPUT:



MATRIX ADDITION

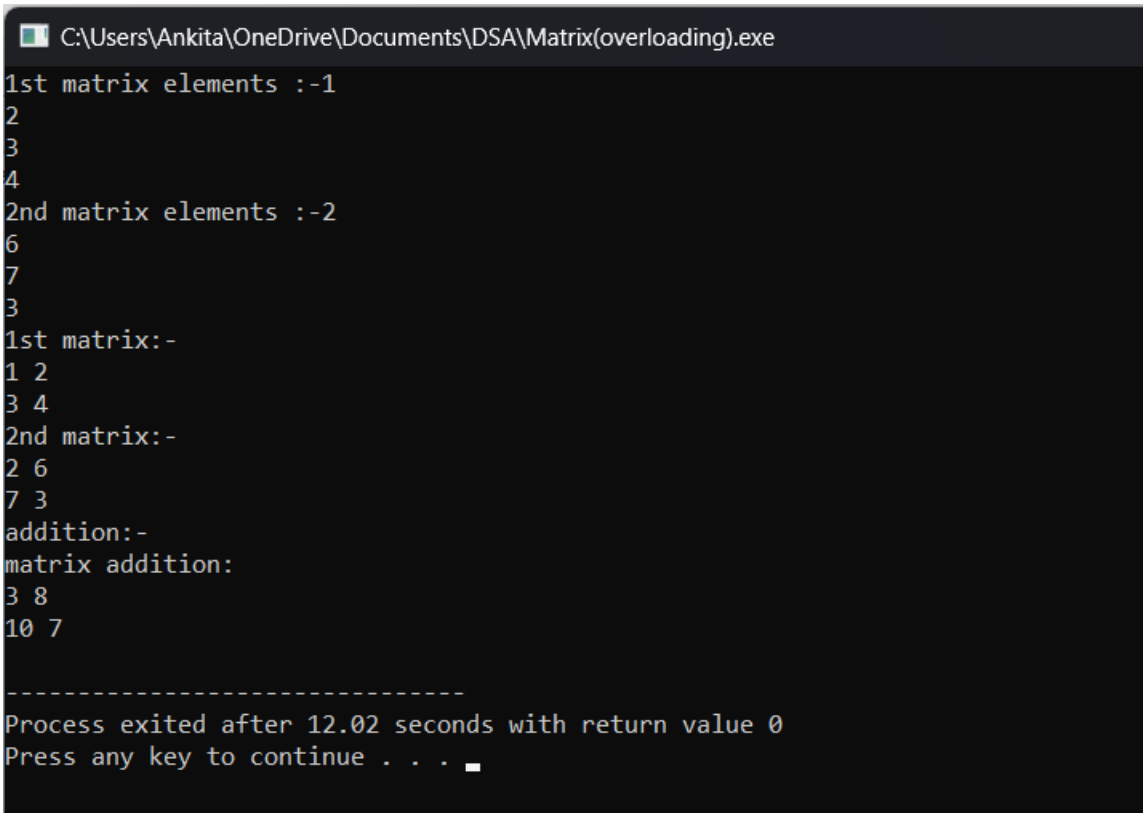
INPUT:

```
#include<iostream>
using namespace std;
class matrix12
{
   int a[2][2],i,j;
   public:
      void getdata();
      void showdata();
      void operator +(matrix12 obj);
};
void matrix12::getdata()
{
   for(i=0;i<2;i++)
   {
      for(j=0;j<2;j++)
      {
         cin>>a[i][j];
```

```
        }
      }
}
void matrix12::showdata()
{
   for(i=0;i<2;i++)
   {
      for(j=0;j<2;j++)
      {
         cout<<a[i][j]<<ends;
      }
      cout<<"\n";
   }
}
void matrix12::operator + (matrix12 obj)
{
   int b[2][2];
   for(i=0;i<2;i++)
   {
      for(j=0;j<2;j++)
      {
         b[i][j]=a[i][j]+obj.a[i][j];
      }
   }
   cout<<"matrix addition:"<<endl;
   for(i=0;i<2;i++)
   {
      for(j=0;j<2;j++)
      {
         cout<<b[i][j]<<ends;
      }
      cout<<"\n";
   }
}
int main()
{
   matrix12 ob1,ob2;
   cout<<"1st matrix elements :-";
   ob1.getdata();
   cout<<"2nd matrix elements :-";
   ob2.getdata();
   cout<<"1st matrix:-"<<endl;
   ob1.showdata();
   cout<<"2nd matrix:-"<<endl;
```

```
    ob2.showdata();
    cout<<"addition:-"<<endl;
    ob1+ob2;
    return 0;
}
```

OUTPUT:



STRING CONCATENATION:

INPUT:

```
#include<iostream>
#include<string>
using namespace std;
class Mystring{
	string str;
	public:
		void input(){
			cout<<"Enter String: ";
			cin>>str;
		}
		void display()
		{
			cout<<"Concatenated String: "<<str<<endl;
		}
		Mystring operator+(Mystring obj)
		{
			Mystring temp;
			temp.str = str + obj.str;
			return temp;
		}
};
int main()
{
	Mystring s1,s2,s3;
	s1.input();
	s2.input();

	s3 = s1 + s2;
	s3.display();

	return 0;
}
```

OUTPUT:

MATRIX ADDITION

INPUT:

```
#include<iostream>
using namespace std;
class matrix12
{
   int a[2][2],i,j;
   public:
      void getdata();
      void showdata();
      void operator +(matrix12 obj);
};
void matrix12::getdata()
{
   for(i=0;i<2;i++)
   {
      for(j=0;j<2;j++)
      {
```

```
            cin>>a[i][j];
        }
    }
}
void matrix12::showdata()
{
    for(i=0;i<2;i++)
    {
        for(j=0;j<2;j++)
        {
            cout<<a[i][j]<<ends;
        }
        cout<<"\n";
    }
}
void matrix12::operator + (matrix12 obj)
{
    int b[2][2];
    for(i=0;i<2;i++)
    {
        for(j=0;j<2;j++)
        {
            b[i][j]=a[i][j]+obj.a[i][j];
        }
    }
    cout<<"matrix addition:"<<endl;
    for(i=0;i<2;i++)
    {
        for(j=0;j<2;j++)
        {
            cout<<b[i][j]<<ends;
        }
        cout<<"\n";
    }
}
int main()
{
    matrix12 ob1,ob2;
    cout<<"1st matrix elements :-";
    ob1.getdata();
    cout<<"2nd matrix elements :-";
    ob2.getdata();
    cout<<"1st matrix:-"<<endl;
    ob1.showdata();
```

```
    cout<<"2nd matrix:-"<<endl;
    ob2.showdata();
    cout<<"addition:-"<<endl;
    ob1+ob2;
    return 0;
}
```

OUTPUT: