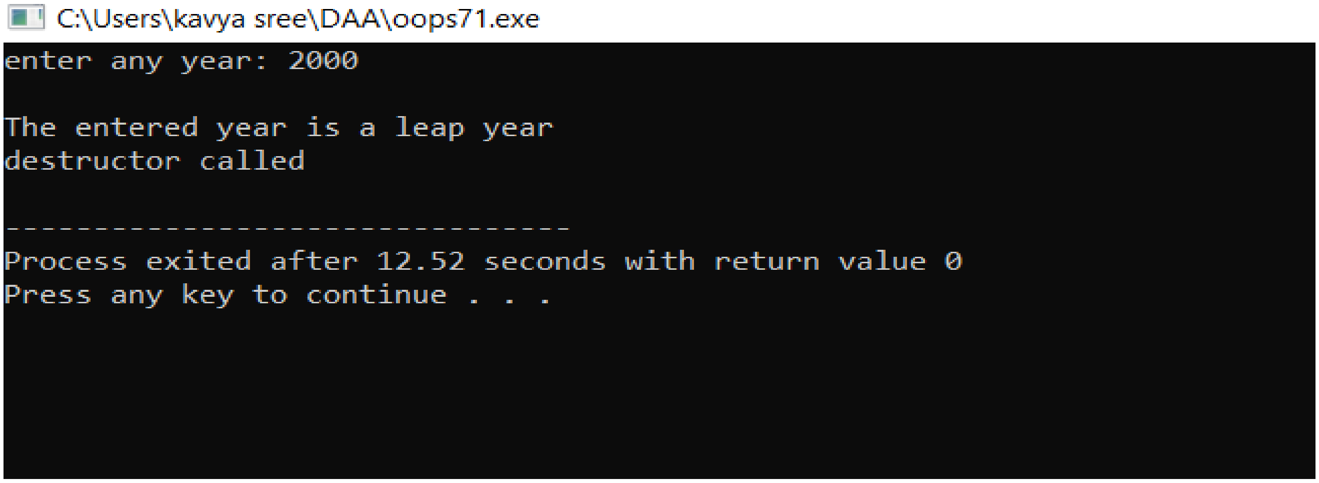
;

LAB-7

```
1. #include <iostream>
using namespace std;
class leapyear
{
private:
int year;
public:
int getdata();
void check_LeapYear(int);
leapyear()                  //calling constructor
{
year=0;
}
~leapyear()                  //calling destructor
{
cout<<"destructor called\n";
}

};
```

```
int leapyear::getdata()
{
cout<<"enter any year: ";
cin>>year;
return year;
}
void leapyear::check_LeapYear(int year)
{
if((year%4==0)&&(year%100!=0))
cout<<"\nThe enetered year is a leap year";
else if(year%400==0)
cout<<"\nThe entered year is a leap year";
else
cout<<"\nThe entered year is not a leap year";
}

int main(int argc, char** argv) {
leapyear c;
int y;
y = c.getdata();
c.check_LeapYear(y);

cout<<endl;

return 0;

}
```

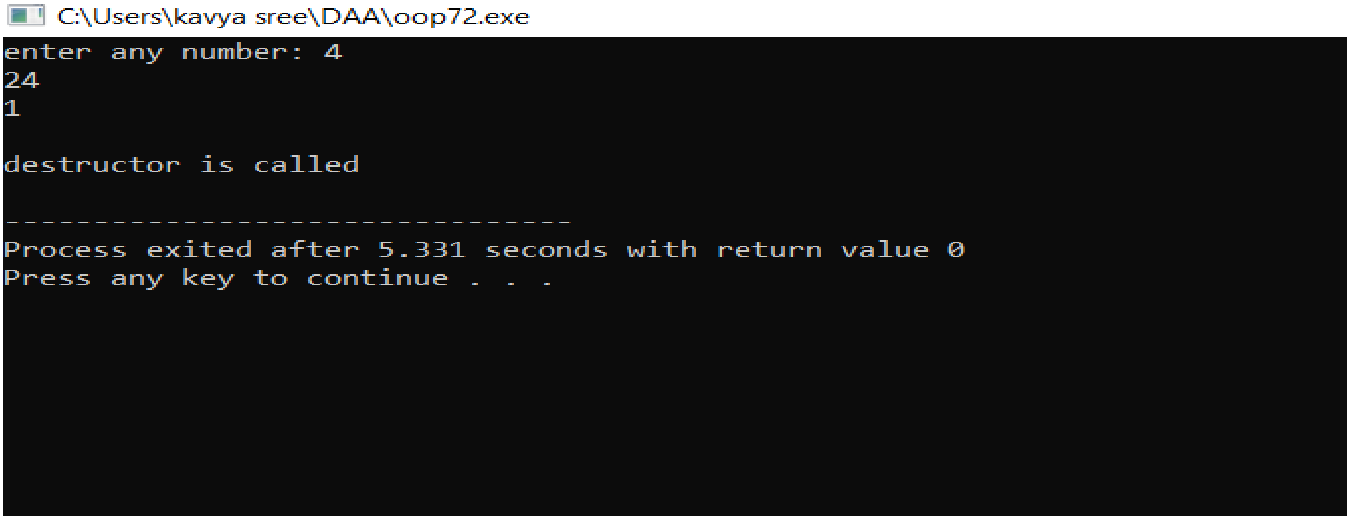


C:\Users\kavya sree\DAA\oops71.exe

```
enter any year: 2000

The entered year is a leap year
destructor called

--------------------------------
Process exited after 12.52 seconds with return value 0
Press any key to continue . . .
```

```
2. #include <iostream>


using namespace std;

class fact{

private:

int n;

public:

fact()              //constructor

{

n=0;
}
fact(int a)
{
n = a;
}
int recursive(int n);
int nonrecursive(int n);
~fact()             //destructor
{
cout<<"\ndestructor is called"<<endl;
}
};
int fact::recursive(int n)
{
if(n<=1)
return 1;
else
{
return n*recursive(n-1);
}
```

```
}
int fact::nonrecursive(int n)
{
int i, factorial=1;
for(i=1;i<=n;i++)
{
factorial = factorial*1;
}
return factorial;
}

int main(int argc, char** argv) {
int number;
cout<<"enter any number: ";
cin>>number;
fact f(number);
cout<<f.recursive(number)<<endl;
cout<<f.nonrecursive(number)<<endl;
return 0;
}
```



C:\Users\kavya sree\DAA\oop72.exe

```
enter any number: 4
24
1

destructor is called

--------------------------------
Process exited after 5.331 seconds with return value 0
Press any key to continue . . .
```