

```

#include <iostream>
#include <fstream>
using namespace std;

struct elem {
    int numero;
    elem* pun;
};

struct Biliardo {
    elem* testa;
};

bool elimina(Biliardo& b, int i)
{
    elem *p, *q;
    for(q=b.testa; q!=NULL && q->numero!=i; q = q->pun)
        p=q;
    if (q==NULL)
        return false;
    if (q==b.testa)
        b.testa = b.testa->pun;
    else
        p->pun = q->pun;
    delete q;
    return true;
}

bool aggiungi(Biliardo& b, int i)
{
    if (i<0)
        return false;
    elem* p, *q;
    for(q=b.testa; q!=NULL && q->numero < i; q = q->pun)
        p=q;
    if (q!=NULL && q->numero == i)
        return false;
    elem *r= new elem;
    r->numero = i;
    r->pun = q;

    if (q==b.testa)
        b.testa = r;
    else
        p->pun = r;
    return true;
}

void inizializzaBiliardo(Biliardo& b, int i)
{
    i=(i<0)?0:i;
    b.testa = new elem;
    b.testa->numero = i;
    b.testa->pun = NULL;
}

```

```

bool aggiungiSfera(Biliardo& b, int i)
{
    return aggiungi(b,i);
}

int sottraiBiliardo(Biliardo& b, Biliardo b1) {
    int k = 0;
    for (elem *p = b1.testa; p != NULL; p = p->pun) {
        if(elimina(b, p->numero))
            k++;
    }
    return k;
}

Biliardo* vettoreSfere(int* v, int n)
{
    Biliardo* p = new Biliardo;
    p->testa=NULL;
    for (int i=0; i<n; i++)
        aggiungi((*p), v[i]);
    return p;
}

void stampaBiliardo(Biliardo b, const char* s)
{
    fstream ff;
    ff.open(s, ios::out);
    if (!ff){
        cerr << "Errore nell'apertura del file!" << endl;
        return;
    }
    ff << '<' ;
    elem *p = b.testa;
    while (p!=NULL)
    {
        ff<<p->numero;
        if (p->pun !=NULL)
            ff << ',';
        p=p->pun;
    }
    ff << '>';
    return;
}

int main(){
    Biliardo b;
    inizializzaBiliardo(b, 7);
    aggiungiSfera(b, 3);
    aggiungiSfera(b, 2);
    aggiungiSfera (b, 3);
    stampaBiliardo(b, "output1.txt");
    // File output.txt

    // <2, 3, 7>

```

```

Biliardo b1;
inizializzaBiliardo(b1, 9);
aggiungiSfera(b1, 2);
cout << sottraiBiliardo(b,b1) <<endl;
                                // 1

stampaBiliardo(b, "output2.txt");
                                // File output.txt

// <3, 7>

int v[5] ={3, 8, 1, 2, 3};
Biliardo* p = vettoreSfere(v, 5);
stampaBiliardo(*p, "output3.txt");
                                // File output.txt

// <1, 2, 8>
return 0;
}

/* SOLUZIONI DOMANDE APERTE
*
* DOMANDA 1: 145 in base 8, lo possiamo espandere in base 2 (
ogni cifra su 3 bit)
* Step1: da base 8 a base 2          ...    145 --> 001 100
101
* Step2: raggruppare i bit a 4 a 4    ...    (000)0 0110 0101
* Step3: convertire i 4 bit in una cifra esadecimale ....
* 0000 = 0
* 0110 = 6
* 0101 = 5
* Quindi la conversione di 145(base 8) è 65(base 16)
*
*
* DOMANDA 2:
* L'output è:
* 0x1A4E
* 10
* 11
*
* DOMANDA 3:
* Errore sintattico: riguarda la forma del programma,
rilevato dal compilatore
* esempio -> int i; i = "ciao";
*
* Errore semantico: riguarda il significato del programma,
potrebbe non essere rilevato dal compilatore
* esempio -> int i, j; i = 0; j = 2/i; (divido per 0)
*
*/

```