**Problem: If obj is an object of user – defined data type, which has an array of integers as member variable, overload the necessary operators to execute the following statement: obj[3][2][1] \* obj[1][2](5) \* obj**

**Source Code:**

#include <bits/stdc++.h>

using namespace std;

typedef long long int lli;

class sample{

int arr[5];

public:

int sum;

sample();

void input();

sample operator[](int);

void operator=(sample);

sample operator()(int);

int operator\*(sample);

friend sample operator\*(int,sample);

void output();

};

sample :: sample(){

sum = 0;

}

void sample :: input(){

cout << "Enter 5 numbers" << endl;

for(lli i = 0; i < 5; i++)

cin >> arr[i];

}

sample sample :: operator[](int index){

sample ans;

ans.sum = 0;

for(lli i = 0; i < 5; i++)

ans.arr[i] = arr[i];

ans.sum = sum + arr[index];

return ans;

}

void sample :: operator=(sample in){

for(lli i = 0; i < 5; i++){

arr[i] = in.arr[i];

}

sum = in.sum;

}

sample sample :: operator()(int val){

sample ans;

for(lli i = 0; i < 5; i++){

ans.arr[i] = arr[i]\*val;

}

ans.sum = sum\*val;

return ans;

}

int sample :: operator \*(sample in){

return sum \* in.sum;

}

void sample :: output(){

for(lli i = 0; i < 5; i++)

cout << arr[i] << " ";

cout << endl;

}

sample operator\*(int val, sample in){

for(lli i = 0; i < 5; i++){

in.arr[i] = in.arr[i] \* val;

}

return in;

}

int main(){

sample obj;

obj.input();

obj = obj[3][2][1] \* obj[1][2](3) \* obj;

obj.output();

}

**Sample Input/Output:**

Enter 5 numbers

1 2 3 4 5

135 270 405 540 675